УДК 512.6, 517.9, 519.6

BASIC DEFINITIONS OF QUANTUM COMPUTING OF IT STUDENTS IN INTELLIGENT COGNITIVE CONTROL AND ROBOTICS

Reshetnikov Andrey[1], Tanaka Takayuki[2], Tyatyushkina Olga[3], Ulyanov Sergey[4]

[1]*PhD in informatics, associate professor;*
Dubna State University,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: agreshetnikov@gmail.com.

[2]*PhD, professor;*
The Graduate School of Information Science and Technology, Hokkaido University;
N14, W9, Sapporo-shi, Hokkaido, Japan;
e-mail:ttanaka@ssc.ssi.ist.hokudai.ac.jp.

3PhD, associate professor;
Dubna State University,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: tyatushkina@mail.ru.

[4]*Doctor of Science in Physics and Mathematics, professor;*
Dubna State University,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: ulyanovsv@mail.ru.

Quantum computations results from the link between quantum mechanics, computer science and classical / quantum information theory. It uses quantum mechanical effects, especially superposition, interference and entanglement, to perform new types of computation which show promise to be more efficient than classical computations. It is the essential trait of the theory of quantum mechanics to make (exclusively) probabilistic predictions, i.e. for a quantum mechanical experiment the theory predicts possible results and their probabilities to occur. This is what makes quantum computing probabilistic.

Keywords: Quantum computing, quantum operators, speed-up of quantum computing, probabilistic inference

ОСНОВНЫЕ ПОНЯТИЯ КВАНТОВЫХ ВЫЧИСЛЕНИЙ ДЛЯ СТУДЕНТОВ ИТ ИНТЕЛЛЕКТУАЛЬНОГО КОГНИТИВНОГО УПРАВЛЕНИЯ И РОБОТОТЕХНИКЕ

Решетников Андрей Геннадьевич[1], Танака Такаюки[2], Тятюшкина Ольга Юрьевна[3], Ульянов Сергей Викторович[4]

[1]*Доктор информатики (PhD in Informatics), к.т.н., доцент;*
ГБОУ ВО МО «Университет «Дубна»,
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: agreshetnikov@gmail.com.

[2]*Доктор наук (PhD in Engineering),*
Высшая школа информатики и технологии,
Университет Хоккайдо;
N14, W9, Саппоро-Ши, Хоккайдо, Япония;
e-mail:ttanaka@ssc.ssi.ist.hokudai.ac.jp.

[3]*Кандидат технических наук, доцент;*
ГБОУ ВО МО «Университет «Дубна»,

Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: tyatushkina@mail.ru.

4Доктор физико-математических наук, профессор;
ГБОУ ВО МО «Университет «Дубна»,
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: ulyanovsv@mail.ru.

Квантовые вычисления основаны на законах квантовой механики, компьютерных технологий и теории информации. При этом применяются квантово-механические эффекты, особенно суперпозиция, интерференция и запутанные состояния, порождая новые типы вычислений, которые являются более эффективными при поиске решений, чем классические вычисления. Особенностью такого рода вычислений является вероятностная природа предсказания ожидаемого результата в силу физической природы законов квантовой механики. Это приводит к вероятностной природе квантовых вычислений и квантовых алгоритмов.

Ключевые слова: Квантовые вычисления, квантовые операторы, быстродействие квантовых вычислений, вероятностный вывод.

Introduction: Main definitions and constraints of quantum computing

The interplay between mathematics and physics has always been beneficial to both fields of endeavor. The calculus was developed by Newton and Leibniz in order to understand and describe the dynamics of motion of material bodies. In general, geometry and physics have had a long and successful symbiotic relationship: classical mechanics and Newton's gravity are based on Euclidean geometry, whereas in Einstein's theory of general relativity the basis is provided by non-Euclidean Riemannian geometry (an important insight taken from mathematics into physics). Although this link between physics and geometry is still extremely strong, one of the most striking connections today is between information theory and quantum physics.

Long time we have not though about computation in physical terms. We considered the *computation from the standpoint of mathematics* and connected it with the notion of algorithm and Turing machine. But computation itself carried out by means of a physical process in a computing device. Computers today become not only faster, they become smaller too. At some stage of miniaturization it is necessary to include in the description of computers a quantum phenomena (Manin, 1980; Feynman, 1982). Deutsch (1985) considered a situation where computers like quantum objects can enter highly non-classical states. These quantum computers could, for example, exist in a superposition of states.

Computation, based on the laws of classical physics, leads to different constraints on information processing than computation based on quantum mechanics. Quantum computers hold promise for solving many intractable problems, but, unfortunately, there currently exist no algorithms for "programming" a quantum computer. The interplay between mathematics and physics has always been beneficial to both fields of endeavor. The calculus was developed by Newton and Leibniz in order to understand and describe the dynamics of motion of material bodies. In general, geometry and physics have had a long and successful symbiotic relationship: classical mechanics and Newton's gravity are based on Euclidean geometry, whereas in Einstein's theory of general relativity the basis is provided by non-Euclidean Riemannian geometry (an important insight taken from mathematics into physics). Although this link between physics and geometry is still extremely strong, one of the most striking connections today is between information theory and quantum physics.

Calculation in a quantum computer, like calculation in a conventional computer, can be described as a marriage of quantum hardware (the physical embodiment of the computing machine itself, such as quantum gates and the like), and quantum software (the computing algorithm implemented by the hardware to perform the calculation). To date, quantum software algorithms, such as Shor's algorithm, used to solve problems on a quantum computer have been developed on an *ad hoc* basis without any real structure or programming methodology.

The lack of a quantum programming or program design methodology for quantum computers severely limits the usefulness of the quantum computer. Moreover, it limits the usefulness of the quantum principles, such as superposition, entanglement and interference, that give rise to the quantum logic used in quantum computations. These quantum principles suggest, or lend themselves, to problem-solving methods that are not typically used in conventional computers.

Quantum principles and quantum logic can be used with conventional computers if we find solutions to simulate and implement quantum algorithms in classical computers. The present paper describes solutions of these and other problems by providing method of simulation and design of quantum algorithm gates to implement quantum algorithms for quantum computing and quantum soft computing that can be classically efficiently simulated.

The interrelations between physics, mathematics and informatics from quantum paradigm point of view is shown in Fig.1. In theoretical backgrounds part we briefly introduce main concepts and ideas of topics depicted on Fig.1.

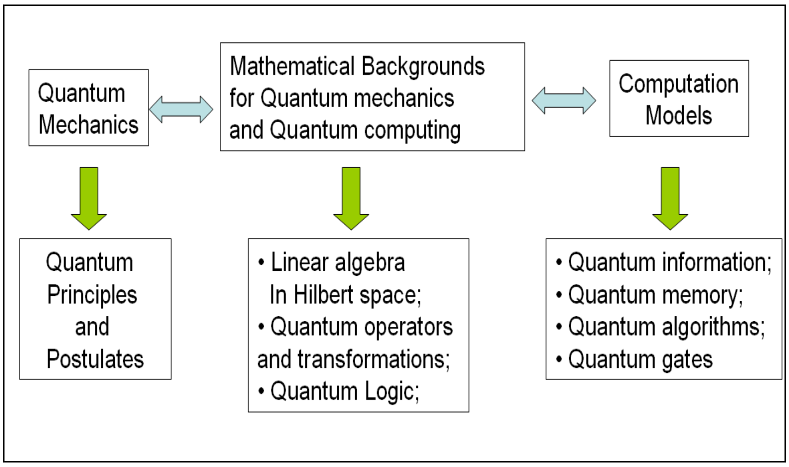
Quantum Mechanics
Mathematical Backgrounds for Quantum mechanics and Quantum computing
Computation Models
Quantum Principles and Postulates
• Linear algebra In Hilbert space;
• Quantum operators and transformations;
• Quantum Logic;
• Quantum information;
• Quantum memory;
• Quantum algorithms;
• Quantum gates

Figure 1. Background of quantum computing

Let us consider a usage of quantum computation ideas for our task formulated above.

The *quantum principles* (such as quantum parallelism, quantum complementary, quantum long-distance correlation, quantum bio-inspired searching, etc.) can be used for applications of quantum strategies for optimal decision making with conventional computers in much the same way that genetic principles of evolution are used in genetic optimizers. Nature also uses the principles of quantum mechanics to solve problems, including quantum-like optimization-type problems, searching-type problems, selection-type problems, etc.

The *quantum operators*, such as *superposition, entanglement* and *interference*, give rise to the quantum logic used in quantum computations. Moreover, the usefulness of these quantum operators gives rise to the new viewpoint on control and self-organization algorithms.

In the classical computation one bit information is coded as 0 or 1. We can say also that one bit of information can represent the state 0 or the state 1. In quantum computation quantum state information is used as a superposition of two states $\left(\alpha_0\left|0\right\rangle+\alpha_1\left|1\right\rangle\right)$ and it is called as *qubit* (quantum bit).

With superposition operator for a given algorithm we can introduce all initial states that include the searching solutions, and accelerate computation processes by massive quantum parallelism.

The entanglement operator has no analog in classical computation. It allows physically set up statistical relations (quantum correlations) between solutions on the searching of space of the algorithm. In particular important case, it is the physical source of quantum oracle algorithms.

The interference operator performs division of solutions obtained by the quantum algorithm by finding a successful solution with maximal probability amplitude.

Quantum principles and quantum logic can be used with conventional computers if we find solutions to simulate and implement quantum algorithms on classical computers.

Remark. That does not mean that deterministic quantum computations are impossible, but that the nature of quantum computing is based on probabilities.

This article describes the fundamental principles of quantum computing from engineering point view. It introduces the quantum circuit model of computation, which provides a "language" to describe quantum algorithms and explains its basic building blocks: quantum bits, quantum operations (gates) and quantum measurements [1-18].

The mathematical framework describing the concepts and principles of quantum mechanics is of course also the theoretical basis of quantum computing. Therefore, it is necessary to deal with the basic postulates of quantum mechanics which connect the physical world with its mathematical model. These postulates directly relate to the modeling of the key elements of quantum computation:

- compositions of such systems,
- operations on quantum systems for the purpose of information processing, and
- the readout (measurement) of information from quantum systems.

Within this introduction the postulates of quantum mechanics are specified in the subsections dealing with the corresponding topics.

Basic knowledge of linear algebra and the tensor product is assumed, but the necessary mathematics can also be found in [1]. The postulates of quantum mechanics can also be found, too (see Appendix 2).

Within this article the postulates of quantum mechanics are specified in the subsections dealing with the corresponding topics.

The main problem of mathematical background of quantum information processing based on Schrödinger and Dirac equations in [2] are discussed. The interrelations between classical and quantum equations using Hamilton-Jacobi formalism are also described in [2] using method of characteristics of partial differential equations.

Basic knowledge of linear algebra and the tensor product is assumed, but the necessary mathematics. The postulates of quantum mechanics can also be found, too.

Short history of quantum computing

The theory of quantum mechanics was established in the mid-1920s. Main contributions were made by M. Born, P. Dirac, W. Heisenberg, E. Schrödinger and others. With quantum mechanics it was possible to explain unknown phenomena raised from various experiments and to resolve inconsistencies in the theories of physics, now designated as classical physics (classical mechanics and classical electrodynamics).

Information can be regarded as not abstract no matter whether it is in someone's mind, written in a book or stored on a magnetic layer of a hard disc — in the words of Rolf Landauer: «Information is physical». It is physics, which sets the main limitations to process and to manipulate information. Since the early beginnings of analog and digital computers classical physics provided the laws for computing devices. The idea of using the laws of quantum physics for information processing did not emerge until the early 1980s. Important influences on the development f the new computation concept are ascribed to Bell (1964), who demonstrated non-local correlations between different parts of a quantum system, as well as Landauer and Bennett, who both dealt with the connection between energy consumption and *irreversibility* of computation.

Remark. A logical gate or function is *reversible*, if the input is uniquely determined by the output, i.e. an inverse function exists mapping the output to its unequivocal input. Otherwise it is *irreversible*.

In 1961 Landauer showed that reassure of information, which is peculiar to irreversible operations, requires the dissipation of energy (*Landauer's principle*). Based on Landauer's work Charles Bennett proved in 1973 that all computation can be performed in principle in a logically reversible manner and therefore does not require dissipation. This result leads in 1980 to Paul Benioff's discovery that quantum systems could perform computation in a coherent manner and to his model of a *Quantum Turing Machine* (QTM). With this proposal the field of quantum computation was born. In 1982 Richard Feynman pointed to the difficulties of classical computers to efficiently simulate quantum physical systems and suggested using computers based on quantum mechanical principles to handle these difficulties. Benioff's QTM was further developed by Deutsch, who also invented the *quantum circuit model of computation*. It can be shown that both models are (nearly) equivalent.

Furthermore, Deutsch formulated an *oracle* problem, today know as *Deutsch's problem*, for which he demonstrated the first (randomized) quantum algorithm that performs better than any comparable classical algorithm. The *Deutsch-Jozsa problem*, a generalization of Deutsch's problem, was the first one that was found to need only linear time on a quantum computer but exponential time on a deterministic Turing machine (although it needs only polynomial time on a probabilistic Turing machine).

A major breakthrough in quantum computing happened in 1994.

First, Simon proposed a quantum algorithm solving an oracle problem in polynomial time on a quantum computer but exponential time on a classical, even probabilistic computer. Simon's work was based on a quantum algorithm introduced by Bernstein and Vazirani.

Inspired by Simon's results Shor published his polynomial time quantum algorithms for integer factorization and discrete logarithm. These quantum algorithms were the first to solve problems of great practical relevance. Both problems are considered to be hard on classical computers.

This difficulty is the basis of many modern public-key cryptography systems such as RSA.

Another quantum algorithm attracted attention in 1996 when Grover presented a quadratic speed-up quantum search algorithm. In the period following, newly discovered quantum algorithms were mainly based on the work of Shor and Grover. It turned out that both computational approaches could be applied to classes of similar problems.

Quantum search walk and quantum games algorithms are examples of speed-up algorithms.

Unfortunately, no other conceptually new quantum algorithms were presented which had such a deep and pioneering impact like Shor's.

A summary of most quantum algorithms is given in [1, 2].

State space of quantum mechanical systems

The model of a quantum computer used here is based on a *closed* or *isolated* quantum mechanical system. This is an ideal system without perturbations and noisy interactions with its surrounding, which are referred to as *decoherence*. Systems in the real world are never absolutely closed. There is always a coupling with the environmental system resulting in a decay of information in the quantum computing device. However, decoherence can be corrected in principle by using *error correcting codes* which also protect against defective quantum operations. Both kinds of *quantum errors*, imperfect operations and quantum noise, are left out of account in [1-3].

The traditional mathematical formalism of quantum mechanics models a closed quantum mechanical system as follows:

Postulate 1. Associated with any closed quantum mechanical system is a *Hilbert space* $\mathcal{H}$ which is a complete (complex) inner-product space. This vector space is also known as the *state space* of the system. Its unit norm vectors are called *(pure) states.*

Remark. A vector is complete if every Cauchy sequence in the space converges, concerning a given norm, to an element in the space. In Hilbert spaces the norm is induced by the inner product. Complex inner product spaces are also called unitary vector spaces.

In quantum computation the state space is limited to finite dimensions. Each state can be regarded as complete description of the physical system.

Notation (Bra/Ket). The standard quantum mechanical notation for (column) vectors in a Hilbert space $\mathcal{H}$ is $|\psi\rangle$. Here, ψ is just a label of the vector. The notation $\langle\psi|$ is used for the vector *dual* to $|\psi\rangle$. This is a row vector, which corresponds to the complex conjugated and transposed column vector $|\psi\rangle$. A column vector $|\psi\rangle$ is sometimes referred to as a *ket*, its dual vector $\langle\psi|$ is referred to as a *bra*. This notation, also called bra-/ket-notation, was invented by Paul Dirac. The inner product of two vectors $|\psi\rangle$ and $|\rho\rangle$ is defined by $\langle\psi|\left(|\rho\rangle\right)\equiv\langle\psi|\rho\rangle$.

An alternative formalism, especially necessary to deal with open and composite quantum systems, uses the *density operator* or *density matrix* notion respectively and the concept of *mixed states.* Both approaches are mathematically equivalent and lead to the same results. The postulated of quantum mechanics can be formulated using both formalisms. As pointed out by Gruska (in Theorem 2.3.46 [3]), «the model of quantum circuits with mixed states is polynomially equivalent, in computational power, to the standard model of quantum circuits over pure states».

Note, within this thesis that a quantum computer is regarded just as an abstract, mathematical object without reference to a specific implementation. Its physical realization is irrelevant; the same applies to possible sources of error.

Quantum information

The simplest possible two-level quantum system and therefore the basic information unit in quantum computing is the *quantum bit*, or qubit for short.

A single qubit

Like its classical counterpart a qubit has two basic states denoted $|0\rangle$ and $|1\rangle$ by analogy with the two values 0 and 1 of a classical bit. But unlike the classical bit a qubit can also be in a *superposition* of its two basic states. Only after the qubit is read out it is with a certain probability in one or the other basic states.

According to the state vector formalism (Postulate 1), a qubit is a unit vector in a two-dimensional complex vector space $\mathcal{H}=\mathbb{C}^2$ with inner product. The states or vectors respectively, $|0\rangle$ and $|1\rangle$, also known as the computational basis states, form an *orthonormal* basis (ON-basis) of this space. Usually $|0\rangle$ and $|1\rangle$ are identified with the standard basis vectors in $\mathbb{C}^2$, $(1,0)$ and $(0,1)$. A qubit in $\mathcal{H}$ can be any arbitrary state formed by linear combination of $|0\rangle$ and $|1\rangle$:

$$|\psi\rangle=\alpha_0|0\rangle+\alpha_1|1\rangle, \tag{0}$$

with $\alpha_0,\alpha_1\in\mathbb{C}$ and $|\alpha_0|^2+|\alpha_1|^2=1$. Here, with $\alpha_0,\alpha_1\neq 0$ the qubit labeled ψ is in a *superposition state.*

The normalization condition relates to equivalence classes of vectors that differ only by a nonzero complex factor. They always describe the same physical state and it is therefore useful to choose unit vectors as representatives of the states. Moreover, the additional condition $|\alpha_0|^2+|\alpha_1|^2=1$ relates to the readout or *measurement* of qubits: Classical bits have to be read to determine their values or states 0 or 1 – the same applies to qubits. However, in quantum computing the outcome of a (single qubit) measurement, «0» or «1», is not deterministic but probabilistic. Measurement of qubit $|\psi\rangle$ gives either the result «0» with probability $|\alpha_0|^2$ or the result «1» with probability $|\alpha_1|^2$. From the normalization condition of probability measures it follows $|\alpha_0|^2+|\alpha_1|^2=1$. By measurement any superposition state collapse to the computational basis state $|k\rangle$ according to the measurement result «*k*». But it also means that, although a qubit is very different from a classical bit, it is not possible to gain more information from a qubit than from a classical bit. Especially, the values of the amplitudes α_0,α_1 are not accessible by measurement. Measurements are discussed in detail below.

Remark. Superdense coding allows communicating two classical bits by transmitting a single qubit of a pair of entangled qubits. At first sight, this might contradict the above statement. However, one needs two qubits perform superdense coding and both qubits must be measured (in the Bell basis).

A geometrical representation of the state of a single qubit is provided by the *Bloch sphere*. It is often used to illustrate the effect of single qubit operations, which are elementary in quantum computing. Unfortunately, there is no equivalent representation for multiple qubits. The Bloch sphere representation of a qubit reads as follows: $|\psi\rangle = e^{\imath\gamma}\left(\cos\frac{\theta}{2}|0\rangle + e^{\imath\gamma}\sin\frac{\theta}{2}|1\rangle\right)$. It is obtained from Eq. (0) by rewriting the complex numbers α_0, α_1 in polar coordinates, $\alpha_0 = re^{\imath\gamma}$ and $\alpha_1 = se^{\imath\delta}$ with $r, s, \gamma, \delta \in \mathbb{R}$ and $r, s \geq 0$, and a suitable choice of the parameters $\varphi = \delta - \gamma$ and $\theta = 2\arccos r$. It is a property of measurement that global phase factors like $e^{\imath\gamma}$ can be ignored. Then, a single qubit state $|\psi\rangle$ can be visualized as a point $(\cos\varphi\sin\theta, \sin\varphi\sin\theta, \cos\theta)$ on the unit sphere in $\mathbb{R}^3$ as it is illustrated in Fig. 2.

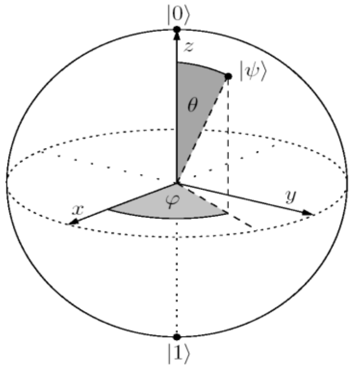

Figure 2. Bloch sphere

The *x*-, *y*- and *z*-axis are defined by the states $1/\sqrt{2}\left(|0\rangle + |1\rangle\right), 1/\sqrt{2}\left(|0\rangle + \imath|1\rangle\right)$ and $|0\rangle$.

Multiple qubits

A *quantum register* is a quantum mechanical system composed if several quantum bits. Considering a system of *n* qubits, its state space is the 2^n-dimensional Hilbert space $\mathcal{H}^{(n)} := \mathbb{C}^{2n}$. Similar to the 1-qubit case, the computational basis states of $\mathcal{H}^{(n)}$, labeled as $|k\rangle \equiv |k_{n-1}...k_0\rangle$, with $k_i \in \{0,1\}$, compare to the 2^n possible states of a classical *n*-bit register. Here, $k_{n-1}...k_0$ is the binary representation of k, where k_i is associated with the *i*-th qubit. In the case where the system I in a computational basis state each qubit has a definite value, either $|0\rangle$ or $|1\rangle$. Note that within this thesis the qubits are counted starting with 0 from the rightmost position in the ket vector (the least significant qubit), as is usual in computer science.

Any linear combination or superposition of the basis vectors is an allowed state of the system (*superposition principle*). Thus, the general (superposition) state of an *n*-0qubit register can be written as $|\psi\rangle = \sum_{k=0}^{2^n-1} \alpha_k |k\rangle$, $\alpha_k \in \mathbb{C}, 0 \leq k \leq 2^n - 1$. Because of the normalization condition of state vectors it is $\sum_{k=0}^{2^n-1} |\alpha_k|^2 = 1$. The probability for the quantum register being in state $|k\rangle$ (measurement result «*k*») is $|\alpha_k|^2$. By convention the 2^n basis states are identified with the standard basis vectors:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \ldots, |2^n - 1\rangle = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}.$$

Mathematically, the extension from one to many qubits or the union of two or more quantum registers to a larger register is made by means of the tensor product $\otimes$:

Postulate 2. The state space of a composite system is the tensor product of the state spaces of the component systems. Let $|\psi_A\rangle$ be the state of system *A* with state space $\mathcal{H}_A$ and $|\psi_B\rangle$ the state of system *B* with state space $\mathcal{H}_B$. Then, the Hilbert space of the bipartite system AB is $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$ and the joint state of the total system is $|\psi_A\rangle \otimes |\psi_B\rangle$.

Moreover, if $\{|\nu\rangle_A\}$ is an ON-basis for $\mathcal{H}_A$ and $\{|\mu\rangle_B\}$ is an ON-basis for $\mathcal{H}_B$, then $\{|\nu\rangle_A \otimes |\mu\rangle_B\}$ is an ON-basis for $\mathcal{H}_{AB}$. In accordance with the tensor product of vector spaces, the dimension of $\mathcal{H}_{AB}$ is the product of the dimension of $\mathcal{H}_A$ and $\mathcal{H}_B$. Therefore, the state space of a quantum register increases exponentially with the number of qubits. Abbreviated notions for the tensor product $|\psi\rangle \otimes |\phi\rangle$ of two arbitrary states $|\psi\rangle \otimes |\phi\rangle$ and $|\phi\rangle$ are $|\psi\rangle|\phi\rangle \equiv |\psi,\phi\rangle \equiv |\psi\phi\rangle$.

As an example, consider a system of two qubits *A* and *B*. The computational basis states $(|00\rangle,|01\rangle,|10\rangle,|11\rangle)$ for system *AB* results for the basis states of *A* and *B* $(|0\rangle,|1\rangle)$ by tensor multiplication: $|x_1 x_0\rangle = |x_1\rangle \otimes |x_0\rangle, \forall (x_0, x_1) \in \{0,1\}^2$.

In multiple qubit systems there are state which cannot be expressed as a tensor product of states of its single qubit components. This property is referred to as *entanglement* or *nonseparability*. Let $|\psi_{AB}\rangle$ be a bipartite state. If there are nay two states $|\phi_{AB}\rangle$ in $\mathcal{H}_A$ and $|\phi_B\rangle$ in $\mathcal{H}_B$, such that $|\psi_{AB}\rangle = |\phi_A\rangle \otimes |\phi_B\rangle$, the state is called separable (or unentangled); otherwise it is *entangled* (or unseparable). The following examples for entangled states of a two qubit system are known as the *Bell* or *EPR states* (due to Einstein, Podolsky and Rosen):

$$|\phi^+\rangle := \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),\ |\phi^-\rangle := \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),$$

$$|\psi^+\rangle := \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle),\ |\psi^-\rangle := \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).$$

When performing a measurement on a subsystem of a composite system with entangled state, another way of thinking about entanglement becomes noticeable. How entanglement is characterized by measurement is described in detail below.

Quantum gates

Roughly speaking, quantum computation means just transforming a state of a given quantum system into another state, usually followed by a measurement. Physicists call this *state transformation* a (time) *evolution* of the quantum system, which can be mathematically represented by a unitary operator.

Remark. The matrix representation U of a linear operator is unitary (and therefore the operator itself), if $U^\dagger U = I$, where $U^\dagger = (U^T)^*$ is the complex conjugate transpose of the U matrix and I is the identity operator.

Postulate 3. The evolution of a closed physical system in a time interval $[t_0, t_1], t_0 < t_1$ is described by a unitary operator $U = (t_0, t_1)$ which depends only on t_0 and t_1. Let $|\psi_t\rangle$ denote the state of the system at time t , then it is $|\psi_{t_1}\rangle = U(t_0, t_1)|\psi_{t_0}\rangle$.

A unitary transformation on n qubit, and thus a vector in $\mathcal{H}^{(n)}$, is a unitary $2^n \times 2^n$ matrix. The set of all unitary matrices of same size is a group in the algebraic sense with the matrix, multiplication as group operation. In particular, it follows:

Theorem 1. If U and V are two unitary matrices (of suitable dimension), then UV unitary as well.

Note that there are infinitely many unitary matrices of a fixed size.

Geometrically, unitary transformations preserve inner products between vectors and with it he length of vectors and the angles between vectors. They are imaginable as rotations of the vector space. Moreover, unitary operators are bijective and therefore reversible.

Following the nomenclature of electrical circuits which consist of wires and logic gates, unitary transformations are called *quantum gates*. May classical gates the most important quantum gates have a certain graphical representation.

A general quantum gate U operating on n qubits is illustrated in Fig. 3 (a). The short wires correspond to the incoming (left) and outgoing qubits (right). Usually, the bottom-most wire corresponds to the least significant qubit (qubit 0). Suppose U is a product of unitary transformations U_1 and U_2, then an equivalents schematic symbol notation, or quantum circuit respectively, is depicted in Fig. 3 (b). Note the different order of U_1 and U_2 in the product notation and on the graphical representation.

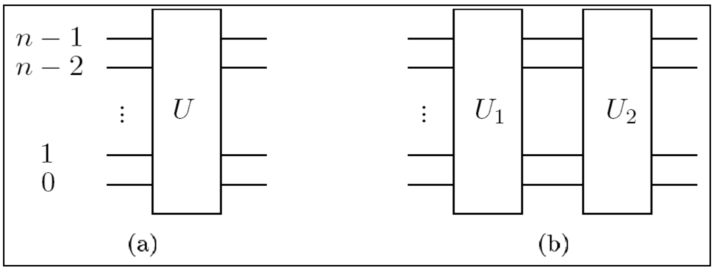

Figure 3. (a) A general n-qubit gate, denoted U. The qubits are counted from bottom (0) to top (n – 1); (b) Quantum circuit implementing $U = U_2U_1$ by means of U_1 and U_2. (Time and control flow in the quantum circuit model goes from left to right)

Single qubit gates

A single qubit gate is identified by a unitary 2×2 matrix. Some important quantum gates are defined by the so-called *Pauli matrices,* denoted X, Y and Z :

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},\ Y = \begin{pmatrix} 0 & -\imath \\ \imath & 0 \end{pmatrix},\ Z = \begin{pmatrix} 1 & 0 \\ o & -1 \end{pmatrix}.$$

Their graphical representation, including the action on an arbitrary single qubit state, is shown in Fig. 4. The *X*-gate is equivalent to quantum *NOT*. Alternative symbols are a box labeled with *NOT* and the ⊕ symbol.

$$\alpha|0\rangle + \beta|1\rangle \ —\boxed{X}—\ \beta|0\rangle + \alpha|1\rangle$$
$$\alpha|0\rangle + \beta|1\rangle \ —\boxed{Y}—\ -\imath\beta|0\rangle + \imath\alpha|1\rangle$$
$$\alpha|0\rangle + \beta|1\rangle \ —\boxed{Z}—\ \alpha|0\rangle - \beta|1\rangle$$

Figure 4. The Pauli matrices

By exponentiation of the Pauli matrices further unitary matrices emerge: calculating the matrix exponentials $e^{-\imath\psi U}$, for $U \in \{X, Y, Z\}$ and $0 \le \psi \le 2\pi$ results in rotations about the *x*-, *y*- and *z*-axis of the

Bloch sphere. From the resulting matrices the following rotation operators R_x, R_y and R_z can be easily derived (using $\cos(\psi) = \cos(-\psi)$ and $\sin(-\psi) = -\sin(\psi)$):

$$R_x(\phi) = \begin{pmatrix} \cos\phi & \imath\sin\phi \\ \imath\sin\phi & \cos\phi \end{pmatrix},\ R_y(\phi) = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix},\ R_z(\phi) = \begin{pmatrix} e^{-\imath\phi} & 0 \\ 0 & e^{\imath\phi} \end{pmatrix}.$$

According to this definition a gate $R_u[\phi], u \in \{x, y, z\}$, is a rotation by 2ϕ about the *u*-axis. In literature, R_x, R_y and R_z are usually defined in a way that they perform rotations by ϕ (by inserting a factor of ½).

The corresponding schematic gate symbols are shown in Fig. 5.

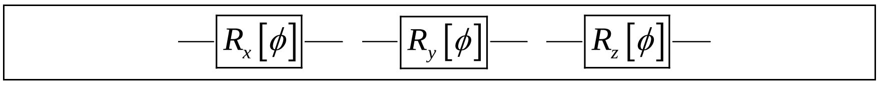

Figure 5. Symbol of the single-qubit gates R_x, R_y and R_z.

An arbitrary unitary operator on a single qubit can be written in different ways as a product of rotation matrices together with an overall phase shift factor. One way is provided by the following theorem:

Theorem 2 *(X-Y decomposition of rotations).* Let U be a unitary single qubit operator. Then $\alpha, \theta, \phi, \psi \in \mathbb{R}$ exist such that $U = e^{\imath\alpha} R_x(\phi) R_y(\theta) R_x(\psi)$.

In particular, each R_z operator can be written as a product of R_x and R_y rotations Moreover, there is also an analogous *Z-Y* decomposition, where in Theorem 2 R_x is substitutes by R_z. Occasionally, such a general one-qubit unitary operator is denoted $U_2(\alpha, \theta, \phi, \psi)$.

R_z gates can also be designated as *phase gates*. This becomes clear when writing them in the form $R_z(\phi) = e^{-\imath\phi}\begin{pmatrix} 1 & 0 \\ 0 & e^{\imath 2\phi} \end{pmatrix}$. Applied to a single qubit state the diagonal coefficient $e^{\imath 2\phi}$ becomes a relative phase factor regarding the $|1\rangle$ amplitude of the state. Because of a peculiarity of quantum measurements the global phase factor is unimportant and can be ignored. The factor 2 in the exponent can be eliminated by defining $PH(\phi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{\imath\phi} \end{pmatrix}$. Up to a global phase factor this class f gates is equivalent to R_z. Besides this definition, the following gate is sometimes referred to as the phase gate:

$$S = PH(\pi/2) = \begin{pmatrix} 1 & 0 \\ 0 & \imath \end{pmatrix}.$$

The square root of gate *S* is gate *T*, the so-called $\pi/8$ gate:

$$T = PH(\pi/4) = \begin{pmatrix} 1 & 0 \\ 0 & {}^{\imath\pi/4} \end{pmatrix}.$$

Remark. The equivalent R_z-gate has diagonal coefficients $e^{\pm\imath\pi/8}$.

Example: Rotation matrices for $SU(2)$. The fundamental $SU(2)$ rotations generated by the Pauli matrices σ_x, σ_y and σ_z are defined as follows:

$$R_x(\theta) = e^{i\frac{\sigma_x\theta}{2}} = \begin{pmatrix} \cos\frac{\theta}{2} & i\sin\frac{\theta}{2} \\ i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \quad R_y(\theta) = e^{i\frac{\sigma_y\theta}{2}} = \begin{pmatrix} \cos\frac{\theta}{2} & \sin\frac{\theta}{2} \\ -\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \quad R_z(\theta) = e^{i\frac{\sigma_z\theta}{2}} = \begin{pmatrix} e^{i\frac{\theta}{2}} & 0 \\ 0 & e^{-i\frac{\theta}{2}} \end{pmatrix}$$

For a general rotation, defined by the unit vector a and the angle θ, we get

$$R_a(\theta) = e^{i\frac{a\cdot\sigma\theta}{2}} = I\cos\frac{\theta}{2} + i(a\cdot\sigma)\sin\frac{\theta}{2}.$$

The effect of conjugating R_a by σ_x is the reversal of the sign of the rotation angle θ iff a is perpendicular to the x-axis: $\sigma_x R_a(\theta)\sigma_x = R_a(-\theta) = R_a^\dagger(\theta) \Leftrightarrow a\cdot u_x = 0$.

Rotations about any single axis are additive: $R_a(\theta_1)R_a(\theta_2) = R_a(\theta_1+\theta_2)$.

A general $SU(2)$ rotation G can be parametrized using the Euler angles (α,β,γ):

$$G = R_x(\alpha)R_y(\beta)R_z(\gamma).$$

Another important single qubit gate is the *Hadamard* gate (H):

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

It maps $|0\rangle$ to $1/\sqrt{2}(|0\rangle+|1\rangle)$ and $|1\rangle$ to $1/\sqrt{2}(|0\rangle-|1\rangle)$. Furthermore, it is $H^2 = I$.

In order to be applicable to an *n*-qubit quantum register with a 2^n-dimensional state vector, quantum gates operating on less than *n* qubits have to be adapted to higher dimensions. For example, let $U = \begin{pmatrix}a & b\\ c & d\end{pmatrix}$ be an arbitrary single qubit gate applied to qubit $q\,(0 \le q < n)$ of an *n*-qubit register. Then the entire *n*-qubit transformation, here denoted with ( ), can be written as a tensor product in the form:

$$U \equiv \underbrace{I\otimes\ldots\otimes I}_{n-(q+1)} \otimes U \otimes \underbrace{I\otimes\ldots\otimes I}_{q}.$$

What was intuitively clear is now confirmed: an empty wire in the graphical representation of quantum gates can be identified with the identity matrix. The matrix U consists of $2^{n-(q+1)}$ major «blocks» on the diagonal, where each block contains 2^q matrices U which are diagonally arranged and shifted by one position.

The structure of U with $n=4$ and $q=1$ is illustrated in Fig. 6.

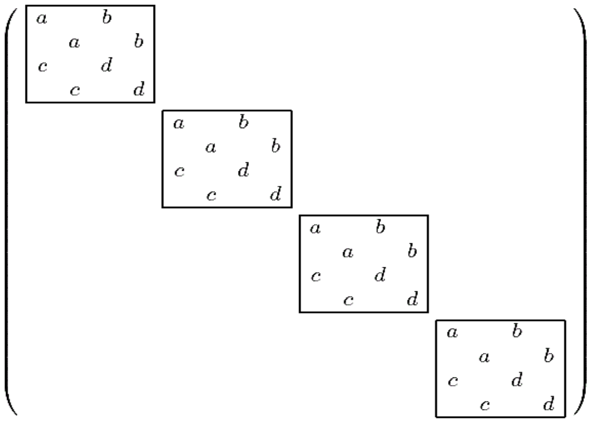

Figure 6. U consists with n=4 and q=1, of four major blocks on the diagonal, where each block contains the matrix U twice. (Within one block the structure of the 2×2 matrices is «broken open» and the matrices overlap each other)

Calculating the new quantum state requires 2^{n-1} matrix-vector-multiplications (each block, each submatrix) of the 2×2 matrix *U*. It is easy to see that the costs of simulating quantum circuits on convectional computers grow exponentially with the number of qubits. In the same way as U is a composition of I gates and U, other transformations can be built by tensor multiplication from single qubit gates which operate in parallel on different qubits.

More generally:

Theorem 3. Let U be a unitary operator on $\mathcal{H}^{(m)}$ and V a unitary operator on $\mathcal{H}^{(n)}$. Then $U \otimes V$ is a unitary operator on $\mathcal{H}^{(mn)}$.

For example, applying the Hadamard gate on each qubit of an n qubit quantum register realizes the unitary transformation $\underbrace{H \otimes ... \otimes I}_{n \quad times} = H^{\otimes n}$. But there are also multiple qubit gates which cannot be decomposed into a tensor product of single qubit transformations, i.e., they are *unseparable*. This, of course, relates to entanglement of quantum states, as entangles states can only be generated by using unseparable gates.

Controlled operations

The controlled-*NOT* gate, also referred to as *CNOT*, operates on two qubits, a *control qubit* and a *target qubit*. The action of the *CNOT* is as follows: it flips the target qubit if the control qubit is set to $|1\rangle$ and leaves it unchanged otherwise. Suppose two qubits $|ct\rangle$ are given where the first qubit it is target qubit and the second is the control qubit. Then, the effect of *CNOT* on the computational basis states is given by $|c\rangle|t\rangle \rightarrow |c\rangle|t \oplus c\rangle$, with $c, t \in \{0,1\}$. Its matrix representational and gate symbol is shown in Fig. 7.

$$\begin{array}{c|cccc} & 00 & 01 & 10 & 11 \\ \hline 00 & 1 & 0 & 0 & 0 \\ 01 & 0 & 1 & 0 & 0 \\ 10 & 0 & 0 & 0 & 1 \\ 11 & 0 & 0 & 1 & 0 \end{array}$$

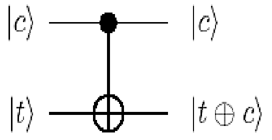

Figure 7. The CNOT gate operates on two qubits: the solid circle indicates the control qubit and the symbol $\oplus$ indicates the target qubit. (The bit tuples labeling the matrix rows and columns indicate the order of basis states)

It is easy to prove that the *CNOT* cannot be decomposed into a tensor product of two single qubit transformations. In a similar way the $\left(C^{*}NOT\right)$ gate is defined on $k+1$ qubits. It flips the target-qubit if the k control-qubits are 1. For $k = 2$ this gate is called a *Toffoli* gate or *CCNOT*.

It acts on the computational basis states as follows: $|a,b,c\rangle \rightarrow |a,b,c \oplus ab\rangle$ for $a,b,c \in \{0,1\}$, where a and b denotes the two control qubits and c the target qubit. Essentially, by preparing qubit c to 1 the outgoing target qubit becomes $\neg(ab)$. Another useful 2-qubit operation is SWAP which interchanges the states of the two input qubits: $|a,b\rangle = |b,a\rangle$. It can be implemented as a sequence of tree *CNOT*s. Its schematic symbol and decomposition in *CNOT* gates is shown in Fig. 8.

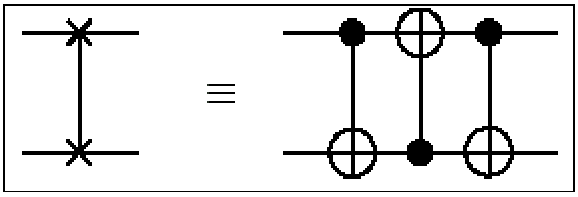

Figure 8. The SWAP gate and the equivalent circuit using CNOT gates. (The matrix SWAP is obtained by multiplication of the corresponding CNOT matrices)

More generally, let U be an m-qubit unitary operator. Then, a controlled operation $C^k\left(U\right)$ on $k+m$ qubits acts on the m target qubits like the U-gate, oriented all k control qubits are 1. Otherwise it has no effect. For example, controlled phase gates with control qubit 1 and target qubit 0 are given by

$$CPH\left(\phi\right)=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&e^{\imath\phi}\end{pmatrix}.$$

Similar to single-qubit gates, controlled quantum gates have to be adapted to higher dimensions, if required by the Hilbert-space. Regarding a controlled operation $C^k\left(U\right)$ with single-qubit gate U, the number of matrix-vector-multiplications of U for calculating the new quantum state is reduced to 2^{n-1-k}. In that case, all diagonal coefficients, assigned to basis states which do not meet the control conditions, are set to 1, as exemplified by Fig. 9.

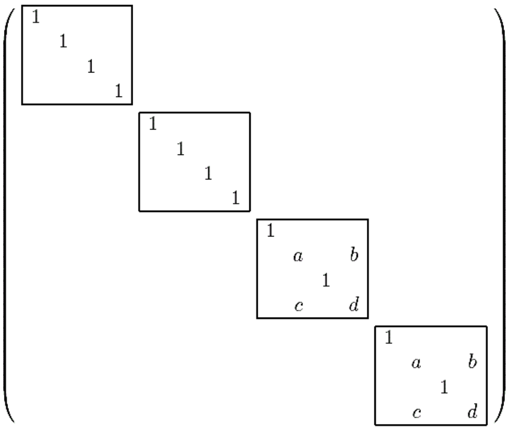

Figure 9. Controlled-U transformation with control qubits {0, 3} and target qubit 1 on a 4-qubit quantum computer. To calculate the new quantum state, only two multiplications of U with the corresponding sub vector of the current state vector are required

Sets of universal quantum gates

A gate or a set of gates is defined to be *universal* for classical computing, if any arbitrary gate or function can be computed requiring only those gates. Since the *NAND* gate is universal in classical computing and has a quantum equivalent provided by the Toffoli or *CCNOT* gate, the set of all quantum circuits comprises all classical circuits. However, this is only a small subset. In contrast to the discrete space of all classical operations the set of quantum operations is continuous. Therefore, the concept of universality for a *discrete* set a quantum gates rests on «good approximations» of arbitrary unitary operations.

In this context it is quite helpful that the sufficiently strong causality principle — similar causes have similar effects — applies also to quantum circuits, that is, small changes or errors in a single unitary operation or a gate sequence cause only small changes in the outcome of the circuit. Specifically, considering a computation where several quantum gates with a common bounded error are applied to an initial state $\left|\psi\right\rangle$, the accumulated error in the resulting state grows linearly with the length of the circuit. This motivates the following *definition*:

Let U and V be unitary operators acting on a Hilbert space $\mathcal{H}^n$. V is called an *approximation of U with error* ϵ, if $\epsilon=\max_{|\psi\rangle}\left\|\left(U-V\right)\left|\psi\right\rangle\right\|$, with $\left|\psi\right\rangle\in\mathcal{H}^n,\left\|\left|\psi\right\rangle\right\|=1$. A set of quantum gates is called universal, if any unitary transformation can be approximated to arbitrary accuracy by a quantum circuit consisting of the gates from that set.

The following two theorems comprise the most important results about the universality of quantum gates and the approximation of quantum circuits.

Theorem 4 *(Universality of single qubit and CNOT gates).* An arbitrary unitary operation *U* on *n* qubits $\left(U:\mathcal{H}^n \to \mathcal{H}^n\right)$ can be realized *exactly* by a quantum circuit requiring $O\left(n^2 2^{2n}\right)$ gates from the set of *CNOT* and single qubit gates.

Theorem 5 (*Universality with a discrete set*). The discrete gate set $\left\{H, CNOT, T\right\}$ forms a universal basis for quantum computation. Moreover, let *U* be an arbitrary unitary operation which can be realized exactly by a quantum circuit containing *m* gates from the set of *CNOT* and single qubit gates. Then, this circuit can be approximated to an accuracy ϵ using $O\left(m \log^c \left(m/\epsilon\right)\right)$ gates from the discrete gate set, where *c* is a constant approximately equal to 2. By adding gate *S*, the approximations can be done fault-tolerantly.

Unfortunately, most unitary transformations cannot be *efficiently* implemented from a small set of elementary gates, i.e. given a unitary transformation *U* on *n* qubit; there is no circuit of size polynomial in *n* approximating *U*.

Decomposition of unitary transformations

Theorem 4 is proven by explicitly constructing a decomposition of an arbitrary unitary matrix into single qubit and *CNOT* gates. On the following, this construction is outlined briefly. In this context, an improvement of this construction is mentioned which was introduced by Ago et all. Here, only the idea behind their approach is described.

The construction of a decomposition can be done in two steps: first, expressing the general unitary matrix of dimension *d* as a product of at most $d\left(d-1\right)/2$ *two-level unitary operators*, that is, matrices which act only non-trivially on two or fewer vector components, as shown in Fig. 10, and second, implementing an arbitrary two-level matrix by single qubit and CNOT gates.

$$U_{s,t} = \begin{pmatrix} 1 &&&&&&&& \\ & \ddots &&&&&&& \\ && 1 &&&&&& \\ &&& a & \cdots & b &&& \\ &&& \vdots & \ddots & \vdots &&& \\ &&& c & \cdots & d &&& \\ &&&&&& 1 && \\ &&&&&&& \ddots & \\ &&&&&&&& 1 \end{pmatrix}$$

Figure 10. A two-level unitary matrix $U_{s,t}$ *with a* 2×2 *(unitary) component matrix* $\tilde{U}$ *consisting of* $a,b,c,d \in \mathbb{C}$ *(The coefficients* a *and* b *are in row* s *, the coefficients* c *and* d *are in row* t *)*

The first step is also called the *two-level decomposition*. G. Cybenko describes this decomposition in terms of traditional algebraic operations as a classical triangulation or *QR*-factorization. Since classical *QR*-factorization is typically based on real valued givens rotations, which are matrices like the one in Fig. 9, except that the component matrix is a real-valued rotation matrix $U = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}$, he calls the two-level unitary matrices *quantum givens operations*. Alternatively it can be explained by means of *Gray code sequences*.

Now, let be *s* and *t* the *n*-bit binary representations of the two basic states, a two-level matrix $U_{s,t}$ with component matrix U acts on. Let bit *k* be a bit, for which *s* and *t* differ. Then, the quantum circuit for the two-level operator is performed by the following three steps: First, apply *CNOT*, with bit *k* as the control bit, on every bit for which *s* and *t* differ (expect bit *k*) and apply *NOT* on every bit of *s* which is 0 (expect bit k). Second, apply U on qubit *k* with all other qubits being control qubits $\left(C^{n-1}\left(U\right)\right)$. Finally, apply all the *NOT* and *CNOT* gates of the first step again, but in reverse order, undoing all permutations. The entire implementation requires $O\left(n\right)$ gates. Of course, the $C^{n-1}\left(U\right)$ gate can be reduced to a sequence of *CNOT* and single-qubit operations as well, requiring another $O\left(n\right)$ gates.

Thus, the entire implementation of an arbitrary unitary matrix uses $O\left(n^2 2^{2n}\right)$ singles qubit and *CNOT* gates. Aho and Svore present a decomposition algorithm with an improved two-level decomposition phase using a technique, which they call *Palindrome Transformation.*

Remark. They call the process that generates for an arbitrary unitary matrix an exact decomposition a *quantum circuit compilation*.

The idea behind this technique is, to find an optimal ordering of two-level operations in the first phase, such that the ordering of the *palindromic subcircuits* of self-inverting gates, resulting from the second phase, leads to a maximal amount of cancellations of the self-inverting gates: A palindromic subcircuit *A* is a gate sequence of the form $A_1A_2...A_kVA_k...A_2A_1$. Two successive palindromic subsequences *A* and *B* can have a subsequence of self-inverting gates in common, like $...A_{j+1}A_j...A_2A_1A_1A_2...A_jB_{j+1}...$, with $B_1 = A_1...B_j = A_j$ which can be reduces to $...A_{j+1}B_{j+1}...$.

It is shown that the *Palindromic Optimization Algorithm* (POA) achieves a large benefit, resulting in significantly smaller decompositions than those obtained by the conventional method. However, even for small numbers of qubits, the resulting quantum circuits are still large. It is unknown, whether there exist more efficient decomposition algorithms. Therefore, other approached might be necessary to find even shorter decompositions. Such a different approach to find optimal quantum circuits is provided by evolutionary algorithms.

Oracle gates

In computer science an oracle is a black-box function, that is, a function whose internal working is unknown. An input for the oracle is directly processed into an output. It is said that the oracle responds on the query immediately. This implies that the costs of operating a black-box are irrelevant for complexity analysis. An oracle gate in quantum computing is usually a «variable» gate. It enables the encoding of problem instances and represents in this way the input of a quantum algorithm. Oracle gates may change from instance to instance of a given problem, while the «surrounding» quantum circuit remains unchanged. Consequently, a proper quantum circuit solving a given problem has to achieve the correct outputs (after measurement) for all oracles representing problem instances.

In certain quantum algorithms, like Grover's or Deutsch's, oracle gates are permutation matrices computing Boolean functions $f:\{0,1\}^n \to \{0,1\}$. The transformation can be defined by the map $U_f:\left|x,y\right\rangle \to \left|x,y\oplus f\left(x\right)\right\rangle$, where $\left|x\right\rangle$ is an n-qubit state, $\left|y\right\rangle$ is a single qubit state and $\oplus$ indicates the addition modulo 2. For $y=0$ the final state of the single qubit becomes $\left|f\left(x\right)\right\rangle$.

The oracle matrix for f inverts the output qubit, iff f yields «1» on the input qubits. In this case, the matrix swaps the amplitudes between the states differing only with respect to the output bit. As an example, Fig. 11 shows a reversible matrix implementing OR on two input bits.

$$\left(\begin{array}{cccc|cccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\hdashline
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{array}\right)$$

Figure 11. Example for an oracle matrix, implementing the OR function of two inputs. The right-most qubit is flipped, if at least one of the two other qubits is «1»

The symbol representation of an oracle gate is usually just an appropriately labeled box, covering all the qubits the oracle is operating on. In cases where the oracle gate corresponds to a Boolean function the target qubit with the function value may be marked by the $\oplus$ symbol.

Behind every oracle gate is a particular quantum circuit calculating the output. Such a quantum circuit usually gets additional inputs and needs also additional ancillary qubits which are left out in the symbol representation of the oracle. This can be done, because the additional qubits are not needed for the remainder of the computation and using a technique called *uncomputing*, one can get rid of the «garbage» assigned to the ancillary qubits and put them back to the initial base state. For instance, a quantum circuit for U_f might have additional input qubits encoding a Boolean function *f* which is then calculated at the output. Since any classical circuit can be simulated efficiently (in linear time) by a quantum circuit, usually one does not need to deal with the exact implementation.

Projective measurements

Quantum information processing is useless without readout or measurement respectively. It is the final step in quantum algorithms, since there is no other way to gain information about the quantum system than by measurement. This section deals only with the *projective* or von *Neumann measurement* in the computational basis, which is a special case of a more general quantum measurement described by measurement operators. However, all other kinds of measurements proved to be equivalent to unitary transformations, using auxiliary qubits, so-called *ancillae*, if necessary, followed by projective measurements.

Before explaining the effects of projective measurements on quantum systems the concept of *observables* is introduced. An observable *M* is a property of a physical system that can be measured. Mathematically *M* is a Hermitian (self-adjoint) operator in a Hilbert space with a spectral decomposition $M = \sum_i \lambda_i \left|i\right\rangle\left\langle i\right|$, where λ_i are the eigenvalues of *M*. The corresponding eigenstates $\left|i\right\rangle$ of *M* form an *ON*-basis in the vector space. Here, $P_i := \left|i\right\rangle\left\langle i\right|$ is the projector into the eigenstate $\left|i\right\rangle$.

Postulate 4. A projective measurement on a quantum system is described by an observable *M*. The possible outcome of a measurement of *M* is an eigenvalue λ_i of *M*. After the measurement, the quantum state is an eigenstate of *M* corresponding to the measured eigenvalue. If the quantum state of the system just before the measurement is $\left|\psi\right\rangle$, then the probability of getting result λ_i is given by $\mathrm{P_r}(i) = \left\| p_i \left|\psi\right\rangle \right\|^2 = \left\langle\psi\right| P_i \left|\psi\right\rangle$. If the outcome is λ_i, the normalized post-measurement state becomes $\dfrac{P_i\left|\psi\right\rangle}{\sqrt{\Pr(i)}}$. A projective measurement *in the computational basis* $\left\{\left|k\right\rangle\right\}$ implies the use of projectors $P_k = \left|k\right\rangle\left\langle k\right|$ to perform the projective measurement. For measurements in the standard basis the numerical outcomes λ_i are identified with «*i*». In this case, a superposition state prior to the measurement collapses with the measurement to $|i\rangle$.

Furthermore, it is not important to known the eigenvalues λ_i (and thus *M*), since probabilities $\Pr(i)$ and post-measurement states do not depend on their numerical values. In the following, the term «measurement» always refers to projective measurements in the computational basis.

A *partial* measurement of a single qubit *q* in an *n*-qubit register with outcome «*i*» is a projection into the subspace, spanned by all computational basis vectors with $q=i$. The probability $\Pr_q(i)$ of measuring a single qubit with result «*i*» is the sum of the probabilities for all basis states with $q=i$ and the post-measurement state is just the superposition of these basis states, re-normalized by the factor $1/\sqrt{\Pr_q(i)}$. For example, measuring the first (right-most) qubit of $|\psi\rangle = \alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle$, with $\alpha_0, \alpha_1, \alpha_2,, \alpha_3 \in \mathbb{C}$ and $|\alpha_0|^2 + |\alpha_1|^2 + |\alpha_2|^2 + |\alpha_3|^2 = 1$, gives «1» with probability $|\alpha_1|^2 + |\alpha_3|^2$, leaving the post-measurement state $|\psi'\rangle = 1/\sqrt{|\alpha_1|^2 + |\alpha_3|^2}\left(\alpha_1|01\rangle + \alpha_3|11\rangle\right)$. The projectors are just $P_i = I \otimes |i\rangle\langle i|$.

It can be proved that multiple qubit measurements can be treated as a serried of single qubit measurements. Note that quantum measurements are irreversible operators, though it is usual to call these operators measurement *gates*. In this thesis a single qubit measurement is assigned the schematic symbol illustrated in Fig. 12.

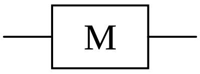

Figure 12. Circuit symbol for a single qubit measurement

The quantum effect of entangled states was already discussed. Measurements provide another equivalent way to define entanglement. A multiple qubit quantum state is not entangles if the measurement of one single qubit has no effect on any other single qubit measurement.

Consider the entangled Bell state $|\phi^+\rangle = 1/\sqrt{2}\left(|00\rangle + |11\rangle\right)$. Provided the second qubit has not been measured before, the probability of measuring the first qubit to be $|0\rangle$ is $1/2$, and vice versa. If a measurement is performed either on the first or the second qubit, the measurement of the other qubit in each case gives the same result. That is, the measurement of one qubit has an effect on the measurement of the other – the measurement outcomes are correlated.

An important principle about measurement in the context of quantum circuits is discussed in the following section.

Quantum circuits

The quantum circuit model of computation is analogous to the classical circuit model. A classical circuit is made of gates computing Boolean functions and wires that connect gates. A quantum circuit has nearly the same structure, but with some restrictions.

Remark. Formally, it can be described by a (acyclic) directed graph whose vertices represent the gates and whose directed arcs represent wires.

Of course, gates in quantum circuits are quantum gates. Their graphical representation is already described. Wires, symbolized by horizontal lines, connect quantum gates and indicate input and output of a quantum circuit. Each wire represents a certain qubit. However, wires generally do not correspond to physical wires, but refer instead to a «time line». The quantum circuit fixes the chronological order of unitary transformations (plus some measurements), which are applied successively to an initialized quantum state. If not stated otherwise the input state of an *n*-qubit quantum circuit is the computational basis state

$|0\rangle^{\otimes n} = |0\rangle \otimes \cdots \otimes |0\rangle$. By convention a quantum circuit has to be read from left to right. The simplest quantum circuit it is single quantum gate. However, to implement this gate, it usually has to be decomposed into a sequence of elementary gates.

The following rules specify the restriction of quantum circuits compared with classical circuits and define allowed connections of quantum gates in the common quantum circuit model:

1.	Only acyclic circuits are valid quantum circuits, i.e. loops are not allowed. A quantum gate which into be applied repeatedly has to be wired as many times in the quantum circuit.
2.	Also, wire crossings are not allowed since arbitrary qubit permutations can be realized by *SWAP* operations.
3.	As a result of the reversibility of operations in quantum circuit it is not allowed to perform the *FANIN* operation, which joins several wires to a single wire containing the bitwise *OR* of the inputs.
4.	Moreover, the number of input and output wires or qubits respectively is exactly the same.
5.	In contrast to classical circuits, it is not possible to split a wire into two or more identical wires, which means, the *FANOUT* operation is not allowed. This corresponds to the following theorem.

Theorem 6 *(No-cloning theorem).* It is not possible to make a copy of an unknown quantum state.

Even thought (universal) cloning is not possible, classical information can be copied with perfect fidelity, as any particular pair of orthogonal states can be cloned perfectly.

In the following, the main aspects of quantum circuits are summarized, beginning with a working definition of a quantum circuit and continued with explanations on inputs and outputs of quantum circuits.

A quantum circuit is a quantum computational model operating on a finite number of qubits. A *quantum circuit on n qubits* is a unitary operation on $\mathcal{H}^{(n)}$ which can be represented by a finite concatenation of quantum gates. In particle, the unitary operation is to be composed of elements from a (small) finite set of quantum gates which form a universal basis for quantum computation. Since discrete, universal gate sets realize any unitary operation with arbitrary accuracy (Theorems 4 and 5), this restriction (also important in the context of circuit evolution) does not limit the set of computable functions. A quantum circuit can get its *input* in two ways: (i) by the initial quantum state (the state of the qubits); or (ii) by input gates (or oracle gates), that is, unitary operations which depend on the input. The first approach is more intuitive and corresponds to the way classical circuits obtain their input. However, encoding inputs may lead to quantum systems with several qubits. Since costs for circuit evaluations on convectional computers increase exponentially with the number of qubits. Large numbers must be avoided. This is possible using the second approach.

Oracle or input gates usually substitute much larger quantum circuits and hide qubits necessary to encode further problem inputs and ancillary qubits. Yet, the use of oracle gates leads to a different complexity measure, if it is assumed, that the oracle performs its operation in a single time step. Using input gates does not contradict to the definition of quantum circuits given above. The input gate in merely integrated in the unitary operation and can be seen as an element of the elementary gate set.

Applying the quantum circuit means multiplying the unitary operation with the initial quantum state. The resulting vector provides the probabilities for all measurement outcomes. A measurement is necessary to obtain any information. From this the *output* of the quantum circuit can be inferred. How this is done is essentially convection. For instance, for a decision problem one can define a particular qubit to carry the answer. In a different way the output of the Deutsch-Jozsa algorithm (described at the end of this item) is obtained. Here, the measurement result of the entire quantum system is decisive for the outcome: One basis state encodes the answer «*f* is constant», all others the answer «*f* is balanced». Moreover, for optimization problems every quantum state may encode a certain solution. So, there are usually many ways to define the output modalities and the decision in favor of either way will affect the quantum circuit solving a given problem.

Note, it is still an open issue whether there exists other models of commutation which are more powerful than the quantum circuit model.

Intermediate measurements

The final element in quantum circuits is the measurement. In circuits without explicit measurements at the end they are implicitly assumed. However, measurements can also be performed as an intermediate step in the circuit. Moreover, they allow conditional branchings in quantum circuits. The advantage of intermediate measurements is that they tend to make quantum circuits more «readable» and interpretable, when they are used in a clever and effective way.

In case of a single-qubit intermediate measurement, depend on the measurement result «0» or «1» one of two quantum subcircuits is applied and describes now the continued evolution of the quantum system. Multiple-qubit intermediate measurements are composed of single-qubit intermediate measurements. Therefore, it is sufficient to focus on the latter. The possibility to use intermediate measurements extends the quantum circuit model described above and requires a new definition: A *quantum circuit on n qubits with intermediate measurements* is a binary tree with quantum (sub)circuit on *n* qubits as nodes. In addition, all inner nodes of the tree are labeled with a single-qubit measurement (the qubit it acts on). By definition the left subtree corresponds to measurement outcome «1», the right to measurement outcome «0».

Applying such a circuit means evaluating a path from the root to a leaf: The quantum circuit in the root is applied to initial quantum states. Each application of a quantum circuit belonging to an inner node is followed by a single qubit measurement which determines the next subtree and the new quantum state the subcircuit is applied to.

According to the *quantum principle of deferred measurement*, «measurements can always be moved from an intermediate stage of a quantum circuit to the end of the circuit». Of course, such a shift has to be compensated by some other changes in the quantum circuit. The transfer of a quantum circuit with intermediate measurements to a quantum circuit without them might lead to a much larger quantum circuit (in the number of elementary quantum gates).

Note that this circuit model with intermediate measurements seems to be different to the circuit model (with intermediate measurements) roughly described by Nielsen and Chuang. Moreover, the quantum circuit models with and without intermediated measurements are computationally equivalent, but it is not quite clear how the circuit size of a quantum circuit with intermediate measurements is related to the circuit size of its equivalent circuit without intermediate measurements.

Mutually unbiased computational bases.

Measurements in a special class of bases, i.e., mutually unbiased bases, not only form a minimal set but also provided the optimal way of determining a quantum state. Mutually unbiased measurements (MUB) corresponds to measurements that are as different as they can be so that each measurement gives as much new information as one can obtain from the system under consideration. In other words MUB operators are maximally noncommuting among themselves. If the result of one MUB can be predicted with certainty then all possible outcomes of every other measurement, unbiased to the previous one are equally likely. The MUB observables can provided an explicit consideration as tensor product of the Pauli matrices for dimension $d = 2^m$.

Remark. When d = 2 the mutually unbiased operators are the three Pauli matrices, but unfortunately this observation cannot be generalized in a straightforward way to higher dimension. In addition to the obvious importance of MUB in the context of quantum state determination and foundations of quantum mechanics and before continuing it is useful to provide a formal definition of MUB.

Definition: Let $B_1 = \{|\varphi_1\rangle, \ldots, |\varphi_d\rangle\}$ and $B_2 = \{|\psi_1\rangle, \ldots, |\psi_d\rangle\}$ be two orthonormal bases in the *d*-dimensional state space. They said to be *mutually unbiased bases* (MUB) if and only if (iff) $\left|\langle\varphi_i|\psi_j\rangle\right| = \frac{1}{\sqrt{d}}$, for every $i, j = 1, \ldots, d$. A set $\{\mathcal{B}_1, \ldots, \mathcal{B}_m\}$ of orthonormal bases in $\mathbb{C}^d$ is called a set of MUB if each pair of bases $\mathcal{B}_i$ and $\mathcal{B}_j$ are mutually unbiased.

Remark. The simplest example of a complete set of MUB is obtained in the case of spin-$\frac{1}{2}$ particle, where each unbiased basis consists of the normalized eigenvectors of the three Pauli matrices respectively. However, the analysis of a set of MUB corresponding to a two level quantum system does not capture one of the basic features of MUB, i.e., its importance in determining the quantum state. In the case of two level systems, the density operator has three independent parameters and almost any choice of the three measurements is sufficient to have the complete knowledge of the system. This is not true in general for any other dimension greater than two, where the existence of MUB becomes more crucial in the context of minimal number of required measurements for quantum state determination.

Remark. *Ivanovic* (1981, 1997) form the first time showed that for any prime dimension *d*, there is a set of *d* + 1 MUB. There is a nice symmetrical structure behind these bases, and their existence as a consequence of properties of Pauli operators on *d*-state quantum systems.

Example. Let $\mathcal{M}_d\left(\mathbb{C}\right)$ be the set of $d\times d$ complex matrices. In a natural way, the set $\mathcal{M}_d\left(\mathbb{C}\right)$ is a d^2 – dimensional linear space. Each matrix *A* in $\mathcal{M}_d\left(\mathbb{C}\right)$ can be also naturally considered as a d^2 – dimensional complex vector$\left|\vartheta_A\right\rangle$, where the entries of the matrix *A* being regarded as the components of the vector$\left|\vartheta_A\right\rangle$. In this way, for matrices $A,B\in\mathcal{M}_d\left(\mathbb{C}\right)$we can define the inner product $\left\langle A,B\right\rangle$ of matrices as the inner product $\left\langle \vartheta_A,\vartheta_B\right\rangle$ of vectors. In this case we have the following result: $\left\langle A,B\right\rangle = Tr\left(A^{\dagger}B\right)$.

We say the matrices $A,B\in\mathcal{M}_d\left(\mathbb{C}\right)$ are orthogonal *iff* $\left\langle A,B\right\rangle = 0$.

Example: *The existence of* $p+1$ *MUB in the space*$\mathbb{C}^p$, *for any prime p*. As mentioned above, this result first shown by *Ivanovic* (1981, 1997), by explicitly defining the MUB, that these bases are in fact bases each consists of eigenvectors of the unitary operators $Z, X, XZ,\ldots, XZ^{d-1}$, where *X* and *Z* are generalizations of Pauli operators to the quantum systems with more than two states. There is a useful connection between MUB and special types of bases for the space of the square matrices. These bases consists of orthogonal unitary matrices which can be grouped in maximal classes of commuting matrices. As a result of this connection every MUB over $\mathbb{C}^d$ consists of at most *d* + 1 bases.

Assume that *V is* a unitary operator such as$V\left|\psi_k\right\rangle = \lambda_k\left|\psi_k\right\rangle$. Then $\left|\lambda_k\right| = 1$and for every$k = 1,\ldots,d$, we have

$\left|\left\langle\psi_k\middle|\varphi_1\right\rangle\right|$	$=$	$\left|\lambda_k^{*}\left\langle\psi_k\middle|V\middle|\varphi_1\right\rangle\right|$
	$=$	$\left|\beta_1\left\langle\psi_k\middle|\varphi_2\right\rangle\right|$
	$=$	$\left\langle\psi_k\middle|\varphi_2\right\rangle$

A similar argument shows

$$\left|\left\langle\psi_k\middle|\varphi_1\right\rangle\right| = \left|\left\langle\psi_k\middle|\varphi_2\right\rangle\right| = \ldots = \left|\left\langle\psi_k\middle|\varphi_d\right\rangle\right|.$$

Therefore,

$$\left|\left\langle\psi_k\middle|\varphi_j\right\rangle\right|^2 = \frac{1}{d}, 1\le j\le d.$$

In this case according to the *Definition* of MUB the next theorem is follows:

Theorem: *Let* $B_1 = \left\{\left|\varphi_1\right\rangle,\ldots,\left|\varphi_d\right\rangle\right\}$ *be an orthonormal basis in*$\mathbb{C}^d$. *Suppose that there is a unitary operator V such that* $V\left|\varphi_j\right\rangle = \beta_j\left|\varphi_{j+1}\right\rangle$, *where* $\left|\beta_j\right| = 1$ *and* $\left|\varphi_{d+1}\right\rangle = \left|\varphi_1\right\rangle$; *i.e., V applies a cycle shift modulo a phase on the elements of the basis* B_1. *Assume that the orthonormal basis* $B_2 = \left\{\left|\psi_1\right\rangle,\ldots,\left|\psi_d\right\rangle\right\}$ *consists of eigenvectors* *of V. Then* B_1 *and* B_2 *are MUB*

We suppose that d is a prime number, and all algebraic operations are modulo d. We consider $\{|0\rangle,|1\rangle,\dots,|d-1\rangle\}$ as the standard basis of $\mathbb{C}^d$. We define the unitary operators X_d and Z_d over $\mathbb{C}^d$, as a natural generalization of Pauli operators σ_x and σ_z as following:

$$\begin{array}{|c:c:c|}\hline X_d\,|\,j\rangle & = & |\,j+1\rangle \\ \hline X_d\,|\,j\rangle & = & \omega^j\,|\,j\rangle \\ \hline \end{array}\,. \qquad (1)$$

where ω is a d^{th} root of unity; more specifically $\omega=\exp\left\{2\pi\frac{i}{d}\right\}$. We are interested in unitary operators of the form $X_d\left(Z_d\right)^k$. In this case: $X_d\left(Z_d\right)^k|\,j\rangle=\left(\omega^k\right)^j|\,j+1\rangle$. The eigenvectors of $X_d\left(Z_d\right)^k$ are $\left|\psi_t^k\right\rangle=\frac{1}{\sqrt{d}}\sum\left(\omega^t\right)^{d-j}\left(\omega^{-k}\right)^{s_j}|\,j\rangle,\, t=0,\dots,d-1,\, s_j=j+\dots+(d-1)$.

The action of $X_d\left(Z_d\right)^\ell$ on $\left|\psi_t^k\right\rangle$ is as follows: $X_d\left(Z_d\right)^\ell\left|\psi_t^k\right\rangle=\omega^{t+k-\ell}\left|\psi_{t+k-\ell}^k\right\rangle$.

The standard basis $\{|0\rangle,|1\rangle,\dots,|d-1\rangle\}$ is the set of the eigenvectors of Z_d. It follows that the $\left|\left\langle j\middle|\psi_t^k\right\rangle\right|^2=\frac{1}{d}$. Therefore, we have proved the following construction.

Theorem: *For any prime d, the set of the bases each consisting of the eigenvectors of* $Z_d, X_d, X_dZ_d, X_d\left(Z_d\right)^2,\dots,X_d\left(Z_d\right)^{d-1}$*, form a set of d + 1 MUB*

Consider the particular cases of this theorem.

Case 1: d = 2. By Theorem, the eigenvectors of the operators σ_z,σ_x and $\sigma_x\sigma_z$ form a set of MUB; i.e., the following set

$\{\|0\rangle,\|1\rangle\}$	$\left\{\frac{1}{\sqrt{2}}\left[\|0\rangle+\|1\rangle\right],\frac{1}{\sqrt{2}}\left[\|0\rangle-\|1\rangle\right]\right\}$	$\left\{\frac{1}{\sqrt{2}}\left[\|0\rangle+i\|1\rangle\right],\frac{1}{\sqrt{2}}\left[\|0\rangle-i\|1\rangle\right]\right\}$

Case 2: d = 3. The set of the eigenvectors of the following unitary matrices form a setoff MUB

$$\begin{array}{|c|c|c|c|}\hline \begin{pmatrix}1&0&0\\0&1&0\\0&0&1\end{pmatrix} & \begin{pmatrix}0&0&1\\1&0&0\\0&1&0\end{pmatrix} & \begin{pmatrix}0&0&\omega^2\\1&0&0\\0&\omega&0\end{pmatrix} & \begin{pmatrix}0&0&\omega\\1&0&0\\0&\omega^2&0\end{pmatrix} \\ \hline \end{array}, \quad \omega=e^{\frac{2\pi}{3}i}$$

We denote the finite field $\{0,1,\dots,p-1\}$ by F_p. Let $\omega=e^{\frac{2\pi}{d}i}$ be a primitive p^{th} root of unity. Then

$Z_pX_p=\omega X_pZ_p$.

Therefore, if $U_1=\left(X_p\right)^{k_1}\left(Z_p\right)^{\ell_1}$ and $U_2=\left(X_p\right)^{k_2}\left(Z_p\right)^{\ell_2}$ then

$U_2 U_1 = \omega^{k_1 \ell_2 - k_2 \ell_1} U_1 U_2$.

Example: *Construction of sets MUB in the space* $\mathbb{C}^d$ *when d is a prime power*. When $d = p^m$, imagine the system consists of *m* subsystems each of dimension *p*. Then the total number of measurements on the whole system, viewed as performing measurement on every subsystem in their respective MUB is $(p+1)^m$. These $(p+1)^m$ operators fall into $p^m + 1$ maximal noncommuting classes where members of each class commute among themselves. The bases formed by eigenvectors of each such mutually noncommuting class are mutually unbiased. It should be mentioned that the operators in each maximal commuting class have the same structure as the stabilizers of additive error correcting codes.

Let $\alpha = (k_1, \dots, k_m)$ and $\beta = (\ell_1, \dots, \ell_m)$, then $\alpha, \beta \in \mathcal{F}_p^m$ and we denote the corresponding operator by $X_p(\alpha) Z_p(\beta)$.

Definition: *The Pauli group* $\mathcal{P}(p,m)$ *is the group of all unitary operators*

$$U = M_1 \otimes \cdots \otimes M_m, M_j = (X_p)^{k_j} (Z_p)^{\ell_j}, 0 \le k_j, \ell_j \le p-1$$

on $\mathcal{H} = \mathbb{C}^p \otimes \cdots \otimes \mathbb{C}^p$ *(tensor product of m copies of* $\mathbb{C}^p$*) of the form*

$$\omega^j X_p(\alpha) Z_p(\beta), \qquad (2)$$

for some integer $j \ge 0$ *and vectors* $\alpha, \beta \in \mathcal{F}_p^m$*, where* $\omega = \exp\left\{ \frac{2\pi}{p} i \right\}$

Let us consider the subset $\mathcal{P}_0(p,m)$ of $\mathcal{P}(p,m)$ of the operators in the form of Eq. (2) with *j* = 0.

Remark. Note that $\mathcal{P}_0(p,m)$ is not a subgroup, but generators of subgroups of the Pauli group can always be considered of $\mathcal{P}_0(p,m)$.

If the operators U and U' in $\mathcal{P}_0(p,m)$ are represented by the vectors $(k_1, \dots, k_m | \ell_1, \dots, \ell_m)$ and $(k'_1, \dots, k'_m | \ell'_1, \dots, \ell'_m)$ respectively, then U and U' are commuting iff $\sum_{j=1}^{m} k_j \ell'_j - \sum_{j=1}^{m} k'_j \ell_j = 0 \bmod p$.

An explicit formula for the action of a $\mathcal{P}_0(p,m)$ *operator* $X_p(\alpha) Z_p(\beta)$. The standard basis of the Hilbert space $\mathcal{H} = \mathbb{C}^p \otimes \cdots \otimes \mathbb{C}^p$ consists of the vectors $|j_1 \cdots j_m\rangle$, where $(j_1 \cdots j_m) \in \mathcal{F}_p^m$. Then

$$X_p(\alpha) Z_p(\beta) | j_1 \cdots j_m \rangle = \omega^{j_1 \cdot \beta_1 + \cdots + j_m \cdot \beta_m} |(j_1 + \alpha_1) \cdots (j_m + \alpha_m)\rangle.$$

Equivalently,

$X_p(\alpha) Z_p(\beta) \vert a\rangle$	=	$\omega^{a \cdot \beta} \vert a + \alpha\rangle, a \in \mathcal{F}_p^m$
$X_p(\alpha) Z_p(\beta)$	=	$\sum_{a \in \mathcal{F}_p^m} \omega^{a \cdot \beta} \vert a + \alpha\rangle\langle a \vert$

where the operations are in the field $\mathcal{F}_p$.

Example. Let $U = X_p(\alpha) Z_p(\beta)$ *and* $U' = X_p(\alpha') Z_p(\beta')$ *be operators in* $\mathcal{P}_0(p,m)$*. Let* $U \ne U'$*, i.e.,* $(\alpha, \beta) \ne (\alpha', \beta')$*. We have*

$\langle U, U' \rangle$	$=$	$Tr\left(U^{\dagger}U'\right)$
	$=$	$Tr\left(\sum_{a\in\mathbb{F}_p^m}\sum_{b\in\mathbb{F}_p^m}\omega^{\beta'\cdot b-\beta\cdot a}\lvert a\rangle\langle a+\alpha \vert b+\alpha'\rangle\langle b\rvert\right)$
	$=$	$\sum_{a\in\mathbb{F}_p^m}\omega^{\beta'\cdot b-\beta\cdot a}\langle a+\alpha \vert a+\alpha'\rangle$

While $\alpha \neq \alpha'$, then $\langle a+\alpha \vert a+\alpha' \rangle = 0$, for every $a \in \mathbb{F}_p^m$. Thus, in this case $\langle U, U' \rangle = 0$. If $\alpha = \alpha'$ and $\beta \neq \beta'$, then $\langle U, U' \rangle = \sum_{a\in\mathbb{F}_p^m} \omega^{(\beta'-\beta)\cdot a} = 0$. Thus operators $U \, and \, U'$ are orthogonal.

Theorem: *Let* $\{A_1, \ldots, A_\ell\}$ *be a set of symmetric* $m \times m$ *matrices over* $\mathbb{F}_p$ *such that* $\det\left(A_j - A_k\right) \neq 0$, *for every* $0 \le j \le k \le \ell$. *Then there is a set of* $\ell+1$ *MUB on* $\mathbb{C}^{p^m}$

Remark. More specifically, the ($\ell+1$)-bases of the above theorem are represented by the matrices

$$\left(0_m \vert 1_m\right), \left(1_m \vert A_1\right), \ldots, \left(1_m \vert A_\ell\right).$$

<u>*Example*</u>: *d* = 4. The four matrices (over $\mathbb{F}_2 = \{0,1\}$), which satisfy conditions of above theorem, are

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

.

Therefore the classes of maximal commuting operators are

$\mathcal{L}_0$	$=$	$\{Z\otimes I, I\otimes Z, Z\otimes Z\}$
$\mathcal{L}_1$	$=$	$\{X\otimes I, I\otimes X, X\otimes X\}$
$\mathcal{L}_2$	$=$	$\{Y\otimes I, I\otimes Y, Y\otimes Y\}$
$\mathcal{L}_3$	$=$	$\{X\otimes Z, Z\otimes Y, Y\otimes X\}$
$\mathcal{L}_4$	$=$	$\{Y\otimes Z, Z\otimes X, X\otimes Y\}$

where

$I = \begin{pmatrix}1&0\\0&1\end{pmatrix}$	$X = \begin{pmatrix}0&1\\1&0\end{pmatrix}$	$Y = \begin{pmatrix}0&-1\\1&0\end{pmatrix} = XZ$	$Z = \begin{pmatrix}1&0\\0&-1\end{pmatrix}$

We represent this basis explicitly. To this end, we naturally represent each basis by a 4×4 matrix such that the j^{th} row of this matrix is the components of the j^{th} vector of the corresponding basis with respect to the standard basis $\{\lvert 00\rangle, \lvert 01\rangle, \lvert 10\rangle, \lvert 11\rangle\}$: the first matrix is $B_0 = 1_4$, and

$B_1 = \frac{1}{2}\begin{pmatrix}1&1&1&1\\1&-1&-1&1\\1&1&-1&-1\\1&-1&1&-1\end{pmatrix}$	$B_2 = \frac{1}{2}\begin{pmatrix}1&i&i&-1\\1&-i&-i&-1\\1&i&-i&1\\1&-i&i&1\end{pmatrix}$
$B_3 = \frac{1}{2}\begin{pmatrix}1&1&-i&i\\1&-1&i&i\\1&1&i&-i\\1&-1&-i&-i\end{pmatrix}$	$B_4 = \frac{1}{2}\begin{pmatrix}1&-i&1&i\\1&i&-1&i\\1&i&1&-i\\1&-i&-1&-i\end{pmatrix}$

Remark. In this case, the mutually unbiasedness condition is equivalent to the condition that $B_i B_i^\dagger = 1_4$, for every $0 \le i \le 4$, and each entry of $B_i B_j^\dagger$, for $0 \le i \le j \le 4$, has absolute value equal to $\frac{1}{2}$.

Example: *The intermediate states in mutually unbiased bases*. Let us consider the two mutually unbiased bases A and A' that contain *N* basis states instead of two. We define the *intermediate states* between these bases. The basis *A* is chosen as computational basis, $|a_0\rangle, \ldots, |a_{N-1}\rangle$ and the second basis A', is the Fourier transform of the computational basis:

$$|a'_k\rangle = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \exp\left\{2\pi ik\frac{n}{N}\right\} |a_n\rangle .$$

These two bases are mutually unbiased, i.e.

$$\langle a_n | a'_k \rangle = \frac{1}{\sqrt{N}} \exp\left\{2\pi ik\frac{n}{N}\right\}.$$

This means that the distance between pairs of state from two bases is

$$\cos(\theta) = \frac{1}{\sqrt{N}} .$$

Remark. Having two states, it is possible to define a state which lies exactly in between the two, which means that it has same overlap with both states and it is *the state closest to the two original states which has this property*. The intermediate state is obtained by forming pairs of the states from two bases. They are shown in the Table 1 below.

Table 1: Possible pairs of the states from the two bases

Pair	a'_0	a'_2	$\cdots$	a'_{N-1}
a_0	m_{00}	m_{01}	$\vdots$	$m_{0,N-1}$
a_2	m_{10}	m_{11}	$\vdots$	$m_{1,N-1}$
$\vdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
a_{N-1}	$m_{N-1,0}$	$m_{N-1,1}$	$\vdots$	$m_{N-1,N-1}$

Explicitly the intermediate state between $|a_n\rangle$ and $|a'_k\rangle$ is defined in the following way

$$|m_{nk}\rangle = \frac{1}{\sqrt{C}}\left[\exp\left\{2\pi ik\frac{n}{N}\right\}|a_n\rangle + |a'_k\rangle\right],$$

where $C = 2\left(1 + \frac{1}{\sqrt{N}}\right)$ is the normalization constant and the phase comes from the overlap between $|a_n\rangle$ and $|a'_k\rangle$. The index of the *m*-states is such that the first index always refers to the *A* and the second to the $A'-$basis. Since each basis contains *N* state it is possible to form N^2 intermediate states, simply by forming all pairs of states from the two bases.

In general the intermediate state $|m_{\alpha\beta}\rangle$ between two arbitrary initial states $|\alpha\rangle$ and $|\beta\rangle$ is defined as

$$|m_{\alpha\beta}\rangle = \frac{1}{\sqrt{2}} \cdot \left\{ \frac{1}{\sqrt{|\langle\alpha|\beta\rangle| + |\langle\alpha|\beta\rangle|^2}} \left[\sqrt{\langle\alpha|\beta\rangle}\,|\alpha\rangle + \sqrt{\langle\beta|\alpha\rangle}\,|\beta\rangle\right] \right\}.$$

Remark. The intermediate states may be defined in completely generality for arbitrary initial states and any number of them. In this case the intermediate state is found by forming the mixture of all initial states with equal weight, the eigenstate with the large eigenvalue of this mixture corresponds to the intermediate state. Naturally these definitions are equivalent and lead to the same intermediate state.

Example: *Probabilistic interpretation of the intermediate state*. Considering the intermediate states leads to the following conditional probabilities

$$p\left(m_{nk}\left|a_n\right.\right)=p\left(m_{nk}\left|a'_n\right.\right)=\frac{1}{2}\left(1+\frac{1}{\sqrt{N}}\right)\equiv F\ .$$

Notice that this definition indeed recover the formula for cosine of half the angle:

$$\cos\left(\frac{\theta}{2}\right)=\sqrt{\frac{1+\cos\theta}{2}}\ .$$

This is why the states have been named intermediate states, since they indeed lie in between the two original states.

Whereas the probability for making an error is

$$p\left(m_{nk}\left|a_q\right.\right)=p\left(m_{nk}\left|a'_p\right.\right)=\frac{1}{2}\left(\frac{1-\frac{1}{\sqrt{N}}}{N-1}\right)\equiv\frac{D}{N-1}\,.$$

It is important to notice that the intermediate states in general not are orthogonal, indeed

$$\left\langle m_{kl}\left|m_{nm}\right.\right\rangle=\frac{1}{C\sqrt{N}}\left[\sqrt{N}\delta_{kn}\exp\left\{\frac{2\pi i}{N}\left(mn-lk\right)\right\}+\sqrt{N}\delta_{kn}+\exp\left\{\frac{2\pi i}{N}\left(m-l\right)k\right\}+\exp\left\{\frac{2\pi i}{N}\left(m-l\right)n\right\}\right]$$

This means that the generalized intermediate states do in general not form bases as in the two dimensional case. But they can still be used as binary measurements.

Example: *Information contents of the intermediate states as binary measurements*. It has been shown above that in general the intermediate states $\left|m_{kl}\right\rangle$ are not orthogonal, and hence they do not form bases as in the two dimensional case. It is however possible to use the corresponding projectors, $\left|m_{kl}\right\rangle\left\langle m_{kl}\right|$ as binary measurements. Since the intermediate states are non-orthogonal, it means the corresponding binary measurements are mutually incompatible. In other words, none of them can be measured together but they have to be measured one by one. A binary measurement has as the name indicates two possible outcomes, 0 and 1. Where the zero outcome is interpreted as "*I guess the state was not*$\left|m_{kl}\right\rangle$", and the "1" outcome is interpreted as "*I guess the state was*$\left|m_{kl}\right\rangle$". However, the answers are statistical, in the sense that there is a certain probability for making the wrong *identification*.

It should be mentioned that the N^2 intermediate states constitutes a generalized measurement namely so called POVM. We have $\frac{1}{N}\sum_{n,k=0}^{N-1}\left|m_{kl}\right\rangle\left\langle m_{kl}\right|=1$.

However, we do not make use of this in what follows. The probability of making the correct *identification* is given by value of *F* and is equal to $\frac{1}{2}+\frac{1}{2\sqrt{N}}$. Whereas the probability of wrong identification, i.e., of an error is given by $p\left(m_{nk}\left|a_q\right.\right)\equiv\frac{D}{N-1}$ and is equal to $\frac{1}{N-1}\left(\frac{1}{2}-\frac{1}{2\sqrt{N}}\right)$. This means that the Shannon information amount obtained by measurement is given by

$$I_{inter}^{N}=\log_2\left(N\right)+\left(\frac{1}{2}+\frac{1}{2\sqrt{N}}\right)\log_2\left(\frac{1}{2}+\frac{1}{2\sqrt{N}}\right)+\left(\frac{1}{2}-\frac{1}{2\sqrt{N}}\right)\log_2\left[\frac{1}{N-1}\left(\frac{1}{2}-\frac{1}{2\sqrt{N}}\right)\right]$$

on the "1" outcomes of intermediate state measurements.

Example: *Information capacity of generalized (non-orthogonal) quantum measurements and optimal detection of quantum information*. Let us to use the standard measure of information developed by Shannon, the standard definition of quantum relative entropy, and von Neumann definition of quantum entropy. Consider the particular case when two quantum systems are identically prepared in different location. It is well known that composite quantum system, consisting of non-interacting parts, can possess non-local properties. In particular, a composite system can exhibit correlations, which cannot be reproduced by any theoretical model that involves only variables belonging to each subsystem separately. A typical example is a pair of $\text{spin}-\frac{1}{2}$ particles produced by the decay of spin-less object. Their combined state, $\frac{1}{\sqrt{2}}\left(\left|0_1 1_2\right\rangle-\left|1_1 0_2\right\rangle\right)$, cannot be reduced to a direct product by any transformation of the bases pertaining to each one of the particles (see in details Appendix 3).

Remark. We consider in this example a different kind of composite system. Its parts never interacted in past. They may have been prepared in different laboratories. However, they were prepared according to the same set of instructions. Therefore, these subsystems are in the same quantum state – insofar as their internal variables are considered. For example, we may have two non-interacting $\text{spin}-\frac{1}{2}$ particles, prepared with the same polarization.

Let us consider the particular case in which there are three possible states for the two particles: Both spins may be directed along the *z* direction, or both may be in the $x-z$ plane, titled at $\frac{2\pi}{3}=120^o$ or -120^o from the *z* axis. (We have chosen this particular setup because for orthonormal projectors $P=\left(P_1,P_2,P_3\right)$ on these three directions $\frac{2}{3}\left(P_1+P_2+P_3\right)=I$ and the three possible states of each particle satisfy $\left\langle\psi_1\middle|\psi_2\right\rangle\left\langle\psi_2\middle|\psi_3\right\rangle\left\langle\psi_3\middle|\psi_1\right\rangle=-\frac{1}{8}$, and this is the most negative value obtainable for such a triple product). Suppose now that an outside observer wants to determine which one of these three known preparations was actually implemented. The answer cannot be unambiguous, because the three states are not orthogonal. The observer may nevertheless assign probabilities to the various preparations. The problem thus is to design a measurement procedure, which minimizes the unavoidable uncertainty of the result. We will to determine whether more information could be means of an apparatus interacting with both particles together, then by separate measurements performed on each one of them individually. An example of a measurement of the former type – which we will call a combined measurement – is the one represented by the "entangled" operator. The results are intriguing: New measurement technique, acting on each particle separately, yields more information than any separate – particle method.

Remark. In a von Neumann measurement model, the various outcomes are associated with orthogonal projection operators P_i satisfying $\sum P_i=1$ (where 1 is the unit operator), and the probability on the *i*-th outcome is given by $\left\langle\psi\middle|P_i\middle|\psi\right\rangle$. However, instead of using a set of orthogonal operators, it may be preferable – to associate the final outcome s with a more general set of non –commuting positive operators A_i, satisfying $\sum A_i=1$. The probability of getting the *i*-th is $\left\langle\psi\middle|A_i\middle|\psi\right\rangle$, so that this set of A_i forms a probability-operator measure (POM). According to Neimark's theorem such a POM is physically realizable: one can extend the Hilbert space $\mathcal{H}$ of quantum states, in which the POM is defined, in such way that there exists, in the extended space $\mathcal{K}$, a set of orthogonal projection operators satisfying $\sum P_i=1$, and such that $A_i=\Pi P_i\Pi$, where Π is the projection operator from $\mathcal{K}$ to $\mathcal{H}$. The method includes the generalized measurement that mathematically is different from simple measurement, while described by non-orthogonal expanding of unity. The measurement on any physical system can be interpreted as the simple measurement on any its enlarged section. These measurements can extract more information than simple measurements.

Let us consider two-dimensional Euclid space $\mathcal{H}$ and fixed in this space three unit vectors $(e_1, e_2, e_3) = \vec{e}$ with equal angle $\frac{2\pi}{3}$ between them. We can define the states $E_\alpha(x) = (e_\alpha, xe_\alpha)$, $\alpha = 1, 2, 3.$ Let the probability density functions in these directions are $\pi_1 = \pi_2 = \pi_3 = \frac{1}{3}$ and define the information amount $I_{E_\alpha}(x)$ in extracted states E_α by an *a priori* probability density functions $\{\pi_i\}$. Mathematically every simple measurement is *orthogonal resolution of identity* in $\mathcal{H}$ with the following properties: (*i*) $x_\omega \geq 0, \omega \in \Omega;$ (*ii*) $\sum\limits_\omega x_\omega = I;$ (*iii*) $x_\omega x_\delta = 0,\ for\ x \neq \delta$. From (*iii*) we are received that for every ω an Hermitian operator x_ω is a projector: $x_\omega^2 = x_\omega$. If $\{X_i\}$ is a family of positive operators in $\mathcal{H}^{\otimes n}$, satisfying $\sum\limits_{j=1}^{N} X_j \leq I$, then defining $X_0 = I - \sum\limits_{j=1}^{N} X_j$ we have a resolution of identity in $\mathcal{H}^{\otimes n}$.

Let on the physical system $\{\mathcal{H}, \mathcal{A}\}$ in any state *E* is produced with probability *p* the simple $\Omega-$ measurement $\{x_\omega\}$, and with the probability $(1-p)$ is produced another simple $\Omega-$ measurement $\{y_\omega\}$. We can introduce the generalized $\Omega-$ measurement as

$$z_\omega = px_\omega + (1-p)y_\omega,$$

and the resulting probability density function can be described as following

$$p_z(\omega) = pE(x_\omega) + (1-p)E(y_\omega).$$

Thus, the generalized $\Omega-$ measurement $z = \{z_\omega\}$ is equal to a randomized experiment with the simple measurements. According to *Neimark*'s theorem the orthogonal resolution $\{\tilde{z}_\omega\}$ exists. Let $\tilde{\mathcal{H}} = \mathcal{H} \otimes \mathcal{H}$ and the spaces $\mathcal{H}$ is embedded in $\tilde{\mathcal{H}}$ as $\mathcal{H} \oplus \{0\}$. The relation

$$\tilde{z}_\omega = \left(\begin{array}{c:c} px_\omega + (1-p)y_\omega & \sqrt{p(1-p)}(x_\omega - y_\omega) \\ \hdashline \sqrt{p(1-p)}(x_\omega - y_\omega) & I - (px_\omega + (1-p)y_\omega) \end{array} \right)$$

is the quantity orthogonal resolution of identity in $\tilde{\mathcal{H}}$.

In this case measurement $x_\omega = \frac{2}{3}P_\omega, \omega = 1, 2, 3,$ is not a simple measurement but is an extremal point of the measurement space: If $x_\omega = \frac{1}{2}y_\omega + \frac{1}{2}z_\omega$, then $y_\omega \leq 2x_\omega = \frac{4}{3}P_\omega;$ while P_ω is one-dimensional projector than $y_\omega = c_\omega P_\omega; c_\omega \geq 0$.

The equation $c_1P_1 + c_2P_2 + c_3P_3 = I$ have only one solution $c_1 = c_2 = c_3 = \frac{2}{3}$. It is means that in the set of generalized $\Omega-$ measurements exists the measurements that are the extremal points of all measurements but are not simple, i.e., non-orthogonal.

For this case we have two possibilities: (1) the trivial expanding of unity as $I + 0 + 0 + \ldots = I$; and (2) in two dimensional space $\mathcal{H}$ possible only the expanding of unit as $P_1 + P_2 = I$, where P_1, P_2 are one dimensional projectors on mutually orthogonal vectors h_1, h_2. Let us define the angle between h_1 and e_1 as α. For this case we can define the transition probabilities as following

$\frac{\alpha}{\omega}$	1	2	3
1	$\cos^2\alpha$	$\cos^2\left(\alpha-\frac{2\pi}{3}\right)$	$\cos^2\left(\alpha+\frac{2\pi}{3}\right)$
2	$\sin^2\alpha$	$\sin^2\left(\alpha-\frac{2\pi}{3}\right)$	$\sin^2\left(\alpha+\frac{2\pi}{3}\right)$

While $\cos^2\alpha+\cos^2\left(\alpha-\frac{2\pi}{3}\right)+\cos^2\left(\alpha+\frac{2\pi}{3}\right)=\frac{3}{2}$, than $\sum_\alpha \pi_\alpha p_{\alpha\omega}=\frac{1}{2}$ for $\omega=1,2$. Thus, the amount of information in extracted measurement can be defined as follows

$$I_{E_\alpha}(\alpha)=\ln 2-\frac{1}{3}\left[H(\alpha)+H\left(\frac{\pi}{6}-\alpha\right)\right],$$

where

$$H(\alpha)=-\left[\cos^2\alpha\ln\cos^2\alpha+\cos^2\left(\alpha-\frac{2\pi}{3}\right)\ln\cos^2\left(\alpha-\frac{2\pi}{3}\right)+\cos^2\left(\alpha+\frac{2\pi}{3}\right)\ln\cos^2\left(\alpha+\frac{2\pi}{3}\right)\right].$$

The function $H(\alpha)$ has the following property: $H(\alpha)=H\left(\alpha+\frac{\pi}{3}\right)=H\left(\alpha-\frac{\pi}{3}\right)$. From this property we can bound the value of α as $0\le\alpha\le\frac{\pi}{6}$ and

$$I_{E_\alpha}(\alpha)=\ln 2-\frac{1}{3}\min_{0\le\alpha\le\frac{\pi}{6}}\left[H(\alpha)+H\left(\frac{\pi}{6}-\alpha\right)\right]. \qquad (2)$$

Let us now estimate the value $\mathcal{J}=\sup_{x\in\mathcal{O}} I_{E_\alpha}(x)$ as a maximum amount of information that we can extract from the measurements. Let P_1,P_2,P_3 are orthogonal projectors on any three directions that have equal $\frac{2\pi}{3}$ angles between them. In this case $x_\omega=\frac{2}{3}P_\omega,\ \omega=1,2,3,$ is a measurement. The angle between a vector e_1 and a direction, that corresponds to P_1 define as α , and $I_\omega(\alpha)$ is the corresponding information amount. Let us defined Ω is the finite set, $\mathcal{O}(\Omega)$ is the set of generalized $\Omega-$ measurements, and $\hat{\mathcal{O}}(\Omega)$ is the set of simple $\Omega-$ measurements on the system $\{\mathcal{H},\mathcal{A}\}$. We are received the following result:

$$\hat{I}(\alpha)=\sup_{x\in\hat{\mathcal{O}}} I_\omega(\alpha)<I_3=\max_{0\le\alpha\le\frac{\pi}{6}} I_\omega(\alpha), \qquad (3)$$

where $I(x)$ is an average information amount about the system state on one experiment and defined as *Holevo-Levitin-Gordon* amount of information

$$I(x)=H_x\left(\sum_\beta \pi_\beta E_\beta\right)-\sum_\beta \pi_\beta H_x(E_\beta),$$

and $H_x(E)=-\sum_\omega E(x_\omega)\ln E(x_\omega)$ is a Shannon entropy of probability density for measurement *x* in state *E*. The greatest information amount with these measurements is $\hat{I}=\sup_{x\in\hat{\mathcal{O}}} I_\omega(x)$, where $\mathcal{O}$ is the set of the all generalized measurements on the system $\{\mathcal{H},\mathcal{A}\}$.

In classical case: $\hat{I}(x)=I_\omega(x)$.

For non-classical systems this equality can be not carried out and in general case we have Eq.(3), $\hat{I} < I$.

The table of transition probabilities $p_{\alpha\omega} = E_\alpha\left(x_\omega\right)$ is described as following

$\frac{\alpha}{\omega}$	1	2	3
1	$\frac{2}{3}\cos^2\alpha$	$\frac{2}{3}\cos^2\left(\alpha-\frac{2\pi}{3}\right)$	$\frac{2}{3}\cos^2\left(\alpha+\frac{2\pi}{3}\right)$
2	$\frac{2}{3}\cos^2\left(\alpha+\frac{2\pi}{3}\right)$	$\frac{2}{3}\cos^2\alpha$	$\frac{2}{3}\cos^2\left(\alpha-\frac{2\pi}{3}\right)$
3	$\frac{2}{3}\cos^2\left(\alpha-\frac{2\pi}{3}\right)$	$\frac{2}{3}\cos^2\left(\alpha+\frac{2\pi}{3}\right)$	$\frac{2}{3}\cos^2\alpha$

According to the relation as $\cos^2\alpha+\cos^2\left(\alpha-\frac{2\pi}{3}\right)+\cos^2\left(\alpha+\frac{2\pi}{3}\right)=\frac{3}{2}$ we have $\sum_\alpha \pi_\alpha p_{\alpha\omega}=\frac{1}{3}$, and $I\left(\alpha\right)=\ln 3+\left(\frac{2}{3}\cos^2\alpha\ln\frac{2}{3}\cos^2\alpha+\ldots\right)$; after algebraic transformations we have

$$I_3\left(\alpha\right)=\ln 2-\frac{1}{3}\min_{0\le\alpha\le\frac{\pi}{6}} 2H\left(\alpha\right). \tag{4}$$

Comparison of Eq.(2) and Eq.(4) give us $\hat{I} < I$.

Circuit complexity measures

There are two important measures to quantify the «costs» of a quantum program: first, the total number of quantum gates concerning a given elementary gate set and, second, the number of qubits needed to implement the circuit. To find the optimal quantum circuit both parameters have to be optimized which might be a conflicting objective. However, suppose a sufficient number of qubits is available. Then, the efficiency can also measured by the time passing from the initialization of the system state to the final measurement, which corresponds to the number of computational steps. Here, a computational step means the application of (maximally) parallel quantum gates. Fig. 13 gives an example.

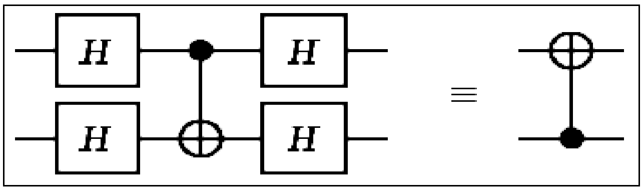

Figure 13. Quantum circuit of elementary quantum gates. The left quantum circuit consists of five elementary quantum gates

Since two Hadamard gates before and after *CNOT* each can be applied parallel, quantum circuit consists of three computational steps. However, the right quantum circuit is equivalent to the left circuit but consists only of a single *CNOT* gate.

If choosing the number of quantum gates as a measure, the simple uniform valuation model, i.e., each gate contributes the same costs, may be replaced by a more «realistic» measure, depending on a certain physical realization, which e.g. weights the costs according to the gate type. Independent of the physical implementation it might make sense, for instance, to rate controlled gates higher than single qubit gates.

Comparison between quantum algorithms which use oracle gates, so-called quantum black-box algorithms, and their corresponding classical algorithms are based on the number of accesses to input data, that is, the number of oracle calls, instead of the number of computational steps, or the number of elementary gates respectively. This complexity measure is also denoted as *query complexity* or *decision tree complexity*. Of course, a comparison of quantum and classical algorithms on the basis of oracle calls is only reasonable, of both algorithms use the same oracle.

Behind this is another computation model, the decision-tree or query model.

Strictly speaking, in the quantum version of this model (deterministic and randomized decision-trees are two other kinds) quantum circuits are studied which can be described by a unitary transformation $A = U_T \cdot O \cdot U_{T-2} \cdots O \cdot U_1 \cdot O \cdot U_0$, where the U_i denote fixed, input-independent unitary transformations and the gate O an oracle call (input-dependent unitary transformation). Relevant from the viewpoint of complexity is the number of queries T .

Why investigating such a restricted model?

As it is not possible to make decisive complexity theoretical statements in more powerful computation models (this would comprise the answer to the question, whether quantum computing is more powerful than classical computing), simpler and more limited models of computation are analyzed. The hope is that understanding of such easier models will lead to a better understanding of the more complex models.

Quantum computational complexity

Like other promising non-standard computing approaches, quantum computing raised the hope that NP-complete problems which seem to be intractable for classical computers could be solved efficiently. Due to the merely quadratic speedup and the optimality of Grover’s quantum search algorithm, it is obvious that approaches primarily based upon (unstructured) quantum search cannot yield efficient solutions to the problems in NP, is particular to the NP-complete problems. This can be considered to be an indication that the class of NP-complete problems cannot be solved efficiently on a quantum computer. However, up to now this is neither proven nor disproven.

Complexity measures for quantum computational circuits

Apart from NP-complete problems, there are some problems, which seem to be intermediate in difficulty between P and NPC problems. This class of (decision) problems is denoted NPI (NP-incomplete) and it is NPI: = NP-NPC-P. Lander has shown that NPI is not empty, iff $NP \neq P$. Thus, funding a problem in NPI would solve the most important and famous problem is computer science. For instance, the decision problem of factoring is regarded as a candidate for NPI: Given a composite integer m and $l < m$, decide whether m has a non-trivial factor less than l. Another problem which is still believed to be in NPI is graph isomorphism. Such problems appear to be hard classically, but they can perhaps be solved efficiently on a quantum computer, as it was shown for factoring.

Another example for a classically hard problem, but nor proven to be in NP-complete, which perhaps can be efficiently solved on quantum computers is the *shortest lattice vector problem* (SVP). The shortest lattice vector problem consists in finding the shortest non-trivial vector of a lattice L generated by d linear independent vectors in the vector space $\mathbb{Q}^d$. Like factoring, the difficulty of this problem ensures security in public key encryption systems (by using the inverse of this problem as a one-way function). However, the provability of Micciancio’s number theoretical conjecture would lead to $SVP \in NPC$, and this would argue against an efficient quantum solution.

From these short and unfinished reflections, it becomes convincing that quantum computing needs a separate complexity theory to understand the potentials and limitations of quantum computing compares to classical computing.

Different complexity measures for quantum circuits were already explained at the end of the last section. Another computation model is the *Quantum Turing Machine* (QTM) which is not explained here. While quantum circuits (like Boolean circuits) are a non-uniform computation model, because they have only constant input length, QTMs (like classical TMs) are uniform, as they work on arbitrary input lengths. As in classical complexity theory, the uniform quantum complexity of computational problems is of particular interest. Therefore, analogous to uniform Boolean circuit families, *uniform quantum circuit families* are considered: A uniform quantum circuit family is an infinite sequence of circuit C_n for each input length n such that C_n can be generated by a QTM on input n (in polynomial time $O\left(poly\left(c\left(n\right)\right)\right)$ where $c\left(n\right)$ is the size of quantum circuit C_n based on a given elementary gate set. They allow a comparison of the computational power of the more abstract QTMs and the more practical (uniform) quantum circuits: It can be shown that QTMs with polynomial runtime can be simulated by uniform quantum circuits of polynomial size (with bounded error probability) and vice versa.

The following paragraph summarizes some results about quantum complexity classes (defined over QTMs) and their relation to classical complexity classes. Bernstein and Vazirani introduce quantum analogous to the classical complexity classes *P* and *BPP EQP* corresponds to *P* and denotes the class of problems solved error-free on a QTM in polynomial time.

Remark. BPP (probabilistic polynomial with bounded error) is the classical complexity class of decision problems that can be solved on polynomial time on a probabilistic Turing machine with bounded error probability.

BQP, the quantum version of BPP, denotes the complexity class of all computational decision problems that can be solved with bounded error probability on a QTM in polynomial time.

How EQP and BQP exactly relate to the classical complexity classes P, NP, PP, BPP and PSPACE is unknown. What is known is that

$$BPP \subseteq BQP \subseteq PP \subseteq PSPACE \text{ and } P \subseteq EQP \subseteq BQP$$

Remark. PP is the classical complexity class of problems solved by randomized algorithms with unbounded error probability. PSPACE is the classical complexity class of all decision problems which can be solved on a deterministic Turing machine using polynomial space and arbitrary time.

It is still an open problem to determine which of the inclusions proper inclusions are and which are not. Especially it is not proven that $BQP \neq BPP$, that is, that quantum computers have capabilities beyond those of classical commuters, although there is strong evidence suggesting this.

The existing QAs described below can be naturally expressed using a black-box model. It is then useful to consider the spatiotemporal complexity of QAs from the quantum query complexity viewpoint. For example, in the case of Simon's problem, one is given a function $f:\{0,1\}^n \to \{0,1\}^n$ and a promise that there is an $s \in \{0,1\}^n$ such that $f\left(i\right) = f\left(j\right)$ iff $i = j \oplus s$.

The goal is to determine whether $s = 0$ or not. Simon's QA yields an exponential speed-up over a classical algorithm. Simon's QA requires an expected number of $O\left(n\right)$ applications of f, whereas, every classical randomized algorithm for the same problem must make $\Omega\left(\sqrt{2^n}\right)$ queries[1]. The function f can be viewed as a black-box $X = \left(x_0,\ldots,x_{N-1}\right)$ of $N = 2^n$ bits, and that an f-application can be simulated by n queries to X.

Thus, Simon's problem fits squarely in the black-box setting, and exhibits an exponential quantum-classical separation for this promise-problem. The promise means that Simon's problem

[1] The readers unfamiliar with asymptotic notation, $O\left(f\left(N\right)\right)$ means «at most order $f\left(N\right)$», $\Omega\left(f\left(N\right)\right)$ means «at least order $f\left(N\right)$», and $\theta\left(f\left(N\right)\right)$ means «exactly order $f\left(N\right)$».

$f:\{0,1\}^n \to \{0,1\}^n$ is partial; i.e., it is not defined on all $X \in \{0,1\}^n$ but only on X that correspond to an X satisfying the promise.

Table 2 list the quantum complexity of various Boolean functions such as OR, AND, PARITY, and MAJORITY.

Table 2. Some quantum complexities

Function	Exact	Zero-error	Bounded-error
OR_N, AND_N	N	N	$\Theta\left(\sqrt{N}\right)$
$PARITY_N$	$\frac{N}{2}$	$\frac{N}{2}$	$\frac{N}{2}$
$MAJORITY_N$	$\Theta(N)$	$\Theta(N)$	$\Theta(N)$

For example, consider the property $OR_N(X) = x_0 \vee \ldots \vee x_{N-1}$. The number of queries required to compute $OR_N(X)$ by any classical (deterministic or randomized) algorithm is $\Theta(N)$.

The lower bound for OR implies a lower bound for the search problem where it is desired to find an i, such that $x_i = 1$, if such an i exists.

Thus, an exact or zero-error QSA requires N queries, in contrast to $\Theta\left(\sqrt{N}\right)$ queries for the bounded-error case. On the other hand, the number of solutions is r and a solution can be found with probability 1 using $O\left(\sqrt{\frac{N}{k}}\right)$ queries. Grover discovered a QSA that can be used to compute OR_N with small error probability using only $O\left(\sqrt{N}\right)$ queries. In this case of OR_N, the function is total; however, the quantum speed-up is only quadratic instead of exponential.

A similar result holds for the order-finding problem, which is the core of Shor's efficient quantum factoring algorithm. In this case, the promise is the periodicity of a certain function derived from the number to be factored.

A Boolean function is a function $f:\{0,1\}^n \to \{0,1\}$. Note that f is total, i.e., it is defined on all n-bit inputs. For an input $x \in \{0,1\}^n$, x_i to denotes its i-th bit, so $x = \{x_1 \ldots x_n\}$. The expression $|x|$ is used to denote the Hamming weight of x (its number of 1's). A more general form of a Boolean function can be defined as $f:\{0,1\}^n \supseteq A \to B = f(A) \subseteq \{0,1\}^m$, for some integers $n, m > 0$. If S is a set of (indices of) variables, then x^S denotes the input obtained by flipping the S-variables in x. The function f is symmetric if $f(x)$ only depends on $|x|$.

Some common symmetric functions are:

$$\text{(i)} OR_n(x) = 1 \;\; iff \;\; |x| \geq 1;$$

$$\text{(ii)} \;\; AND_n(x) = 1 \;\; iff \;\; |x| = n;$$

$$\text{(iii)} \;\; PARITY_n(x) = 1 \;\; iff \;\; |x| \;\; is\, odd \;;$$

$$\text{(iv)} \;\; MAJ_n(x) = 1 \;\; iff \;\; |x| > \frac{n}{2}.$$

The quantum oracle model is used to formalize a query to an input $x \in \{0,1\}^n$ as a unitary transformation O that maps $|i, b, z\rangle$ to $|i, b \oplus x_i, z\rangle$ is most some m-qubit basis state, where i takes $\lceil \log n \rceil$ bits, b is one bit. The value z denotes the $(m - \lceil \log n \rceil - 1)$-bit «workspace» of the quantum computer, which is not affected by the query. Applying the operator O_f twice is equivalent to applying the identity operator, and thus O_f is unitary (and reversible) as required. The mapping changes the content of the second register $(|b\rangle)$ conditioned on the value of the first register $|i\rangle$.

The queries are implemented using unitary transformations O_j in the following standard way. The transformation O_j only affects the leftmost part of a basis state: it maps basis state $|i,b,z\rangle$ to $|i,b\oplus x_i,z\rangle$. Note that the O_j are all equal. This generalizes the classical setting where a query inputs an i into a black-box, which returns the bit x_i. Applying O to the basis state $|i,0,z\rangle$ yields $|i,x_i,z\rangle$, from which the i-th bit of the input can be read. Because O has to be unitary, it is specified to map $|i,1,z\rangle$ to $|i,1-x_i,z\rangle$. Note that a quantum computer can make queries in superposition: applying O once to the state $\frac{1}{\sqrt{n}}\sum_{i=1}^{n}|i,0,z\rangle$ gives $\frac{1}{\sqrt{n}}\sum_{i=1}^{n}|i,x_i,z\rangle$, which in some sense contains all bits of the input.

A quantum decision tree has the following form: start with an m-qubit state $|\vec{0}\rangle$ where every bit is 0. Since it is desired to compute a function of X, which is given as a black-box, the initial state of the network is not very important and can be disregarded. Thus, the initial state is assumed to be $|\vec{0}\rangle$ always. Next, apply a unitary transformation U_0 to the state, then apply a query O, then another transformation U_1, etc. A T-query quantum decision tree thus, corresponds to a unitary transformation $A=U_T O U_{T-1}\ldots OU_1OU_0$. Here the U_i are fixed unitary transformations, independent of the input x. The final state $A|\vec{0}\rangle$ depends on the input x only via the T applications of O. The output obtained by measuring the final state and outputting the rightmost bit of the observed basis state. Without loss of generality, it can be assumed that there are no intermediate measurements.

A quantum decision tree is said to compute f exactly if the output equals $f(x)$ with probability 1, for all $x\in\{0,1\}^n$. The tree computes f with bounded-error if the output equals $f(x)$ with probability at least $\frac{2}{3}$, for all $x\in\{0,1\}^n$.

The function $Q_E(f)$ denotes the number of queries of an optimal quantum decision tree that computes f exactly, $Q_2(f)$ is the number of queries of an optimal quantum decision tree that computes f with bounded-error. Note that the number of queries is counted, not the complexity of the U_i.

Unlike the classical deterministic or randomized decision trees, the QAs are not necessarily trees anymore (the names «quantum query algorithm» or «quantum black-box algorithm» can also be used). Nevertheless, the term «quantum decision tree» is useful, because such QAs generalize classical trees in the sense that they can simulate them as described below.

Consider a T-query deterministic decision tree. It first determines which variable it will query first; then it determines the next query depending upon its history, and so on for T queries. Eventually, it outputs an output-bit depending on its total history. The basis states of the corresponding QA have the form $|i,b,h,a\rangle$, where i,b is the query-part, h ranges over all possible histories of the classical computation (this history includes all previous queries and their answers), and a is the rightmost qubit, which will eventually contain the output. Let U_0 map the initial state $|\vec{0},0,\vec{0},0\rangle$ to $|i,0,\vec{0},0\rangle$, and x_i is the first variable that classical tree would query.

Now, the QA applies O, which turns the state into $\left| i, x_i, \vec{0}, 0 \right\rangle$. Then the algorithm applies a transformation U_1 that maps $\left| i, x_i, \vec{0}, 0 \right\rangle$ to $\left| j, 0, h, 0 \right\rangle$, where h is the new history (which includes i and x_i) and x_j is the variable that the classical tree would query given the outcome of the previous query. Then when the quantum tree applies O for the second time, it applies a transformation U_2 that updates the workspace and determines the next query, etc. Finally, after T queries, the quantum tree sets the answer bit to 0 or 1 depending on its total history. All operations U_i performed here are injective mappings from basis states to basis states, hence they be extended to permutations of basis states, which are unitary transformations. Thus a T-query deterministic decision tree can be simulated by an exact a T-query quantum decision tree with the same error probability (basically because a superposition can «simulate» a probability distribution). Accordingly,

$$Q_2(f) \le R_2(f) \le D(f) \le n \text{ and } Q_2(f) \le Q_E(f) \le D(f) \le n \text{ for all } f.$$

If f is non-constant and symmetric, then

$$\text{(i) } D(f) = \left(1 - o(1)\right) n;$$

$$\text{(ii) } R_2(f) = \Theta(n);$$

$$\text{(iii) } Q_E(f) = \Theta(n);$$

$$\text{(iv) } Q_2(f) = \Theta\left(\sqrt{n\left(n - \Gamma(f)\right)}\right),$$

where $\Gamma(f) = \min\left\{\left|2k - n + 1\right| : f_k \ne f_{k+1}\right\}$ is quantity measure of length of the interval around Hamming weight $\frac{n}{2}$ where f_k is constant. The function f flips value if the Hamming weight of the input changes from k to $k+1$ (this $\Gamma(f)$ is a number that is low if f flips for inputs with Hamming weight close to $\frac{n}{2}$). This can be compared with the classical bounded-error query complexity of such functions, which is $\Theta(n)$. Thus, $\Gamma(f)$ characterizes the speed-up that QAs give for all total functions.

Unlike classical decision trees, a quantum decision tree algorithm can make queries in a quantum superposition, and therefore, may be intrinsically faster than any classical algorithm. The quantum decision tree model can also be referred to as the quantum black-box model.

Information analysis of quantum complexity of QAs: Quantum query tree complexity

Let $Q(f)$ be the quantum decision tree complexity of f with error- bounded probability by $\frac{1}{3}$. It is possible to derive a general lower bound for $Q(f)$ in terms of Shannon entropy $S^{Sh}(f)$ defined as follows. For any f, define the entropy of f, $S^{Sh}(f)$, to be the Shannon entropy of $f(X)$, where X is taken uniformly random from A:

$$S^{Sh}(f) = -\sum_{y \in B} p_y \log_2 p_y, \text{ where } p_y = \Pr_{x \in_R A}\left[f(x) = y\right].$$

For any f,

$$Q(f) = \Omega\left(\frac{S^{Sh}(f)}{\log n}\right). \tag{5}$$

In this case, the computation process can be viewed as a process of communication. To make a query, the algorithm sends the oracle $\lceil \log n \rceil$ bits, which are then returned by the oracle. The first $\lceil \log n \rceil$ bits specify the location of the input bit being queried and the remaining one bit allows the oracle to write down the answer.

The QA runs on $\frac{1}{\sqrt{|A|}}\sum_{x\in A}|x\rangle_X\,|y\rangle_Y$, where $X\left(Y\right)$ denotes the qubits that hold the input (intermediate results of computing), respectively. It is useful to now consider the von Neumann entropy, $S^{vN(t)}\left(f\right)$, of the density matrix ρ_Y after t-th query. If the QA computes f in T queries, at the end of computation, one expect to have a vector close to $\frac{1}{\sqrt{|A|}}\sum_{x\in A}|x\rangle_X\,|f\left(x\right)\rangle_Y$. For the initial (pure) state, $S^{vN(0)}\left(f\right)=0$. By using Holevo's theorem (see [2]), one can show that $S^{vN(T)}\left(f\right)\approx S^{Sh}\left(f\right)$. Furthermore, by the subadditivity of the von Neumann entropy $\left|S^{vN(t+1)}\left(f\right)-S^{vN(t)}\left(f\right)\right|=O\left(\log n\right)$ for any t with $0\le t\le T-1$. Therefore, $T=\Omega\left(\frac{S^{Sh}\left(f\right)}{\log n}\right)$. This bound is tight.

This means one quantum query can get $\log n$ bits of information, while any classical query get no more than 1 bit of information. This power of getting $O\left(1\right)$ bits of information from a query is not useful in computing total functions, which are functions that are defined on every string in $\left\{0,1\right\}^n$, in the sense that each quantum query can only yield $O\left(1\right)$ bits of information on average.

For this more general case, for any total function f ,

$$Q\left(f\right)=\Omega\left(S^{Sh}\left(f\right)\right). \tag{6}$$

Kolmogorov complexity of quantum query algorithms

In the quantum query model, as in its classical counterpart, we pay for accessing the oracle, but unlike the classical case, the machine can use the power of quantum parallelism to make queries in superposition. Access to the input $x\in\sum^n$, where $\sum$ is a finite set, is achieved by way of a query operator O_x. The query complexity of an algorithm is the number of calls to O_x. In quantum computing the state of computation is represented by a register R composed of three sub-registers: the *query register* $i\in\left\{0,1,\ldots,n\right\}$, the *answer register* $z\in\sum$ and the *work register* w. We denote a register using the ket notation $|R\rangle=|i\rangle|z\rangle|w\rangle$, or simply $|i,z,w\rangle$. In the quantum setting, the state of the computation is a complex combination of all possible values of the registers.

For the corresponding finite-dimensional vector space $\mathcal{H}$ we denote the state of the computation by a vector $|\psi\rangle\in\mathcal{H}$ over the basis $\left(|i,z,w\rangle\right)_{i,z,w}$. Furthermore, the state vectors are unit length for the ℓ_2 norm in the quantum setting. A T*-query* algorithm A is specified by a $\left(T+1\right)$-uple $\left(U_0,U_1,\ldots,U_T\right)$ of matrices. When A is quantum, the matrices U_i are unitary. The computation takes place as follows.

The query operator is the unitary matrix O_x that satisfies $O_x|i,z,w\rangle=|i,z\oplus x_i,w\rangle$ for every i,z,w, where by convention $x_0=0$. Initially the state is set to some fixed value $|0,0,0\rangle$. Then the sequence of transformations $\left(U_0,O_x,U_1,O_x,\ldots,U_{T-1},O_x,U_T\right)$ is applied.

We say that the algorithm A ε-computes a function $f : S \to S'$ for some sets $S \subseteq \sum^n$ and S', if the observation of the last bits of the work register equals $f(x)$ with probability at least $1-\varepsilon$, for every $\varepsilon \in S$. The quantum query complexity (QQC) is the minimum query complexity of quantum query algorithm that ε_0-compute f, where ε_0 is a fixed positive constant no greater than $\frac{1}{3}$.

We use a few standard results in Kolmogorov complexity and information theory. We denote the length of finite string x by $|x|$. We assume that the Turing machine's alphabet is the same finite alphabet as the alphabet used instance of the function under consideration. Letters x, y typically represent instance; i is an index into the representation of the instance; and p, q are probability distributions. Programs are denoted P, and output of a Turing machine M on input is written $M(x)$. When there are multiple inputs, we assume that a standard encoding of tuples is used. The Kolmogorov complexity of x given y with respect to M is denoted $C_M(x|y)$, and defined as follows: $C_M(x|y) = \min(|P| \textit{ such that } M(P, y) = x)$, where x, y be finite strings. A set of strings is prefix-free if no string is a prefix of another in the set. The prefix-free Kolmogorov complexity of x given y with respect to M is denoted $K_M(x|y)$, and defined as follows:

$$K_M(x|y) = \min(|P| \textit{ such that } M(P, y) = x),$$

where P is taken in some fixed prefix-free set. There exists a constant $c \geq 0$ such that for every finite string σ,

$$K(x|\sigma) \leq K(x) + c, \text{ and } K(x) \leq K(\sigma) + K(x|\sigma) + c.$$

Let A be a QA that for all $x \in S$ computes f, with bounded ε and at most T-queries to the input. Then there exists a constant C_0 such that for every $x, y \in S$ with $f(x) \neq f(y)$:

$$T \geq C_0 \times \frac{1 - 2\sqrt{\varepsilon(1-\varepsilon)}}{\sum\limits_{i,\, x_i \neq y_i} \sqrt{2^{-(K(i|x,A)+K(i|y,A))}}}.$$

Example. Grover's QSA (see [2]). Fix n and a QA A for Grover search for instance of length n. Let z be a binary string of length $\log n$, with $K(z|A) \geq \log n$. Let j be the integer between 0 and $n-1$ whose binary expansion is z. Consider x, the all 0's string, and let y be everywhere 0 except at position $i = j+1$, where it is 1. Then $K(i|x, A) \geq \log n - O(1)$, and $K(i|y, A) = O(1)$.

Therefore, $QQC(QSA) = \Omega(\sqrt{n})$.

Thus, the minimum of Shannon entropy in the final solution output of the QA means its has minimal quantum query complexity. The interrelations in Eqs (1) and (2) between quantum query complexity and Shannon entropy are used in the solution of QA-termination problem (see below in Chapter 4). As mentioned above, the number of queries is counted, not the complexity of the U_i. The complexity of a quantum operator U_i and its interrelations with the temporal complexity of a QA is considered below.

The matrix-based approach can be efficiently realized for a small number of input qubits. The matrix approach is used above as a useful tool to illustrate complexity issues associated with QA simulation on classical computer.

References

1. Gruska J. Quantum computing. – Advanced Topics in Computer Science Series, McGraw-Hill Companies, London. – 1999.
2. Nielsen M.A. and Chuang I.L. Quantum computation and quantum information. – Cambridge University Press, Cambridge, Englandю – 2000.
3. Hirvensalo M. Quantum computing. – Natural Computing Series, Springer-Verlag, Berlinю – 2001.
4. Hardy Y. and Steeb W.-H. Classical and quantum computing with C++ and Java Simulations. – Birkhauser Verlag, Basel. – 2001.
5. Hirota O. The foundation of quantum information science: Approach to quantum computer (in Japanese). – Japan. – 2002.
6. Pittenberg A.O. An introduction to quantum computing and algorithms. – Progress in Computer Sciences and Applied Logic. – Vol. 19. – Birkhauser. – 1999.
7. Brylinski F.K. and Chen G. (Eds). Mathematics of quantum computation. – Computational Mathematics Series. – CRC Press Co. – 2002.
8. Lo H.-K., Popescu S. and Spiller T. (Eds). Introduction to quantum computing and information. – World Scientific Publ. Co. – 1998.
9. Berman G.P., Doolen G.D., Mainieri R. and Tsifrinovich V.I. Introduction to quantum computers. – World Scientific Publ. Co. – 1999.
10. Rieffel E. and Polak W. An introduction to quantum computing for non-physicists // ACM Computing Surveys. – 2000. – Vol. 32. – No 3. – pp. 300 – 335.
11. Hogg T., Mochon C., Polak W. and Rieffel E. Tools for quantum algorithms // International Journal of Modern Physics. – 1999. – Vol. C10. – No 7. – pp. 1347 – 1361.
12. Uesaka Y. Mathematical principle of quantum computation (in Japanese). – Corona Publ. Co. Ltd. – 2000.
13. Marinescu D.C. and Marinescu G.M. Approaching quantum computing. – Pearson Prentice Hall, New Jersey. – 2005.
14. Benenti G., Casati G. and Strini G. Principles of quantum computation and information. –Singapore: World Scientific. – Vol. I. – 2004; – Vol. II. – 2007.
15. Nakahara M. and Ohmi T. Quantum computing: From Linear Algebra to Physical Realizations. – Taylor & Francis. – 2008.
16. Stenholm S. and Suominen K.-A. Quantum approach to informatics. – Wiley- Interscience. A J. Wiley&Sons, Inc. – 2005.
17. Jaeger G. Quantum Information: An Overview. – N.Y.: Springer Verlag. – 2007.
18. McMahon D.Quantum computing explained. – Wiley- Interscience. A J. Wiley&Sons, Inc. – 2008.