International Journal of Quantum Information
Vol. 4, No. 4 (2006) 715–737
© World Scientific Publishing Company

World Scientific
www.worldscientific.com

ALGORITHMIC COMPLEXITY OF QUANTUM STATES

CATERINA E. MORA* and HANS J. BRIEGEL*,†

*Institut für Quantenoptik und Quanteninformation der,
Österreichischen Akademie der Wissenschaften, Innsbruck, Austria

†Institut für Theoretische Physik, Universität Innsbruck,
Technikerstraße 25, A-6020 Innsbruck, Austria

Received 7 October 2005
Revised 27 January 2006

We give a definition for the Kolmogorov complexity of a pure quantum state. In classical information theory, the algorithmic complexity of a string is a measure of the information needed by a universal machine to reproduce the string itself. We define the complexity of a quantum state by means of the classical description complexity of an (abstract) experimental procedure that allows us to prepare the state with a given fidelity. We argue that our definition satisfies the intuitive idea of complexity as a measure of "how difficult" it is to prepare a state. We apply this definition to give an upper bound on the algorithmic complexity of a number of known states. Furthermore, we establish a connection between the entanglement of a quantum state and its algorithmic complexity.

Keywords: Algorithmic complexity; entanglement; Shannon entropy.

1. Introduction and Notation

Algorithmic information theory has provided a concise notion of randomness for individual objects. It has also revealed deep connections between thermodynamics and the theory of classical computation.[a] The algorithmic complexity (or randomness) of an object — usually a binary string — is thereby defined as the length in bits of the shortest program for a universal computer that reproduces the string under question.[2]

Quantum theory, on the other hand, has provided a new conceptual basis for the theory of computation. Attempts have been made to also generalize the notion of algorithmic complexity to quantum mechanical objects, described by states in a Hilbert space. These attempts are motivated primarily by the desire to formulate a comprehensive theory of quantum information. We may also expect further insights into the theory of entanglement.

In this paper, we want to characterize the algorithmic complexity of a given quantum state. Proposals have already been made by Vitanyi[3] and Berthiaume

[a]For a review, see e.g. the paper by Bennett *et al.*[1]

et al.,[4] who have introduced two possible definitions of quantum algorithmic complexity based on the reproducibility of the state via Turing machines. Gàcs[5] has instead adopted an approach based on universal probability. Our definition will be closer to the one proposed by Vitanyi, but deviating from his definition in two crucial aspects. Furthermore, we shall establish a connection between algorithmic complexity and entanglement.[6]

To introduce our definition of the algorithmic complexity of a quantum state, we shall consider the following scenario. Imagine that Alice has created a certain quantum state in her laboratory and wants to describe this state to Bob, who wants to reproduce it in his laboratory. How difficult is it for Alice to describe to Bob the state of her system? We may distinguish the two situations in which Alice and Bob communicate via a classical or a quantum channel, respectively. Depending on the choice of the communication channel, we may arrive at different notions of complexity of a quantum state.

In the first situation, Alice has to use classical information to describe her state to Bob. This appears to be a restriction, at first sight. However, we may always regard the quantum state of a system as the result of some experimental preparation procedure. The complexity of a quantum state is then associated, in a very natural way, with the (classical) description complexity of an experimental preparation procedure. The resulting notion of complexity could therefore also be called *preparation complexity*.[b]

In the second situation, Alice may use quantum information to describe her quantum state to Bob. In doing this, she has several options. She may send either the quantum state altogether to Bob, or a copy (if available), or the state in some Schumacher compressed form, or some other quantum state which Bob can transform into the desired state. If we adopt this scenario, we arrive — with Berthiaume *et al.*[4] — at a quite different notion of quantum complexity, which could also be called *encoding complexity*.

In this paper, we shall follow the first approach and identify the algorithmic complexity of a quantum state with its preparation complexity, i.e. the classical description complexity of the preparation procedure. Although the second approach looks "more quantum," it lacks an important feature that we usually associate with the *description* of an object. Even if Bob has the state sent by Alice in his hands, he might not know what state he has received. A proper description of the state, on the other hand, will allow him to reproduce the state himself.

To be able to communicate, Alice and Bob must first have agreed on a common language which they are using to describe their preparation procedure. Ideally, they will use the same "toolbox" to compose their experiments and the same words when referring to elements of this toolbox. In quantum information theory, we abstract from a particular physical system in which a quantum state is realized.

[b]This approach is close to Bohr's viewpoint that the quantum state is essentially an expression of an experimental scenario.[7]

The experimental toolbox is thereby replaced by a set of elementary operations on a Hilbert space with a given tensor-product structure and dimension.[c] The toolbox in quantum information theory is thus a gross abstraction from an experimental scenario. Here, the toolbox will include the possibility to prepare some standard reference state, and a finite set of elementary unitary transformations. A complete preparation procedure is then described as a sequence of unitary transformations (a quantum circuit) applied to the reference state.

Considering that a quantum state can be characterized by a circuit with which the state can be prepared, we want to define the complexity of a state referring to that of the circuit itself. It is known that a finite set of gates (constituting a complete basis) is suitable for preparing any state (up to an arbitrary precision); through an appropriate coding, thus, the circuit itself can be reduced to a (classical) string whose Kolmogorov complexity is well defined and which can be associated to the original state. In this way, the algorithmic complexity of a state satisfies the intuitive idea of complexity as a measure of "how difficult" it is to prepare a state.

From now on, we will represent with $\mathcal{Q}_N$ the space generated by N qubits and with $|0\rangle$ the vector $\mathcal{Q}_N \ni |0\rangle = |0\rangle_N = |0\rangle_{(1)}|0\rangle_{(2)} \cdots |0\rangle_{(N)}$ (where $|0\rangle_{(i)} \in \mathcal{Q}_1^{(i)}$ is an element of the computational basis $\{|0\rangle_1, |1\rangle_1\}$ of $\mathcal{Q}_1^{(i)}$).

We will represent with $\mathcal{C}|0\rangle$ the result of the application of a circuit $\mathcal{C}$ on the vector $|0\rangle$; if $|\langle\varphi|\mathcal{C}|0\rangle|^2 \geq 1-\varepsilon$ we will say that $\mathcal{C}$ prepares $|\varphi\rangle$ with precision ε (where $0 \leq \varepsilon \leq 1$). We will say that two states $|\varphi\rangle$ and $|\psi\rangle$ are *ε-distinguishable* if $|\langle\psi|\varphi\rangle|^2 \leq 1-\varepsilon$; a circuit that prepares $|\varphi\rangle$ with precision ε thus prepares any state $|\psi\rangle$ that is not ε-distinguishable from $|\varphi\rangle$. When saying that $\mathcal{C}$ prepares $|\varphi\rangle$ we mean that $\mathcal{C}|0\rangle = |\varphi\rangle$.

In particular, we will be interested in building quantum circuits from a fixed set of gates: as we want to be able to reproduce any state (at least up to a given precision) this set must constitute a *complete gate basis*.

Example 1 [Standard basis]. An example of a complete gate basis is the so-called standard basis[8] $\mathcal{B} = \{H, T, C\}$, where H is the Hadamard gate, T a $\frac{\pi}{8}$-gate, and C the controlled NOT:

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}; \quad T = \begin{pmatrix} \mathrm{e}^{-i\pi/8} & 0 \\ 0 & \mathrm{e}^{i\pi/8} \end{pmatrix}; \quad \begin{aligned} C\ |0\rangle|\varphi\rangle &= |0\rangle|\varphi\rangle \\ C\ |1\rangle|0\rangle &= |1\rangle|1\rangle\ . \\ C\ |1\rangle|1\rangle &= |1\rangle|0\rangle \end{aligned}$$

T represents a $\pi/4$ rotation about the z-axis, while HTH is a $\pi/4$ rotation around the x-axis.

Given a fixed (finite) number of gates, only a countable set of states can be prepared exactly. If we consider a complete (finite) gate basis, it is possible, however, to reproduce any unitary transformation $\mathcal{U}$ (and thus any state $|\varphi\rangle$) up to an

[c]We would expect that any notion of complexity should be asymptotically invariant under coarsening of the description of the toolbox (for discussion of this point, see Sec. 3).

arbitrary precision. Considering that the definition of the complexity of a quantum state will be based on its preparation by means of a quantum circuit, it is therefore necessary to introduce a *precision parameter* in such a definition. We will nevertheless start by defining the algorithmic complexity on the set of states that can be exactly prepared with circuits built from a fixed basis. In this case, it is obviously not necessary to introduce this parameter; it will appear only when generalizing this definition to arbitrary states.

2. Classical Kolmogorov Complexity

The definition of algorithmic complexity proposed by Kolmogorov[2,9] is meant to give an answer to the question: "Is a (classical) sequence random?"

Algorithmic complexity gives a definition of randomness very close to the intuitive idea of "structureless" and is based on the concept of *algorithmic reproducibility* of a sequence. In mathematical terms, the (classical) Kolmogorov complexity K_{Cl} of a (binary) string $\boldsymbol{\omega}$ is defined as *the length of the shortest program that, running on a universal Turing machine, gives* $\boldsymbol{\omega}$ *as output.* It follows quite easily from the definition that the algorithmic complexity of a string $\boldsymbol{\omega} = \omega_{i_1}\omega_{i_2}\cdots$ can grow at most linearily with the length of $\boldsymbol{\omega}$. It is in fact always possible to reproduce the string by means of a program of the form: "*write* $\omega_{i_1}\ \omega_{i_2}\cdots$." This actually means that the length of a string constitutes (up to a constant) an upper bound for the complexity of the string itself,[d]

$$K_{\mathrm{Cl}}(\boldsymbol{\omega}_n) \lesssim n. \tag{1}$$

Naturally, there are sequences for which this upper bound is far too large: it is easily shown, for example, that the complexity of a periodic string grows only *logarithmically* with the length of the sequence. A string is said to be *complex* (or structureless, or random) if its algorithmic complexity grows linearly with its length: these are the strings typically generated by random sources (such as, for example, a coin toss).

We want to find a "natural" upper bound also for the complexity of a quantum state. A difficulty arises from the fact that there seems to be no natural quantum counterpart to the classical "number of bits in the string." We thus find it necessary to look for another quantity, classically related to the number of bits, for which such a counterpart exists.

In order to do this, let us consider the set of all infinite binary strings: it is easily shown that this set is isomorphic to the unit interval $[0,1]$.[e] Through this isomorphism, it is possible to construct a (normalized) measure on the set of infinite

[d]Here and in the following, we use the "approximate" signs ($\simeq$, $\lesssim$) when the relationships are expressed up to the leading order.

[e]For each $\alpha \in [0,1]$, there exists in fact one (and only one) sequence $\{\alpha_i\}_i$ (with $\alpha_i \in \{0,1\}$) such that $\alpha = \sum_i \alpha_i 2^{-i}$.

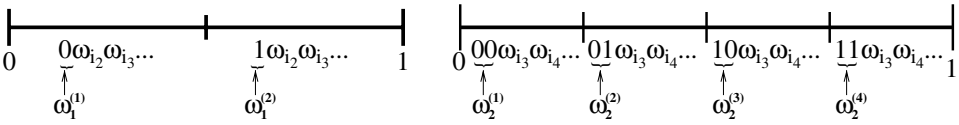


Fig. 1. If we consider the isomorphism between infinite binary strings and the interval $[0,1] \in \mathbb{R}$, then each n-bit string $\boldsymbol{\omega}_n$ identifies the subset of all infinite strings whose first n bits coincide with $\boldsymbol{\omega}_n$. This subset corresponds to a subinterval of $[0,1]$ with length (volume) 2^{-n}.

strings. Any n-bit binary string $\boldsymbol{\omega}_n$ identifies the set of infinite strings whose first n bits coincide with $\boldsymbol{\omega}_n$ (Fig. 1): the volume of this set (a ball, $B_{\boldsymbol{\omega}_n}$) is $V(B_{\boldsymbol{\omega}_n}) \simeq 2^{-n}$.

The unit interval is thus divided in $[V(B_{\boldsymbol{\omega}_n})]^{-1} \simeq 2^n$ subintervals, each identified by a n-bit sequence $\boldsymbol{\omega}_n^{(i)}$ with $i = 1, 2, \ldots, 2^n$. Once we have numbered all the n-bit sequences, it follows immediately that each of them can be reproduced by a program that specifies its index i, that is, by a program that requires at most (up to some constant) $n = -\log 2^{-n} = -\log V(B_{\boldsymbol{\omega}_n})$ bits. This simple "counting" argument gives an upper bound for the complexity of an n-bit string which coincides with the one given by Eq. (1):

$$K_{\mathrm{Cl}}(\boldsymbol{\omega}_n) \lesssim -\log V(B_{\boldsymbol{\omega}_n}). \tag{2}$$

The advantage of this argument is that it can be easily adopted to find an upper bound to the complexity of quantum states.

In the quantum case, in fact, we will be looking at a circuit (or quantum Turing machine or any other appropriate model) that reproduces a normalized quantum state $|\varphi\rangle$ up to a fixed (given) precision ε. This means that the circuit must prepare some quantum state $|\psi\rangle$ such that $|\langle\psi|\varphi\rangle|^2 \geq 1 - \varepsilon$. The set of all the normalized states that satisfy this condition difines a "patch" on the surface of a $(2^N - 1)$-dimensional hypersphere,[f] with volume V such that $V \simeq 2^{-N}\varepsilon^{2^N}$ (see Fig. 2).

This means that, if $K^\varepsilon(|\varphi\rangle)$ is the complexity of the state $|\varphi\rangle \in \mathcal{Q}_N$ (when reproduced with precision ε), we must have

$$K(|\varphi\rangle) \lesssim -\log V \;\Leftrightarrow K^\varepsilon(|\varphi\rangle) \lesssim -2^N \log \varepsilon + N. \tag{3}$$

For high enough precision ($\varepsilon \lesssim 2^{-N}$) the linear term can be omitted; we will thus usually consider simply the condition

$$K^\varepsilon(|\varphi\rangle) \lesssim -2^N \log \varepsilon. \tag{4}$$

Remark 1. We underline that this is a preliminary condition, that should hold true independently of the way one chooses to define quantum algorithmic complexity. It has in fact no relation to the model chosen to reproduce the state, but depends instead only on *a priori* properties such as the dimension of the space where the state is defined and the precision with which the state must be reproduced.

[f]The "patch" is given by the intersection of a cone (with angle determined by ε) with the surface of the hypersphere.

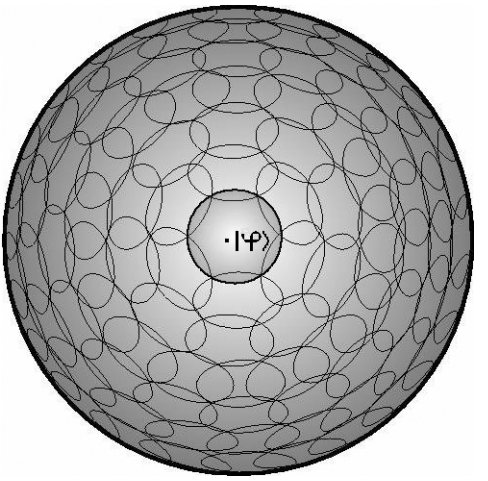


Fig. 2. A state $|\varphi\rangle$, determined with a finite precision ε, determines a "patch" on the ball of normalized N-qubit states. This figure is inspired by Fig. 4.18 in Ref. 8. Courtesy of I. Chuang and M. Nielsen.

3. Algorithmic Complexity on a Fixed Set of States

In the following section, we will assume to have fixed a complete gate basis $B = \{G_1, G_2, \ldots, G_k\}$ and we will consider only states $|\varphi\rangle$ that can be prepared exactly by circuits built from B.

Once we fix a code (that is, an alphabet $\Omega = \{\omega_1, \omega_2, \ldots, \omega_l\}$), the procedure to compute the algorithmic complexity of a state $|\varphi\rangle$ the procedure is very simple.

(i) With the gates contained in the basis B, build a circuit $\mathcal{C}^B$ such that $\mathcal{C}^B|0\rangle = |\varphi\rangle$.
(ii) Code the circuit, obtaining a classical sequence $\boldsymbol{\omega}^{\Omega}(\mathcal{C}^B) = \omega_{i_1}\omega_{i_2}\cdots\omega_{i_m}$ of symbols $\omega_{i_j} = \omega_{i_j}(\mathcal{C}^B) \in \Omega$.

Remark 2. Most of the code (that is, excluding some parts, e.g. a "new line" instruction or a way to identify the different lines, that will be more or less common to all codes) is strictly related to the gate basis. In fact the code can be seen as a function that associates each gate of the basis with a symbol (letter) or group of symbols (word).

(iii) We have now all the elements to define the algorithmic complexity of a state $|\varphi\rangle$.

Definition 1. The algorithmic complexity of a state, relative to the basis B, the code Ω and the circuit $\mathcal{C}^B$ is

$$K_{\mathrm{Net}}^{\Omega,B,\mathcal{C}^B}(|\varphi\rangle) = K_{\mathrm{Cl}}(\boldsymbol{\omega}^{\Omega}(\mathcal{C}^B)).$$

(iv) In general, there are more circuits that prepare the same state $|\varphi\rangle$, and in principle the corresponding complexities can be different. In order to define a property of the state itself (and not related to the circuit used to reproduce it), we consider the following definition.

Definition 2. The algorithmic complexity of the state $|\varphi\rangle$, relative to the code Ω and the gate basis B is

$$K_{\text{Net}}^{\Omega,B}(|\varphi\rangle) = \min_{\mathcal{C}^B \in \tilde{\mathcal{C}}^B} K_{\text{Net}}^{\Omega,B,\mathcal{C}^B}(|\varphi\rangle), \tag{5}$$

where $\tilde{\mathcal{C}}_B$ is the set of all the circuits built with gates from B that prepare $|\varphi\rangle$.

Naturally, considering that the choices of code and basis are arbitrary, it is necessary to study how they influence the complexity of the state.

Proposition 1 ("Asymptotic" invariance of the complexity of a state for code choice). *If Ω and $\underline{\Omega}$ are two different codes, then, for any state $|\varphi\rangle$*

$$K_{\text{Net}}^{\Omega,B}(|\varphi\rangle) = K_{\text{Net}}^{\underline{\Omega},B}(|\varphi\rangle) + k_{\Omega,\underline{\Omega}}, \tag{6}$$

where $k_{\Omega,\underline{\Omega}}$ is a constant that depends only on Ω and $\underline{\Omega}$.

This can be seen as follows. For every $\mathcal{C}^B \in \tilde{\mathcal{C}}^B$, let $\boldsymbol{\omega}^{\Omega}(\mathcal{C}^B) = \omega_{i_1}^{\Omega}\omega_{i_2}^{\Omega}\cdots\omega_{i_m}^{\Omega}$ and $\underline{\boldsymbol{\omega}}^{\underline{\Omega}}(\mathcal{C}^B) = \underline{\omega}_{j_1}^{\underline{\Omega}}\underline{\omega}_{j_2}^{\underline{\Omega}}\cdots\underline{\omega}_{j_n}^{\underline{\Omega}}$ be the strings that code $\mathcal{C}^B$ using, respectively, codes Ω and $\underline{\Omega}$.

$k_{\Omega,\underline{\Omega}}$ represents the length of a "dictionary" with which it is possible to translate the description made using one code to that made using the other. Since both codes are finite, such dictionary is finite too. The invariance is asymptotic since, in general, $k'_{\Omega,\underline{\Omega}}$ can be very big and its relevance is lost only for $K_{\text{Net}}^{\Omega,B}(|\varphi\rangle) \gg 1$.

Considering the code-invariance property, we can from now on omit explicating the dependence on the code (we can imagine fixing it once and for all) and write simply: $K_{\text{Net}}^{B}(|\varphi\rangle)$.

4. Algorithmic Complexity for Arbitrary States

We want to generalize to arbitrary states what we have seen before. In this case, it is necessary to introduce the precision parameter ε: we can expect, in fact, that the greater the precision with which the state must be reproduced by the circuit, the longer the circuit itself will be.

Remark 3. The fact that the circuit becomes longer does not necessarily mean that the complexity of the string that codes it (and thus that of the state prepared by the circuit) increases. In fact, we can imagine some states that can be prepared with better and better precision by simply iterating the application of a particular gate (or set of gates). In this case, the length of the string that codes the circuit would naturally grow with the precision, but not so the complexity of the circuit.

However, this will not hold true in general, so it is necessary to keep the explicit dependence on the precision parameter.

The precision parameter enters into the definition of the algorithmic complexity of the state $|\varphi\rangle$ at the very first step, that is, in building the circuit that prepares it. When considering an arbitrary state in the Hilbert space, we will in fact need to specify the precision up to which the circuit must prepare the state. We will represent with $\mathcal{C}^{B,\varepsilon}$ a circuit (built with gates from B) that prepares $|\varphi\rangle$ with precision ε and with $\boldsymbol{\omega}^{\Omega}(\mathcal{C}^{B,\varepsilon})$ the (classical) sequence that codes $\mathcal{C}^{B,\varepsilon}$.

Definition 3. The algorithmic complexity of state $|\varphi\rangle$, relative to code Ω and gate basis B *with precision parameter* ε is

$$K_{\mathrm{Net}}^{\Omega,B,\varepsilon}(|\varphi\rangle) = \min_{\mathcal{C}^{B,\varepsilon}\in\tilde{\mathcal{C}}^{B,\varepsilon}} K_{\mathrm{Cl}}(\boldsymbol{\omega}^{\Omega}(\mathcal{C}^{B,\varepsilon})), \tag{7}$$

where $\tilde{\mathcal{C}}^{B,\varepsilon}(|\varphi\rangle)$ is the set of all the circuits built with gates from B that prepare $|\varphi\rangle$ with precision ε.

Remark 4. The proof of the code-invariance of the complexity of a state, seen in the previous section, did not require that the state was reproduced exactly by the circuits; thus the code-invariance property still holds true. Therefore, we can again omit the dependence on the code and write simply

$$K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle). \tag{8}$$

The definition we have given for the algorithmic complexity of a quantum state is based on the algorithmic complexity of the classical string that describes the preparation procedure of the state itself. A quantum state is then complex if such a string is complex (see Sec. 5). It has been shown in classical theory of algorithmic complexity,[2,9,12] though, that it is not possible to prove that a classical string is complex. The statement "The complexity of the string $\boldsymbol{\omega}$ is greater than N" is unprovable: proving the truth of such a statement would infact inevitably lead to contradiction. This fact, which has roots in Gödel's incompleteness theorem and derives more directly from Turing's "Halting Problem," implies that it is impossible to prove that a given classical string (and, as a consequence, a given quantum state) is complex. The converse statement "The given string (or state) has complexity less than N" can always be proved. Nevertheless, the distribution of algorithmic complexity in physical or mathematical ensembles, its value for classes of strings (or, in the quantum case, states) and its relations with other physically relevant quantities (such as, for example, entanglement) can be discussed in a rigorous manner.

5. Complexity and Precision

In this section, we want to verify that our definition of complexity satisfies the preliminary condition given in Sec. 2. In order to do this, it is necessary to estimate the upper bound of the algorithmic complexity of an arbitrary state $|\varphi\rangle$.

Recently, it has been shown[10] that using only the (continuous) set of all 1-qubit gates, plus the controlled NOT (C), it is possible to reproduce any unitary operation U over $\mathcal{Q}_N$ using $\mathcal{O}(4^N)$ gates. However, if one is interested in reproducing the action of a unitary operation on one particular (given) state, only $\mathcal{O}\left(2^N\right)$ such gates are sufficient; this number of gates is thus sufficient to prepare any state $|\varphi\rangle$ from the given initial state $|0\rangle$.

We consider now the Solvay–Kitaev theorem,[11] which implies that any circuit acting on $\mathcal{Q}_N$ built with m 1-qubit and C gates can be reproduced up to precision ε using $\mathcal{O}\left(m \log^c\left(\frac{m}{\varepsilon}\right)\right)$ gates from a finite gate basis ($c \in [1,2]$ is a constant whose exact value is yet not known).

It follows immediately that the action of any unitary transformation on $|0\rangle$ can be implemented (and thus any $|\varphi\rangle \in \mathcal{Q}_N$ can be prepared) up to precision ε via a circuit built only with gates from any finite and complete basis; furthermore, if M is the number of gates in the circuit, we have

$$M = \mathcal{O}\left(2^N \log^c\left(\frac{2^N}{\varepsilon}\right)\right) \Rightarrow M \simeq -2^N \log^c \varepsilon, \tag{9}$$

where the last expression is given considering only the leading order in the two variables.

Naturally, the length of the string that codes the circuit grows linearly with the number of gates of the circuit itself: in order to code a circuit that prepares a general state $|\varphi\rangle \in \mathcal{Q}_N$ we need thus a word whose length is (proportional to) M. Referring to the definition given in the previous paragraph, in order to say that a state $|\varphi\rangle$ is complex it is necessary that its complexity (or, equivalently, the complexity of the string $\boldsymbol{\omega}_\varepsilon^B(|\varphi\rangle)$) grows linearly with the length (M) of $\boldsymbol{\omega}_\varepsilon^B(|\varphi\rangle)$. From Eq. (9), we obtain immediately the logarithmic dependence on precision and the exponential dependence on N that were presented in Sec. 2 as expected upper bounds for the complexity; we have thus[g]

$$K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle \in \mathcal{Q}_N) \lesssim -2^N \log \varepsilon. \tag{10}$$

Complex States

We are now in the position to define what we mean by a *complex state*.

As defined above, the complexity of a state $|\varphi\rangle$ is the minimum complexity of a word that codes a circuit in $\tilde{\mathcal{C}}_\varepsilon^B(|\varphi\rangle)$. This means that there is a circuit $\hat{\mathcal{C}}_\varepsilon^B(|\varphi\rangle) \in \tilde{\mathcal{C}}_\varepsilon^B(|\varphi\rangle)$, coded (using some alphabet) by $\hat{\boldsymbol{\omega}}_\varepsilon^B(|\varphi\rangle)$, such that $K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle) = K_{\mathrm{Cl}}(\hat{\boldsymbol{\omega}}_\varepsilon^B(|\varphi\rangle))$.

Remark 5. It is in general possible that there are more circuits that satisfy the same condition $K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle) = K_{\mathrm{Net}}(\hat{\mathcal{C}}_\varepsilon^B(|\varphi\rangle))$. In this case, it is sufficient to choose one of them. The choice is arbitrary and not relevant for the following.

[g]Here and in the following, we omit the dependence on the constant c, as it would only change the dependence on the precision (in terms of a polynomial factor), which is not essential in the following discussion.

Once we have associated a classical string (the *characterizing string* $\hat{\boldsymbol{\omega}}_{\varepsilon}^{B}(|\varphi\rangle)$) to the state $|\varphi\rangle$ we can introduce the following definition:

Definition 4. A quantum state $|\varphi\rangle \in \mathcal{Q}_N$ is said to be *complex* if the classical string $\hat{\boldsymbol{\omega}}_{\varepsilon}^{B}(|\varphi\rangle)$ is complex.

As always, a classical string $\boldsymbol{\omega}$, of length N, is said to be complex if $K_{\mathrm{Cl}}(\boldsymbol{\omega}) \simeq N$.

Remark 6. This definition satisfies the intuitive idea of the complexity of a state. Let us, for illustration, consider the following situation. Alice has obtained a state ($|\varphi\rangle$) and wants Bob to reproduce it (at least with some precision ε). Anticipating this situation, they had previously agreed on a common code. All that Alice then has to pass to Bob is the information on how to compose a circuit that prepares $|\varphi\rangle$ with the given precision, and this means sending Bob the string $\boldsymbol{\omega}(\mathcal{C}_{\varepsilon}^{B}(|\varphi\rangle))$ that codes the circuit. In this case, the complexity of the state $|\varphi\rangle$ measures exactly the minimum amount of information that Alice needs to send to Bob. We underline that this information is *not* given by the length of the coding string, but by its complexity. This simply reflects the fact that a state could be prepared using a very big circuit (that will be coded by a consequently long string), but a very simple one (for example, a circuit obtained repeating many times the same set of gates). In this case, the amount of information Alice needs will be much smaller than the length of the coding word.

6. The "Basis Problem"

The definition we have given for the algorithmic complexity of a state has a very strong dependence on the choice of the basis. Fixing a particular state, it is in fact possible to build a particular basis so that the description of $|\varphi\rangle$ is trivial. One could thus argue that the definition has no relevant meaning.

Let us consider again the situation in which Alice prepares a state and wants to describe it to Bob. If they have previously agreed on using a certain gate basis, then Alice has only to describe to Bob the circuit (that means passing to Bob the sequence $\boldsymbol{\omega}_{\mathcal{C}}$). If they have not agreed on a particular gate basis, then Alice could indeed build a circuit using the "best" basis, but in this case she would have to describe the basis itself to Bob, and this would be in general a similarly difficult task.[h]

One might nevertheless wonder whether there is some particular basis that allows one to describe all (or almost all) states with simple circuits. If such a basis existed, it would obviously be convenient for Alice and Bob to agree on using that! In this case, we would obtain that (almost) all quantum states are non-complex. In the following, we will show that such a basis cannot exist as, once *any* gate basis

[h]This, in fact, would again require the description of arbitrary unitary transformations.

is fixed, the number of non-complex states is always small in relation to the total number of states.

In classical information theory, it is well known that the number of compressible (bit) strings is "small"; more precisely, one has

$$\frac{\#\{\boldsymbol{\omega}_n = \omega_{i_1} \cdots \omega_{i_n} | K_{\mathrm{Cl}}(\boldsymbol{\omega}_n\} < c\}}{\#\{\boldsymbol{\omega}_n = \omega_{i_1} \cdots \omega_{i_n}\}} \leq \frac{2^c - 1}{2^n}. \tag{11}$$

Now let us consider the quantum case: as we have seen in the previous paragraphs, once we fix a basis B and a precision parameter ε, we can associate to every quantum state $|\varphi\rangle$ a $(-2^N \log \varepsilon)$-bit string $\hat{\boldsymbol{\omega}}_\varepsilon^B(|\varphi\rangle)$ whose classical algorithmic complexity coincides with the complexity of $|\varphi\rangle$. Applying Eq. (11) to the set of strings $\hat{\boldsymbol{\omega}}_\varepsilon^B$, we obtain

$$\frac{\#\{|\varphi\rangle \in \mathcal{Q}_N | K_\varepsilon^B(|\varphi\rangle) < c\}}{\#\{|\varphi\rangle|_\varepsilon\}} \leq \frac{2^c - 1}{2^{-2^N \log \varepsilon}} \simeq 2^{2^N \log \varepsilon + c}, \tag{12}$$

where with $\#\{|\varphi\rangle|_\varepsilon\}$ we represent the number of ε-distinguishable normalized states. Such a relation holds true also in the case when $c = c(N, \varepsilon)$ is a function. State $|\varphi\rangle$ will be non-complex only if its complexity is $o(-2^N \log \varepsilon)$. This means that the right member of the inequality becomes $2^{2^N \log \varepsilon + o(-2^N \log \varepsilon)} \simeq 2^{2^N \log \varepsilon} \ll 1$. Thus, applying Eq. (12) to these states, one obtains that for any fixed basis B, the number of non-complex states is exponentially small.

Remark 7. There are cases in which there exists an invariance for basis choice. This happens, for example, when we consider a *coarsening* of the gate basis, that is if we consider two gate bases $\mathcal{B}$ and $\underline{\mathcal{B}}$, one of which ($\underline{\mathcal{B}}$) constituted of non-elementary gates that can be built with gates from $\mathcal{B}$ (e.g. $\underline{\mathcal{B}}$ contains a Toffoli gate, while $\mathcal{B}$ is chosen as in Example 1). In this case, in fact, any circuit made by gates from $\underline{\mathcal{B}}$ can be reproduced by one made by gates from $\mathcal{B}$. The string that codes this circuit will in general be longer than that of the original circuit, but their complexities will change only for a (small) constant $k_{\mathcal{B},\underline{\mathcal{B}}}$ (that represents the length of a "dictionary" between the two gate bases).

7. Entropy and Complexity

In classical information theory, one can consider a random source that emits (with a certain probability distribution) letters drawn from some finite alphabet. In this case, there is a strong relationship between the Shannon entropy of the source and the algorithmic complexity of the emitted sequences. In particular, if $\boldsymbol{\omega}_n$ is an n-letter sequence and $p(\boldsymbol{\omega}_n)$ is its probability, one obtains[13]

$$|\langle K_{\mathrm{Cl}}(\boldsymbol{\omega}_n) - H \rangle| \leq c, \tag{13}$$

where the average is taken over all n-letter sequences, $H = -\sum_{\boldsymbol{\omega}_n} p(\boldsymbol{\omega}_n) \log p(\boldsymbol{\omega}_n)$ is the Shannon entropy of the source, and c is a constant that depends on the probability distribution. When such a distribution does not depend on the length

of the sequences (as in the case of Bernoulli sources), this implies that, in the limit of $n \gg 1$, the average complexity production of the source coincides with its entropy production

$$\frac{\langle K_{\mathrm{Cl}}(\boldsymbol{\omega}_n)\rangle}{n} \xrightarrow{n\to\infty} \frac{H}{n}. \tag{14}$$

It comes quite natural to seek a similar relation in the context of quantum information theory. In order to do this, we will first of all find the corresponding relation in the case of a classical source that emits words (and not letters). After that, we will define what we mean by quantum source and find, in this case, the wanted relationship between complexity and entropy.

7.1. *Classical case*

Let us consider now a variation of the classical letter-source: in this case, we will have a source that emits, with some given probability, *words* drawn from a finite dictionary D. Each of the words will be a sequence of letters of alphabet A; without loss of generality, we can assume all words to have the same length l. The output of such a source will thus be a sequence of words (or *sentence*). As the Shannon entropy depends only on the probability distribution, once we fix such a distribution, it is the same for letter- and for word-sources. If we consider the complexity of the emitted message, though, it is evident that there must be some differences. A word-source that emits m objects will in fact have transmitted a sequence of ml letters: it is now easy to believe that the complexity of such a sequence can be higher than that of an m-letter sequence (even though letters and words are emitted following the same probability distribution). This is an immediate consequence of the fact that words are composite objects that have non-zero complexity themselves.

Naturally, when we consider very long output sentences, such a difference becomes negligible. Any sentence can in fact be reduced to a word by a program that associates to each word a symbol: the length of this program will be determined by the complexity of the dictionary (that is, by the complexity of the single words in the dictionary) and will thus be bound by the dictionary length $l\#D$. This contribution, though, can be extremely relevant while we consider "short" sentences, that is, sentences whose lengths are comparable to that of the dictionary.

Thus, if $\boldsymbol{\Omega}_m = \boldsymbol{\omega}_{i_1} \cdots \boldsymbol{\omega}_{i_m}$ is an m-word sentence, its complexity is given by

$$K_{\mathrm{Cl}}(\boldsymbol{\Omega}_m) \simeq K_{\mathrm{Cl}}(\boldsymbol{i}_m = i_1 \cdots i_m) + \sum_{j=1}^{\#D} K_{\mathrm{Cl}}(\boldsymbol{\omega}_j) \lesssim K_{\mathrm{Cl}}(\boldsymbol{i}_m) + l\#D, \tag{15}$$

where i_k is the symbol that codes $\boldsymbol{\omega}_{i_k}$, and l is the common length of the words $\boldsymbol{\omega}_k \in D$.

The sequences $\boldsymbol{i}_m$ can easily be seen as the outputs of a letter-source whose entropy H coincides with that of the considered word-source: for such sequences, thus, the relation expressed by Eq. (13) still holds true. When generalizing it to word-sources, though, it is necessary to consider the contribution of the dictionary,

obtaining

$$\left| \langle K_{\mathrm{Cl}}(\boldsymbol{\Omega}_m) - H \rangle - \sum_{j=1}^{\#D} K_{\mathrm{Cl}}(\boldsymbol{\omega}_j) \right| \leq c_1. \tag{16}$$

Naturally, being the complexity of the dictionary constant, Eq. (14) remains of the same form for word-sources, too:

$$\frac{\langle K_{\mathrm{Cl}}(\boldsymbol{\Omega}_m) \rangle}{m} \xrightarrow{m \to \infty} \frac{H}{m}. \tag{17}$$

7.2. *Quantum case*

Before being able to say anything for the quantum case, it is necessary to specify what we will consider as a quantum source. In principle, any mixed state can be viewed as a quantum source. In the following, though, we will consider a quantum source as a "black box" that emits (with a given probability) pure states drawn from a given set.[14] This corresponds to considering not a general mixed state, but rather a well-defined mixture (or blend[15]) $\{(p_j, |\varphi_j\rangle)\}_{j \in D}$ of pure states ($|\varphi_j\rangle \in \mathcal{Q}_N$). In this case, the source has a kind of "semi-classical" nature: it is in fact quantum only in the sense that it emits quantum states and not (as in the cases considered in the previous paragraphs) classical objects, while we demand that it cannot, for example, emit states that are other than the tensor product of the ones in the ensemble. For these sources, one can define the Shannon entropy, and it coincides with the Shannon entropy of a classical source that emits objects (letters or words) with the same probability distribution $\{p_j\}_{j \in D}$.

As we have seen when defining the complexity of a quantum state, once we fix a finite precision ε, it is possible to describe any quantum state by means of a finite word (whose length depend both on the dimension of the Hilbert space in which the state lives and on the value of ε). This fact allows us to identify a quantum source of this kind with a corresponding word-source. The expression for the complexity of the emitted messages (tensor-product states) follows thus immediately from what seen above

$$\begin{aligned} K_{\mathrm{Net}}^{B,\varepsilon}(|\boldsymbol{\Phi}_m\rangle = |\varphi_{i_1}\rangle|\varphi_{i_2}\rangle \cdots |\varphi_{i_m}\rangle) &\simeq K_{\mathrm{Cl}}(\boldsymbol{i}_m = i_1 \cdots i_m) + \sum_{j \in D} K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi_j\rangle) \\ &\lesssim K_{\mathrm{Cl}}(\boldsymbol{i}_m) - \#D 2^N \log \varepsilon, \end{aligned} \tag{18}$$

where i_k is the symbol that codes state $|\varphi_{i_k}\rangle$. We underline that, in this case, the contribution due to the complexity of the dictionary can be relevant indeed, being bound by the complexity of the states (that can, in principle, be very large). The relation between the entropy H of the source (that is, the Shannon entropy of the probability distribution $\{p_j\}_j$) and the complexity of the message is analogously obtained:

$$|\langle K_{\mathrm{Net}}^{B,\varepsilon}(|\boldsymbol{\Phi}_m\rangle) - H \rangle - \sum_{j \in D} K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi_j\rangle)| \leq c. \tag{19}$$

As in the classical case, though, if we consider the limit of infinitely long state sequences, the complexity rate and the entropy rate tend to coincide: in this limit, thus, the source "reveals" its semiclassical nature.

Remark 8. Quite naturally, one could ask what happens if we have the possibility to apply Schumacher's noiseless coding theorem[14] and thus compress the emitted states. If $S(\rho)$ is the von Neumann entropy of the source $\rho = \sum_j p_j |\varphi_j\rangle\langle\varphi_j|$, each state can be compressed into a new state $|\varphi'_j\rangle$ in a $(2^{NS(\rho)})$-dimensional Hilbert space. In this case, we can rewrite Eq. (18) as

$$\begin{aligned} K^{B,\varepsilon}_{\mathrm{Net}}(|\boldsymbol{\Phi}_m\rangle = |\varphi_{i_1}\rangle|\varphi_{i_2}\rangle\cdots|\varphi_{i_m}\rangle) &\simeq K^{B,\varepsilon}_{\mathrm{Net}}(|\boldsymbol{\Phi'}_m\rangle = |\varphi'_{i_1}\rangle|\varphi'_{i_2}\rangle\cdots|\varphi'_{i_m}\rangle) \\ &\simeq K_{\mathrm{Cl}}(\boldsymbol{i}_m = i_1\cdots i_m) + \sum_{j\in D} K^{B,\varepsilon}_{\mathrm{Net}}(|\varphi'_j\rangle) \\ &\lesssim K_{\mathrm{Cl}}(\boldsymbol{i}_m) - \#D(NS(\rho))^2 2^{NS(\rho)} \log\varepsilon. \end{aligned} \tag{20}$$

8. Applications and Examples

8.1. *Complexity of copies*

One of the properties of classical algorithmic complexity is that obtaining m copies of a given string is (almost) free, the dependence of the complexity on the number of copies being only logarithmic,

$$K_{\mathrm{Cl}}(\boldsymbol{\omega}^{(m)} = \underbrace{\boldsymbol{\omega}\boldsymbol{\omega}\cdots\boldsymbol{\omega}}_{m \text{ times}}) \leq K_{\mathrm{Cl}}(\boldsymbol{\omega}) + \mathcal{O}(\log m).$$

This bound is easily obtained by considering that, once one has a program that reproduces string $\boldsymbol{\omega}$ it is sufficient to run it m times to reproduce $\boldsymbol{\omega}^{(m)}$.

In the case of a quantum system, the situation is slightly more involved. It is first of all necessary to decide exactly what we mean by "preparing m copies" of a quantum state: in fact, either we will require the circuit to prepare m times a quantum state $|\varphi\rangle \in \mathcal{Q}_N$, with some fixed precision parameter ε, or it must prepare the global state $|\varphi\rangle^{\otimes m} \in \mathcal{Q}_N^{\otimes m}$ with the given precision. The two situations are extremely different.

In the first case, we obtain the same relation we have seen in the classical case

$$K^{B}_{\mathrm{Net}}(|\varphi\rangle|^{(m)}_{\varepsilon}) = K^{B}_{\mathrm{Net}}(\underbrace{|\varphi\rangle|_{\varepsilon}\cdots|\varphi\rangle|_{\varepsilon}}_{m \text{ times}}) \leq K^{B,\varepsilon}_{\mathrm{Net}}(|\varphi\rangle) + \mathcal{O}(\log m),$$

where the expression $|\varphi\rangle|^{(m)}_{\varepsilon}$ reminds us that each copy of $|\varphi\rangle$ is reproduced with precision ε. This is an immediate consequence of the definition of complexity: a circuit that prepares $|\varphi\rangle|^{(m)}_{\varepsilon}$ can be in fact built by repeating m times the one that prepares $|\varphi\rangle$ with precision ε.

If, instead, we require the state $|\varphi\rangle^{\otimes m}$ be prepared with precision ε, the situation is different and the classical relation does not (necessarily) hold true any longer. In this case, the relation between the complexity of $|\varphi\rangle$ and that of $|\varphi\rangle^{\otimes m}$ has

the form

$$K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle^{\otimes m}) \leq K_{\mathrm{Net}}^{B,\frac{\varepsilon}{m}}(|\varphi\rangle) + \mathcal{O}(\log m) \lesssim -2^N \log \frac{\varepsilon}{m}. \tag{21}$$

This expression follows immediately by the fact that the state $|\varphi\rangle^{\otimes m}$ can be prepared with precision ε by a circuit $\mathcal{C}_{\varepsilon}^{B}(|\varphi\rangle^{\otimes m})$ built with m identical copies of a smaller circuit $\mathcal{C}_{\varepsilon/4m}^{B}(|\varphi\rangle)$.[16]

The last inequality follows immediately from Eq. (10). We underline that in most cases (that is when $4m\varepsilon^{m-1} \leq 1$) this bound is much stricter than the one we could obtain by directly applying (10) with which one has: $K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle^{\otimes m}) \lesssim -m2^N \log \varepsilon$.

8.2. *Complexity of graph states*

Graph states are multi-particle entangled states that can uniquely be described by mathematical graphs, where the vertices of the graph take the role of qubits and edges represent unitary operations between the relative qubits.[17]

Given a graph $G = (V, E)$, one can easily prepare the corresponding graph state using the following procedure:

(i) prepare all qubits in state $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$;
(ii) when there is an edge between two vertices k and l, apply a controlled-phase gate between the two qubits. This actually means applying to the two qubits a transformation of the form $U_{kl} = \mathrm{e}^{-i\frac{\pi}{4}(\mathbf{1}^{(k)}-\sigma_z^{(k)})(\mathbf{1}^{(l)}-\sigma_z^{(l)})}$.

The resulting state $|\psi\rangle_G$ will be an entangled state uniquely described by the graph G.

Once the number N of vertices are fixed, there are at most $2^{N(N-1)/2}$ different graphs $G_1, G_2, \ldots, G_{2^{N(N-1)/2}}$ (each vertex can in fact be connected or not to each other vertex by an edge). Correspondingly, given N qubits, we can build at most $2^{N(N-1)/2}$ different graph states $|\psi\rangle_{G_1}, |\psi\rangle_{G_2}, \ldots, |\psi\rangle_{G_{2^{N(N-1)/2}}}$.

Once we define some sort of lexicographic order in the set of all graphs (or equivalently in that of the graph states), only $\mathcal{O}(N^2)$ bits are sufficient to specify a determinate state. This value constitutes an upper bound for the complexity of a graph state.

We will now show that this same value can be obtained for the algorithmic complexity of a graph state. As seen above in the more general case, the upper bound for the complexity of a state is obtained finding a bound for the length (and thus for the complexity) of the characterizing string. In the case of graph states, this bound is obtained in a simple way: if N is the number of vertices of the graph, the maximum number of edges is $N(N-1)/2$. This implies that a corresponding graph state can be obtained applying to the N qubits at most $N + N(N-1)/2$ gates (N Hadamard gates, needed to initially prepare all the qubits in state $|+\rangle$,

and $N(N-1)/2$ controlled-phase gates). If G has N vertices, we have thus

$$K_{\mathrm{Net}}(|\psi\rangle_G) = K_{\mathrm{Cl}}(\hat{\boldsymbol{\omega}}^B(|\psi\rangle_G)) \lesssim l(\hat{\boldsymbol{\omega}}^B(|\psi\rangle_G)) \lesssim N + N(N-1)/2 \lesssim N^2. \quad (22)$$

As there is no dependence on the precision parameter, Eq. (22) holds true only if we have the possibility of reproducing exactly the controlled-phase and the Hadamard gates.

It is nevertheless possible to obtain a (more general) upper bound for the complexity of this family of states, valid also in the case in which our basis is not of the above type. In order to do this we will use some considerations regarding the complexity of sentences, seen in Sec. 7.1.

In order to prepare a graph state, only two different types of gates, controlled-phase and Hadamard, are sufficient. If these gates cannot be reproduced exactly by those in our basis then, naturally, the wanted state can be prepared only with finite precision. As we have seen above, we need at most $\mathcal{O}(N^2)$ of these gates to prepare any arbitrary graph state. To guarantee that the desired state is prepared with precision ε, it is thus enough to reproduce each gate with precision ε/N^2. From the Solvay–Kitaev theorem, we know that implies we need to use $\mathcal{O}(-\log \frac{\varepsilon}{N^2})$ gates to simulate each Hadamard (or controlled-phase) gate. Once we code the circuit, to each Hadamard (or controlled-phase) will correspond the same $\mathcal{O}(-\log \frac{\varepsilon}{N^2})$-letter word. Using Eq. (15) (where $\#D = 2$)[i] we obtain

$$K_{\mathrm{Net}}(|\psi\rangle_G) \lesssim N^2 - \log \frac{\varepsilon}{N^2}. \quad (23)$$

The case of *weighted graph states*,[18] which occur, for example, in the description of spin gases,[19] is different. These states are generalizations of graph states, in which every edge is specified by a (different) phase. The procedure to prepare these states is analogous to that illustrated for graph states; the only difference is that, in this case, whenever two vertices k and l are connected, one must now apply a transformation of the form $U_{kl} = \mathrm{e}^{-i\frac{\varphi_{kl}}{4}(\mathbf{1}^{(k)}-\sigma_z^{(k)})(\mathbf{1}^{(l)}-\sigma_z^{(l)})}$, with a specified phase φ_{kl}.

While the total number of gates in the circuit is still at most $\mathcal{O}(N^2)$, in this case it is not sufficient to consider only Hadamard and controlled-phase gates as, in principle, each phase-gate could be different. When preparing a weighted graph state, it is thus necessary to reproduce $\mathcal{O}(N^2)$ different gates with precision $\mathcal{O}(-\log \frac{\varepsilon}{N^2})$. Again, we can obtain an upper bound for the complexity of these states by using Eq. (15), only that this time the size of the dictionary does depend on N: $\#D \simeq N^2$. We have thus

$$K_{\mathrm{Net}}(|\psi\rangle_{\mathrm{W.G.}}) \lesssim N^2 - N^2 \log \frac{\varepsilon}{N^2} \simeq -N^2 \log \frac{\varepsilon}{N^2}. \quad (24)$$

[i]Actually, the cardinality of D can be larger than 2, as there will be some words that correspond to operations such as "new line" or similar, but it will always be independent of N.

9. Entanglement and Complexity

Let us consider a state $|\varphi\rangle \in \mathcal{Q}_N$; suppose it can be written as

$$|\varphi\rangle = |\varphi_1\rangle \otimes |\varphi_2\rangle \otimes \cdots |\varphi_J\rangle, \quad \text{with } |\varphi_j\rangle \in \mathcal{Q}_{N_j} \text{ such that } \dim(\mathcal{Q}_{N_j}) = 2^{N_j}, \quad \text{and} \quad \sum_{j=1}^{J} N_j = N.$$

This means that the state $|\varphi\rangle$ is not totally entangled, and it can thus be considered as the tensor product of other (possibly entangled) states $|\varphi_j\rangle$ (see Fig. 3).

As a consequence of this fact, we have

$$K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle) \lesssim \sum_{j=1}^{J} K_{\mathrm{Net}}^{\varepsilon/J}(|\varphi_j\rangle) \lesssim -\sum_{j=1}^{J} 2^{N_j} \log \frac{\varepsilon}{J}. \tag{25}$$

We want to show that this upper bound is actually stricter than the one obtained for general states in $\mathcal{Q}_N$: let us rewrite the bound given by Eq. (10) as $K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle) \lesssim -\sum_{j=1}^{J} \frac{2^N}{J} \log \varepsilon$. The wanted inequality is proved by considering

$$2^{N_j} \log \frac{J}{\varepsilon} \leq \frac{2^N}{2^{J-1}} \log \frac{J}{\varepsilon} < \frac{2^N}{J} \log \frac{1}{\varepsilon}.$$

Thus, the *maximal complexity* can be obtained *only* by a truly N-party entangled state (in the sense that it cannot be written as tensor product of states contained in subspaces of $\mathcal{Q}_N$). We stress that this consideration does not imply that all totally entangled states have maximal complexity (as a counterexample, it is enough to consider the GHZ states).

It is nevertheless interesting to consider how a property that is characteristic of quantum systems (that is, entanglement) has a direct effect on the complexity of a state.

Example 2 (Complexity of a completely separable state). As an example, let us consider a state $|\varphi\rangle \in \mathcal{Q}_N$ of the form $|\varphi\rangle = \bigotimes_{j=1}^{N} |\varphi_j\rangle$, $|\varphi_j\rangle \in \mathcal{Q}_1$. In this

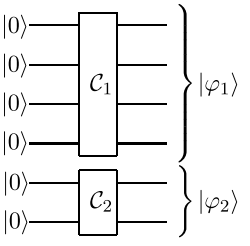


Fig. 3. If a state $|\varphi\rangle$ is separable (e.g. $|\varphi\rangle = |\varphi_1\rangle \otimes |\varphi_2\rangle$) then it is possible to divide the circuit that prepares $|\varphi\rangle$ into two independent sub-circuits. This allows us to lower the bound for the complexity of the state, thus proving that full N-party entanglement is necessary in order to have maximal complexity.

case, we have

$$K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi\rangle) \lesssim \sum_{j=1}^{N} K_{\mathrm{Net}}^{B,\varepsilon/N}(|\varphi_j\rangle) \lesssim -2N \log \frac{\varepsilon}{N},$$

and thus the complexity of a separable state grows only at most linearly with the number of qubits.

Remark 9. As we have seen, in general the growth of the complexity of a state with the number of qubits is exponential, and this is substantially different to what happens in the classical case, where the upper bound is linear with the number of bits. In this last example, though, we see that the absence of entanglement re-establishes the classical limit: entanglement thus proves again to be a fundamental feature that distinguishes quantum objects from classical ones.

9.1. *Complexity and Schmidt measure*

The relation, illustrated by the examples above, between the complexity of a state and its entanglement can be formalized if we choose the *Schmidt measure*[20] as a measure of entanglement. Given a quantum state $|\Phi\rangle \in \mathcal{Q}_N$, its Schmidt measure $E_S(|\varphi\rangle)$ is defined as the logarithm of the minimum number r of separable states $|\varphi_i\rangle \in \mathcal{Q}_N$ such that $|\Phi\rangle = \sum_{i=1}^{r} \alpha_i |\varphi_i\rangle \in \mathcal{Q}_N$. In the following, we show how the knowledge of the Schmidt measure of a quantum state $|\Phi\rangle$ allows us to give a better bound on its complexity.

In order to do this, we will build a circuit that prepares $|\Phi\rangle$. The general idea is to prepare such a circuit "using" the circuits that prepare the states $|\varphi_i\rangle$ that appear in the decomposition of $|\Phi\rangle$. Such a circuit (see Fig. 4) is built with the aid of $\log r$ "ancilla" qubits, initially prepared in the superposition state $|a\rangle = \sum_{i=0}^{r} \alpha_i |i\rangle$ (where $|i\rangle$ is the state whose binary representation gives the state of the $\log r$ qubits). We then apply to the initial state $|0\rangle$ a circuit consisting of controlled-$\mathcal{C}_{\varepsilon}^{B}(|\varphi_i\rangle)$ (in series), the application of each of which is conditional on the ancilla being in state $|i\rangle$. The last step is to project the ancilla on state $\sum_{i=1}^{r} |i\rangle/\sqrt{r} = \bigotimes_{j=1}^{\log r} (|0\rangle_j + |1\rangle_j)/\sqrt{2}$: state $|\Phi\rangle$ is prepared if the result is non-zero. If this is not the case, it is sufficient to repeat the procedure from the beginning.

Remark 10. The probability of this procedure succeeding is exponentially low, but this is not relevant from our point of view: we are in fact interested in the description of the procedure, not in the time or resources needed to apply it.

The complexity of the state $|\Phi\rangle$ can now be expressed in terms of the complexity of the ancilla state $|a\rangle$ and that of the remaining circuit.

The ancilla can in principle be any state of $\log r$ qubits; from what we have seen in the above sections, we have thus

$$K_{\mathrm{Net}}^{B,\varepsilon}(|a\rangle) \lesssim -2^{\log r} \log \varepsilon = -r \log \varepsilon.$$

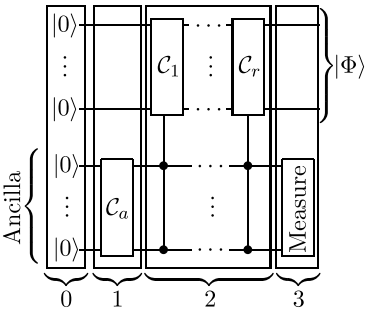


Fig. 4. The circuit shown is such that it is possible to give a bound on the complexity of a quantum state based on its entanglement. The circuit is composed of three main parts and requires the use of $\log r$ ancilla qubits.

(i) The initial state is $|\mathrm{Tot}(0)\rangle = |0\rangle_N \otimes |0\rangle_{\log r}$.
(ii) Prepare $\log r$ "ancilla" qubits in state $|a\rangle = \sum_{i=1}^{r} \alpha_i |i\rangle$. The total state is now $|\mathrm{Tot}(1)\rangle = |0\rangle_N \otimes |a\rangle$.
(iii) Apply to the initial state $|0\rangle$ a circuit consisting of controlled-$\mathcal{C}_i^{B,\varepsilon}$, the application of each conditional on the ancilla being in state $|i\rangle$: $|\mathrm{Tot}(2)\rangle = \sum_{i=1}^{r} \alpha_i |\varphi_i\rangle \otimes |i\rangle$.
(iv) Project the ancilla onto state $\sum_{i=1}^{r} |i\rangle/\sqrt{r}$: if the result is non-zero, state $|\Phi\rangle$ is prepared. If this is not the case, it is sufficient to repeat the procedure.

Once the ancilla is prepared in the required state, it is necessary to describe the remaining circuit. First of all, we must describe the individual circuits $\mathcal{C}_\varepsilon^B(|\varphi_i\rangle)$ (and this requires a number of bits bounded by the sum of the complexities of the single states $|\varphi_i\rangle$). Performing each of these circuits as a controlled operation requires additional $\mathcal{O}(-(\log r)^2 \log \varepsilon)$ gates from B. We have thus

$$\begin{aligned} K_{\mathrm{Net}}^{B,\varepsilon}(|\Phi\rangle) &\lesssim -r \log \varepsilon + \sum_{i=1}^{r} K_{\mathrm{Net}}^{B,\varepsilon}(|\varphi_i\rangle) - \mathcal{O}((\log r)^2 \log \varepsilon) \\ &\lesssim -r \log \varepsilon - 2Nr \log \varepsilon \simeq -2Nr \log \varepsilon, \end{aligned} \tag{26}$$

where we have used the fact that, the states $|\varphi_i\rangle$ being separable, their complexity is bound by $-2N \log \varepsilon$.

We notice that these results include the two particular cases treated previously. If $|\Phi\rangle$ is a separable state, in fact, we have $r = 1$, and Eq. (26) takes the form $K_{\mathrm{Net}}^{B,\varepsilon}(|\Phi\rangle) \lesssim -2N \log \varepsilon$. When instead the state $|\Phi\rangle$ is truly N-party entangled, its Schmidt measure is $\log r = N$ and we have $K_{\mathrm{Net}}^{B,\varepsilon}(|\Phi\rangle) \lesssim -2N2^N \log \varepsilon \simeq -2^N \log \varepsilon$ (up to a polynomial term).

10. Conclusions and Outlook

In this paper, we have introduced a new definition for the algorithmic complexity of quantum states. We have defined the complexity of a quantum state as the description complexity of its experimental preparation which is abstractly described via a quantum circuit. We have investigated the relation between the Shannon

entropy of a source and the algorithmic complexity of the emitted message. We could also straightforwardly apply this definition to find upper bounds for a number of interesting cases. We have seen a relation between entanglement and algorithmic complexity and shown how it is possible to give a bound on the complexity of a state in terms of its entanglement. In particular, we have seen that this implies that the absence of entanglement reduces the upper bound for the algorithmic complexity to the classical one.

While we have studied the algorithmic complexity of some classes of states, one could pursue this investigation analyzing other states, for example, states that appear in the context of quantum phase transitions and quantum adiabatic computation.[21]

Furthermore, we have done a preliminary study of the extension of the notion of complexity to mixed states. The first issue one has to address is the possible dependence of the complexity of a mixed state on the particular choice of blend (pure state mixture such that $\rho = \sum_i \lambda_i |\varphi_i\rangle\langle\varphi_i|$). We believe that such a dependence must be avoided when one wishes to define a physical property of mixed states. Considering this condition, there are still several possible ways to define complexity of mixed states. We choose to define the algorithmic complexity of a state ρ as $K^{B,\varepsilon}_{\mathrm{Net}}(\rho) = \min \prod_i [K^{B,\varepsilon}_{\mathrm{Net}}(|\varphi\rangle)]^{\lambda_i}$, where the minimum is taken over all possible blends. Thus, the complexity of a mixed state is the average complexity of the states that appear in its (optimal) mixture. The choice of the geometric average (rather than the arithmetic average) is motivated by the necessity of giving an adequate weight to the probability distribution. Such a definition reduces to the one we have introduced in this paper in the case of pure states. Furthermore, the relationship between entanglement and complexity is maintained. In particular, we have $K^{B,\varepsilon}_{\mathrm{Net}} \lesssim 3N2^{E_S(\rho)} \log \frac{1}{\varepsilon}$, where $E_S(\rho)$ is the Schmidt measure of the mixed state ρ as defined in Ref. 20. A deeper study of the properties of the algorithmic complexity of mixed states will be the subject of further work.

After completion of this work, we learnt about a previous work by Yamakami [quant-ph/0308072 (2003)], which introduces a related notion of complexity.

Acknowledgments

We want to thank Fabio Benatti and Paul Vitanyi for useful discussions and comments. We also thank Tomoyuki Yamakami for drawing our attention to his work. This work was supported in part by the Austrian Science Foundation (FWF), the Deutsche Forschungsgemeinschaft (DFG), and the European Union (IST-2001-38877, -39227, SCALA).

Appendix A. Quantum Algorithmic Complexity of a Classical String

We want to see if our definition is coherent with that of classical Kolmogorov complexity. In order to do this, we consider a classical N-bit string $\mathbf{x} = x_{i_1} x_{i_2} \cdots x_{i_N}$ (where $x_{i_j} \in \{0, 1\}$).

In order to use a procedure built to characterize the complexity of a quantum state, it is necessary to "translate" the classical string into a quantum state; this is easily done simply by considering the state $|\mathbf{x}\rangle = |x_{i_1}\rangle|x_{i_2}\rangle \cdots |x_{i_N}\rangle$, with $|x_{i_j}\rangle \in \{|0\rangle, |1\rangle\}$.

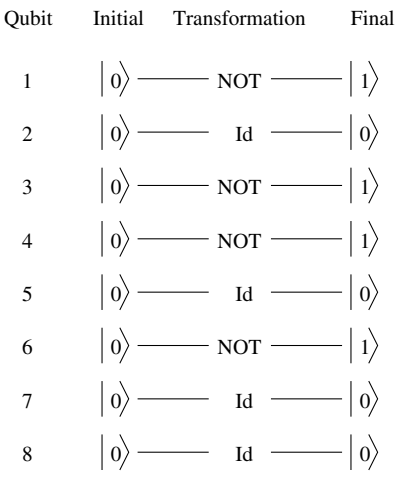
Considering the nature of this state, it is trivial to see that it can be obtained from the initial state $|0\rangle|0\rangle \cdots |0\rangle$ by simply applying one-qubit NOT gates to the bits corresponding to $|x_{i_j}\rangle = |1\rangle$ and leaving unvaried (applying the identity transformation) the bits $|x_{i_j}\rangle = |0\rangle$.

To encode these particular circuits, thus, three symbols are sufficent[j]:

$$I \leftrightarrow \text{identity},$$
$$N \leftrightarrow \text{NOT gate},$$
$$L \leftrightarrow \text{NEWLINE}.$$

Let us now consider a simple example:

$$\text{Classical string } \mathbf{x} = 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0$$
$$\text{Quantum state } |\mathbf{x}\rangle = |1\rangle\ |0\rangle\ |1\rangle\ |1\rangle\ |0\rangle\ |1\rangle\ |0\rangle\ |0\rangle.$$



After coding, then, we have the following string representing the circuit:

$$\boldsymbol{\omega}(\mathcal{C}) = N\ L \quad I\ L \quad N\ L \quad N\ L \quad I\ L \quad N\ L \quad I\ L \quad I\ L$$

(where, to help visualization, we have put larger spaces after line breaks L). It is evident that there is a direct correspondence between $\boldsymbol{\omega}(\mathcal{C})$ and the original classical string $\mathbf{x}$: in fact, a program that reproduces $\mathbf{x}$ reproduces also $\boldsymbol{\omega}(\mathcal{C})$ (with the agreement that $0 \leftrightarrow I$ and $1 \leftrightarrow N$) (it will be enough to add a *constant* part that tells the program to insert a line-break after every symbol). Thus, it follows

[j]In principle, in this case we need not define a complete gate basis as the NOT gate alone is sufficient.

immediately that the (classical) Kolmogorov complexity of $\boldsymbol{\omega}(\mathcal{C})$ coincides with that of $\mathbf{x}$. Considering the definition of the network complexity, we have immediately

$$K_{\mathrm{Net}}(|\mathbf{x}\rangle) \lesssim K_{\mathrm{Cl}}(\mathbf{x}). \tag{A.1}$$

Naturally, the exact procedure followed in the example can be applied to any N-bit string $\mathbf{x}$ so the result is true in general.

As in the case of the study of the complexity of graph states, in the expressions given above there is no dependence on the precision parameter ε as it is assumed that the identity and the NOT gates are included in the gate basis. If this is not the case, an argument analogous to that in Sec. 8.1 yields the following, more general, inequality

$$K_{\mathrm{Net}}^{B,\varepsilon}(|\mathbf{x}\rangle) - \log\frac{1}{\varepsilon} \lesssim K_{\mathrm{Cl}}(\mathbf{x}). \tag{A.2}$$

References

1. C. H. Bennett, P. Gács, M. Li, P. M. B. Vitanyi and W. H. Zurek, Thermodynamics of computation and information distance, in *Proc. 25th ACM Symp. Theory of Computation* (ACM Press, 1993).
2. A. N. Kolmogorov, Three approaches to the quantitative definition of information, *Probl. Inform. Trans.* **1** (1965) 1.
3. P. Vitanyi, Three approaches to the quantitative definition of information in an individual pure quantum state, quant-ph/9907035 (2000).
4. A. Berthiaume, W. van Dam and S. Laplante, Quantum Kolmogorov complexity, quant-ph/005018 (2000).
5. P. Gács, Quantum algorithmic entropy, quant-ph/0011046 v2 (2001).
6. C. E. Mora and H. J. Briegel, Algorithmic complexity and entanglement of quantum states, *Phys. Rev. Lett.* **95** (2005) 20.
7. N. Bohr, *Atomic Physics and Human Knowledge* (Wiley, New York, 1958).
8. M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, 2000).
9. G. J. Chaitin, *Information, Randomness and Incompleteness* (World Scientific, 1987).
10. J. J. Vartiainen, M. Möttönen and M. M. Salomaa, Efficient decomposition of quantum gates, *Phys. Rev. Lett.* **92** (2004) 17.
11. A. Y. Kitaev, Quantum computations: Algorithms and error correction, *Russ. Math. Surv.* **52** (1997) 1191.
12. M. Li and P. Vitanyi, *An Introduction to Kolmogorov Complexity and Its Applications* (Springer, 1997).
13. P. Grünwald and P. Vitanyi, Shannon information and Kolmogorov complexity, cs.IT/0410002v1 (2004).
14. B. Schumacher, Quantum coding, *Phys. Rev. A* **51** (1995) 4.
15. B.-G. Englert, Remark on some basic issues in quantum mechanics, *Z. Naturforsch.* **54a** (1999) 11.
16. A. S. Holevo, Bounds for the quantity of information transmitted by a quantum communication channel, *Probl. Inform. Trans.* **9** (1973) 177.
17. M. Hein, J. Eisert and H. J. Briegel, Multi-party entanglement in graph states, *Phys. Rev. A* **69** (2004) 6.
18. W. Dür, L. Hartmann, M. Hein and H. J. Briegel, Entanglement in spin chains and lattices with long-range Ising-type interactions, *Phys. Rev. Lett.* **94** (2005) 9.

19. J. Calsamiglia, L. Hartmann, W. Dür and H.-J. Briegel, Spin gases: Quantum entanglement driven by classical kinematics, *Phys. Rev. Lett.* **95** (2005) 18.
20. J. Eisert and H. J. Briegel, Schmidt measure as a tool for quantifying multiparticle entanglement, *Phys. Rev. A* **64** (2001) 022306.
21. J. I. Latorre and R. Orus, Adiabatic quantum computation and quantum phase transitions, *Phys. Rev. A* **69** (2004) 062302.