Dense Quantum Coding and a Lower Bound for 1-way Quantum Automata

Andris Ambainis * *Ashwin Nayak* † *Amnon Ta-Shma* ‡ *Umesh Vazirani* §

Abstract

We consider the possibility of encoding m classical bits into much fewer n quantum bits so that an arbitrary bit from the original m bits can be recovered with a good probability, and we show that non-trivial quantum encodings exist that have no classical counterparts. On the other hand, we show that quantum encodings cannot be much more succint as compared to classical encodings, and we provide a lower bound on such quantum encodings. Finally, using this lower bound, we prove an exponential lower bound on the size of 1-way quantum finite automata for a family of languages accepted by linear sized deterministic finite automata.

1 Introduction

The tremendous information processing capabilities of quantum mechanical systems may be attributed to the fact that the state of an n quantum bit (qubit) system is given by a unit vector in a 2^n dimensional complex vector space. Can this fact - that $2^n - 1$ complex numbers are necessary to completely specify the state of n quantum bits- be used to encode and transmit classical information with exponentially fewer qubits. A fundamental result in quantum information theory—Holevo's theorem [9]—states that no more than n classical bits of information can be transmitted by transferring n quantum bits from one party to another. In view of this result, it is tempting to conclude that the exponentially many degrees of freedom latent in the description of a quantum system must necessarily stay hidden or inaccessible.

*Computer Science Division, UC Berkeley. Email: ambainis@cs.berkeley.edu. Supported by Berkeley Fellowship for Graduate Studies and, in part, by NSF grant CCR-9800024.

†Computer Science Division, UC Berkeley. Email: ashwin@cs.berkeley.edu. Supported by JSEP grant FDF 49620-97-1-0220-03-98 and NSF grant CCR-9800024.

‡International Computer Science Institute, Berkeley. Email: amnon@icsi.berkeley.edu. Supported by NSF grant CCR-9800024.

§Computer Science Division, UC Berkeley. Email: vazirani@cs.berkeley.edu. Supported by JSEP grant FDF 49620-97-1-0220-03-98 and NSF grant CCR-9800024.

STOC '99 Atlanta GA USA

However, the situation is more subtle since in quantum mechanics, the recipient of the n qubits has a choice of measurements he can make to extract information about their state. In general, these measurements do not commute. Thus making a particular measurement will, in general, disturb the system, thereby destroying some or all the information that would have been revealed by another possible measurement. This opens up the possibility of quantum *random access* encodings. Say we wish to encode m classical bits $b_1 \cdots b_m$ into n quantum bits ($m \gg n$). Then a quantum *random access* encoding with parameters m, n, p (or simply an $m \overset{p}{\mapsto} n$ encoding) consists of an encoding map from $\{0,1\}^m$ to $\mathbb{C}^{2^n}$, together with a sequence of m possible measurements for the recipient. The encoding has a success probability p if for any i, if the recipient chooses the ith measurement and applies it to the encoding of $b = b_1 \ldots b_m$, the result of the measurement is b_i with probability at least p.

Definition 1.1 *A* $m \overset{p}{\mapsto} n$ random access *encoding is a function* $f : \{0,1\}^m \times R \mapsto \mathbb{C}^{2^n}$ *such that for every* $1 \leq i \leq m$*, there is a measurement* $\mathcal{O}_i$ *that returns* 0 *or* 1 *and has the property that*

$$\forall b \in \{0,1\}^m : \ \mathrm{Prob}(\ \mathcal{O}_i\,|f(b,r)\rangle = b_i\ ) \ \geq\ p.$$

We call f *the* encoding function, *and* $\mathcal{O}_i$ *the* decoding functions.

Notice that random access encodings with $m \gg n$ and $p > 1/2$ does not necessarily violate Holevo's bound, since the m possible measurements may be non-commuting. Thus, the recipient cannot make all of them to recover all the encoded bits. Indeed, there is no a priori reason to rule out the existence of a $c^n \overset{p}{\mapsto} n$ encoding for constants $c > 1$, $p > 1/2$. In fact, even though $\mathbb{C}^k$ can accommodate only k mutually orthogonal unit vectors, it can accommodate c^k almost mutually orthogonal unit vectors (i.e. vectors such that the dot product of any two has absolute value less than 1/10, say). This might lead one to believe that such encodings exist. If such quantum random access encodings were possible, it would be possible to, for instance, encode the contents of an entire telephone directory in a few quantum bits such that the recipient of these qubits could, via a suitably chosen measurement, look up any *single* telephone number of his choice. Also, this would have implied $IP \subseteq QuantumNP$

since one could encode an exponential size proof into a polynomial number of qubits.

The main question that we consider in this paper is: for what values of m, n and p do $m \overset{p}{\mapsto} n$ encodings exist? For classical encodings, where we encode m classical bits into n classical bits, we know the answer. Let, for $p \in [0,1]$, $H(p) = -p\log p - (1-p)\log(1-p)$ denote the *binary entropy function.* We show:

Theorem 1.1 *For any $p > 1/2$, there exist $m \overset{p}{\mapsto} n$ classical encodings with $n = (1-H(p))m + O(\log m)$, and any $m \overset{p}{\mapsto} n$ classical encoding has $n \geq (1-H(p))m$.*

We then show that quantum encodings are more powerful than classical encodings. On the one hand, we show that no classical encoding can encode two bits into one bit with decoding success probability greater than 0.5, and on the other hand, we exhibit a $2 \overset{0.85}{\mapsto} 1$ quantum encoding. In fact, as Ike Chuang [5] has shown, it is possible to encode 3 bits into 1 qubit with success probability ≈ 0.79 by taking advantage of the fact that the amplitudes in quantum states can be complex numbers. The 2-into-1 quantum encoding and the 3-into 1 encoding easily generalize to a $2n \overset{0.85}{\mapsto} n$ and a $3n \overset{0.79}{\mapsto} n$ encoding, respectively. However, the question as to whether quantum encodings can *asymptotically* beat the classical lower bound of Theorem 1.1 is left open. Our main result about quantum encodings is that they cannot be much smaller than the encoded strings.

Theorem 1.2 *If a $m \overset{p}{\mapsto} n$ quantum encoding exists with $p > \frac{1}{2}$ a constant, then $n \geq \Omega(\frac{m}{\log m})$.*

Thus, even though quantum random access encodings can beat classical encodings, they cannot be much more succinct.

We finish the paper with a novel application of our lower bound on quantum random access codes to showing a lower bound on the size of 1-way quantum finite automata (QFAs). (See Section 5.1 for a precise definition of 1-way QFAs.) In [10] it was shown that not every language recognized by a (classical) deterministic finite automaton (DFA) can be recognized by a 1-way QFA. On the other hand, there are languages that can be recognized by 1-way QFAs with size exponentially smaller than that of corresponding classical automata [2]. It remained open whether, for any language that can be recognized by a 1-way finite automaton both classically and quantum-mechanically, we can efficiently simulate the classical automaton by a 1-way QFA. Our result answers this question in the negative, and demonstrates that while in some cases one is able to exploit quantum phenomena to construct highly space-efficient 1-way QFAs, in others, as it will become apparent, the requirement of the unitarity (or, in other words, reversibility) of evolution seriously limits their efficiency.

Theorem 1.3 *Let $\{L_n\}_{n\geq 1}$ be a family of languages defined by $L_n = \{wa \mid w \in \{a,b\}^*, |w| \leq n\}$. Then,*

1. *L_n is recognized by a 1-way deterministic automaton of size $O(n)$,*
2. *L_n is recognized by some 1-way quantum finite automaton, and,*
3. *Any 1-way quantum automaton recognizing L_n with some constant probability greater than $\frac{1}{2}$ has $2^{\Omega(n/\log n)}$ states.*

The lower bound on quantum random access codes plays the following role in this context: For this language, a quantum automaton has to remember every bit of the input because of the reversibility requirement. If exponentially dense quantum random access codings were possible, then the QFA might be able to store this information space-efficiently. Thus, the lower bound on quantum random access codings plays a crucial role in the lower bound on QFAs.

2 The classical bounds

We first prove a lower bound on the number of bits required for a *classical* random access encoding, and then show that there are classical encodings that nearly achieve this bound. Together, these yield Theorem 1.1 of the previous section.

The proof of the lower bound involves the concepts of the *Shannon entropy* $S(X)$ of a random variable X, the Shannon entropy $S(X|Y)$ of a random variable X *conditioned on* another random variable Y, and the *mutual information* $I(X:Y)$ of a pair of random variables X, Y. For definitions and basic facts involving these concepts, we refer the reader to a standard text (such as [7]) on information theory.

Theorem 2.1 *Let $1/2 < p \leq 1$. For any classical $m \overset{p}{\mapsto} n$ encoding, $n \geq (1-H(p))m$.*

Proof: Suppose there is such a (possibly probabilistic) encoding f. Let $X = X_1 \cdots X_m$ be chosen uniformly at random from $\{0,1\}^m$, and let $Y = f(X) \in \{0,1\}^n$ be the corresponding encoding. Let Z be the random variable with values in $\{0,1\}^m$ obtained by generating the bits $Z_1 \cdots Z_m$ from Y using the m decoding functions.

The mutual information of X and Y is clearly bounded by the number of bits in Y, i.e. n:

$$I(X:Y) \leq S(Y) \leq n.$$

We show below that it is, in fact, lower bounded by $(1-H(p))m$, thus getting our lower bound.

Now,

$$I(X:Y) = S(X) - S(X|Y) = m - S(X|Y).$$

But, using standard properties of the entropy function, we have

$$S(X|Y) \leq S(X|Z) \leq \sum_{i=1}^{m} S(X_i|Z) \leq \sum_{i=1}^{m} S(X_i|Z_i).$$

It is not difficult to see that $S(X_i|Z_i) \leq H(p)$. It follows that $S(X|Y) \leq H(p)m$, and that $I(X:Y) \geq (1-H(p))m$, as we intended to show. ■

We now give an almost matching upper bound:

Theorem 2.2 *There is a classical $m \overset{p}{\mapsto} n$ encoding with $n = (1 - H(p))m + O(\log m)$ for any $p > \frac{1}{2}$.*

Proof: The encoding is trivial for $p > 1 - \frac{1}{m}$. We describe the encoding for $p \leq 1 - \frac{1}{m}$ below.

We use a code $S \subseteq \{0,1\}^m$ such that, for every $x \in \{0,1\}^m$, there is a $y \in \{0,1\}^m$ within Hamming distance $(1-p-\frac{1}{m})m$. It is known (see, e.g., [6]) that there is such a code S of size

$$|S| = 2^{(1-H(p+\frac{1}{m}))m+2\log m} \leq 2^{(1-H(p))m+4\log m}.$$

Let $S(x)$ denote the codeword closest to x. One possibility is to encode a string x by $S(x)$. This would give us an encoding of the right size. Further, for every x, at least $(p+\frac{1}{m})m$ out of the m bits would be correct. This means that the probability (over all bits i) that $x_i = S(x)_i$ is at least $p+1/m$. However, for our encoding we need this probability to be at least p for *every* bit, not just on average over all bits. This can be achieved with the following modification.

Let r be an m-bit string, and π be a permutation of $\{1,\ldots,m\}$. For a string $x \in \{0,1\}^m$, let $\pi(x)$ denote the string $x_{\pi(1)}x_{\pi(2)}\cdots x_{\pi(m)}$.

We consider encodings $S_{\pi,r}$ defined by $S_{\pi,r}(x) = \pi^{-1}(S(\pi(x+r)) + r$. We show that if π and r are chosen uniformly at random, then for any x and any index i, the probability that the ith bit in the encoding is different from x_i is at most $1-p-1/m$. First, note that if i is also chosen uniformly at random, then this probability is clearly bounded by $1-p-1/m$. So all we need to do is to show that this probability is independent of i.

If π and r are uniformly random, then $\pi(x+r)$ is uniformly random as well. Furthermore, for a fixed $y = \pi(x+r)$, there is exactly one r corresponding to any permutation π that gives $y = \pi(x+r)$. Hence, if we condition on $y = \pi(x+r)$, all π (and, hence, all $\pi^{-1}(i)$) are equally likely. This means that the probability that $x_i \neq S_{\pi,r}(x)_i$ (or, equivalently, that $\pi(x+r)_{\pi^{-1}(i)} \neq (S(\pi(x+r))_{\pi^{-1}(i)})$ for random π and r is just the probability of $y_j \neq S(y)_j$ for random y and j. This is clearly independent of i (and x).

Finally, we show that there is a small set of permutation-string pairs such that the desired property continues to hold if we choose π, r uniformly at random from *this* set, rather than the entire space of permutations and strings. We employ the probabilistic method to prove the existence of such a small set of permutation-string pairs.

Let $\ell = m^3$, and let the strings $r_1,\ldots,r_\ell \in \{0,1\}^m$ and permutations $\pi_1,\ldots,\pi_\ell$ be chosen independently and uniformly at random. Fix $x \in \{0,1\}^m$ and $i \in [1..m]$. Let X_j be 1 if $x_i \neq S_{\pi_j,r_j}(x)_i$ and 0 otherwise. Then $\sum_{j=1}^{\ell} X_j$ is a sum of ℓ independent Bernoulli random variables, the mean of which is at most $(1-p-1/m)\ell$. Note that $\frac{1}{\ell}\sum_{j=1}^{\ell} X_j$ is the probability of encoding the ith bit of x erroneously when the permutation-string pair is chosen uniformly at random from the set $\{(\pi_1,r_1),\ldots(\pi_\ell,r_\ell)\}$. By the Chernoff bound, the probability that the sum $\sum_{j=1}^{\ell} X_j$ is at least $(1-p-1/m)\ell + m^2$ (i.e., that the error probability $\frac{1}{\ell}\sum_{j=1}^{\ell} X_j$ mentioned above is at least $1-p$) is bounded by $e^{-2m^4/\ell} = e^{-2m}$. Now, the union bound implies that the probability that the ith bit of x is encoded erroneously with probability more than $1-p$ for *any* x or i is at most $m2^m e^{-2m} < 1$. Thus, there is a combination of strings $r_1,\ldots,r_\ell$ and permutations $\pi_1,\ldots,\pi_\ell$ with the property we seek. We fix such a set of ℓ strings and permutations.

We can now define our random access code as follows. To encode x, we select $j \in \{1,\ldots,\ell\}$ uniformly at random and compute $y = S_{\pi_j,r_j}(x)$. This is the encoding of x. To decode the ith bit, we just take y_i. For this scheme, we need $\log(\ell|S|) = \log\ell + \log|S| = (1-H(p))m + 7\log m$ bits. This completes the proof of the theorem. ■

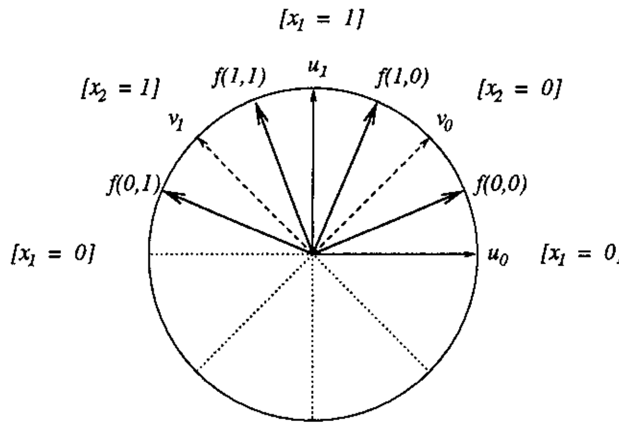

Figure 1: *A 2-into-1 quantum encoding with probability of success* ≈ 0.85.

3 A gap between quantum and classical encodings

In this section, we construct a quantum encoding that has no classical counterpart.

Lemma 3.1 *There is a $2 \overset{0.85}{\mapsto} 1$ quantum encoding.*

Proof: Let $u_0 = |0\rangle$, $u_1 = |1\rangle$, and $v_0 = \frac{1}{\sqrt{2}}(|1\rangle + |0\rangle)$, $v_1 = \frac{1}{\sqrt{2}}(|1\rangle - |0\rangle)$. Define $f(x_1,x_2)$, the encoding of the string x_1x_2 to be $u_{x_1} + v_{x_2}$ normalized (See Figure 1). The decoding functions are defined as follows: for the first bit x_1, we measure the message qubit according to the u basis and associate u_0 with $x_1 = 0$ and u_1 with $x_1 = 1$. Similarly, for the second bit, we measure according to the v basis, and associate v_0 with $x_2 = 0$ and v_1 with $x_2 = 1$.

It is easy to verify that for all four codewords, and for any $i = 1,2$, the angle between the codeword and the right subspace is $\pi/8$. Hence the success probability is $\cos^2(\pi/8) \approx 0.853$. ■

Lemma 3.2 *No $2 \overset{p}{\mapsto} 1$ classical encoding exists for any $p > \frac{1}{2}$.*

Proof: Suppose there is a classical $2 \overset{p}{\mapsto} 1$ encoding for some $p > \frac{1}{2}$. Let $f : \{0,1\}^2 \times R \mapsto \{0,1\}$ be the corresponding probabilistic encoding function and $V_i : \{0,1\} \times R' \mapsto$

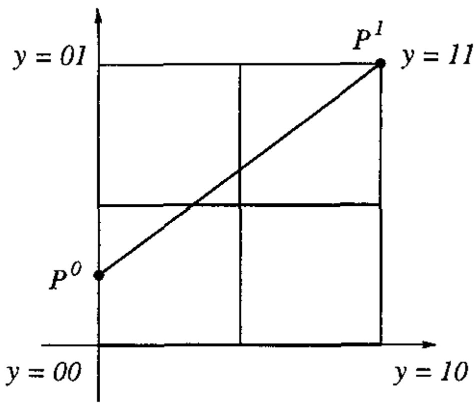

Figure 2: *A geometric characterization of the probabilistic decoding functions of Lemma 3.2.*

$\{0,1\}$ the probabilistic decoding functions. If we let y_i be the random variable $V_i(f(x,r),r')$, then for any $x \in \{0,1\}^2$, and any $i \in \{1,2\}$, $\mathrm{Prob}_{r,r'}(y_i = x_i) \geq p$.

We first give a geometric characterization of the decoding functions. Each V_i clearly depends only on the encoding, which is either 0 or 1. Define the point P^j (for $j = 0,1$) in the unit square $[0,1]^2$ as $P^j = (a_0^j, a_1^j)$, where $a_i^j = \mathrm{Prob}_{r'}(\ V_i(j,r') = 1\ )$. The point P^0 characterizes the decoding functions when the encoding is 0, and P^1 characterizes the decoding functions when the encoding is 1. For example, $P^1 = (1,1)$ means that given the encoding 1, the decoding functions return $y_1 = 1$ and $y_2 = 1$ with certainty, and $P^0 = (0, 1/4)$ means that given the encoding 0, the decoding functions return $y_1 = 0$ and, with probability 1/4, that $y_2 = 1$.

Any string $x = x_1x_2 \in \{0,1\}^2$ is encoded as a 0 with some probability p_x and as a 1 with some probability $1-p_x$. If we let $P^x = (a_0^x, a_1^x)$, where a_i^x is the probability that $y_i = 1$, then $P^x = p_xP^0 + (1-p_x)P^1$. Thus, P^x lies on the line connecting the two points P^0 and P^1. On the other hand, for the encoding to be a valid 2-into-1 encoding, the point P^x should lie *strictly* inside the quarter of the unit square $[0,1]^2$ closest to (x_1, x_2).

Now, the line connecting P^0 and P^1 intersects the interiors of only three of the four quarters of the unit square $[0,1]^2$. For instance, if P^0 and P^1 are as above, then the line connecting them does not pass through the lower right quarter (see Figure 2). Thus, for the string x_1x_2 which is favored by that quarter (e.g. the string $x = 10$ in the example above), either V_1 or V_2 errs with probability at least a half—which is a contradiction. ∎

4 The quantum lower bound

We now prove Theorem 1.2. We first show that the success probability of the decoding process can be amplified at the cost of a small increase in the length of the random access code.

Lemma 4.1 *If for a constant $p > \frac{1}{2}$ there is an $m \overset{p}{\mapsto} n$ encoding, then there is also an $m \overset{1-\epsilon}{\mapsto} O(n \log \frac{1}{\epsilon})$ encoding for any $\epsilon = \epsilon(m) > 0$.*

Proof: Suppose there is an encoding $f : \{0,1\}^m \times R \mapsto \mathbb{C}^{2^n}$ with decoding algorithms $\mathcal{O}_i$ $(i = 1,\ldots,m)$ with success probability $p > 1/2$. We define a new encoding $f^{(t)} : \{0,1\}^m \times R^t \mapsto (\mathbb{C}^{2^n})^t$ as $f^{(t)}(x, r_1, \ldots, r_t) = f(x,r_1) \otimes \cdots \otimes f(x,r_t)$. I.e., it is the tensor product of t independent identical copies of the original code. The new decoding functions $\mathcal{O}'_i$ consists of applying $\mathcal{O}_i$ to each of the t independent copies of the code, and answering according to the majority. The Chernoff bound shows that the error probability decays exponentially fast in the number of trials, and is therefore at most ϵ when t is chosen to be $O(\log \frac{1}{\epsilon})$. ∎

By choosing $\epsilon = 1/q(m)$ for some polynomial q, we achieve an encoding with error ϵ at the cost of using an $O(\log m)$ factor more qubits for the encoding. Now the result of any measurement cannot perturb the state vector too much (i.e. by more than $\sqrt{\epsilon}$). It might seem that this is sufficient to give us the lower bound, since we need to make only m measurements to recover all m encoded bits, and the error per measurement is only $1/\mathrm{poly}(m)$. However, the situation is more subtle, since the error on subsequent measurements must take into account both the encoding error, as well as the error introduced by previous measurements. In fact, a straightforward analysis suggests that the error doubles with each measurement, thus making such a proof infeasible. Instead, we prove that the errors grow linearly (rather than exponentially), by first invoking the principle of safe storage (see [4]) to defer all measurements to the end of a sequence of unitary operations, and then bounding the errors in the computation via a hybrid technique from [3] (which is made more explicit in [13]).

Lemma 4.2 *If a $m \overset{1-\epsilon}{\mapsto} n$ quantum encoding with $\epsilon \leq \frac{1}{64m^2}$ exists, then $n \geq \Omega(m)$.*

Proof: We first deal with *deterministic* quantum encoding, in which the encoding function $f : \{0,1\}^m \mapsto \mathbb{C}^{2^n}$ maps inputs to *pure* states. Any such encoding has, for every $i \in [1..m]$, a decoding function which takes a codeword $|\phi\rangle$ and an ancilla $|0^l\rangle$, applies a unitary transformation V_i, and makes a measurement. Thus, it resolves $\mathbb{C}^{2^n}$ into two subspaces W_i^0 and $(W_i^0)^\perp$ corresponding to the answers 0 and 1 (for the ith bit), respectively. Given $|\phi, 0^l\rangle$, we can thus decompose it as $|\phi_i^0\rangle + |\phi_i^1\rangle$, where $|\phi_i^0\rangle \in W_i^0$ and $|\phi_i^1\rangle \in (W_i^0)^\perp$.

We now apply the principle of safe storage. Instead of applying V_i and measuring, we use unitary transformations U_i $(i = 1,\ldots,m)$ that work over the codeword $|\phi\rangle$, the ancilla $|0^l\rangle$ and m output bits $|0^m\rangle$, such that $U_i|\phi_i^0, a\rangle = |\phi_i^0, a\rangle$ and $U_i|\phi_i^1, a\rangle = |\phi_i^1, a \oplus e_i\rangle$, where e_i is the vector $|0,\ldots,0,1,0,\ldots 0\rangle$ having a 1 entry only in the ith place.

The transformations U_i introduce some garbage at each step, and their composition $U_1 \cdots U_m$ is quite messy. To analyse their behavior, we first fix an input x, and imagine ideal unitary transformations $U'_i = U'_i(x)$ that have the property that for the codeword $|\phi_x\rangle$ of x, $U_i' |\phi_x, a\rangle = |\phi_x, a \oplus (x_i \cdot e_i)\rangle$. Since for any $x \in \{0,1\}^m$ and any $i \in [1..m]$, the transformation U_i correctly yield the ith bit of x with high probability, the reader can verify that

$$\left\| U_i \left|\phi_x, 0^l, a\right\rangle - U_i' \left|\phi_x, 0^l, a\right\rangle \right\|^2 \leq 2\epsilon. \quad (1)$$

We now claim that the result of applying the transformations U_i does not differ much from that of applying the ideal transformations U_i'.

Claim 4.1

$\left\| U_1 \cdots U_m \left|\phi_x, 0^l, 0^m\right\rangle - U_1' \cdots U_m' \left|\phi_x, 0^l, 0^m\right\rangle \right\| \leq 2m\sqrt{\epsilon}$.

Proof: We use a hybrid argument:

$$\left\| U_1 \cdots U_m \left|\phi_x, 0^l, 0^m\right\rangle - U_1' \cdots U_m' \left|\phi_x, 0^l, 0^m\right\rangle \right\| \leq$$

$$\left\| U_1 \cdots U_{m-1} U_m \left|\phi_x, 0^l, 0^m\right\rangle - U_1 \cdots U_{m-1} U_m' \left|\phi_x, 0^l, 0^m\right\rangle \right\| +$$

$$\left\| U_1 \cdots U_{m-1} U_m' \left|\phi_x, 0^l, 0^m\right\rangle - U_1 \cdots U_{m-1}' U_m' \left|\phi_x, 0^l, 0^m\right\rangle \right\| +$$

$$\cdots +$$

$$\left\| U_1 \cdots U_{m-1}' U_m' \left|\phi_x, 0^l, 0^m\right\rangle - U_1' \cdots U_{m-1}' U_m' \left|\phi_x, 0^l, 0^m\right\rangle \right\|$$

But, since the transformations U_i are unitary, we have:

$$\left\| U_1 \cdots U_t U_{t+1}' \cdots U_m' \left|\phi_x, 0^l, 0^m\right\rangle - \right.$$

$$\left. U_1 \cdots U_t' U_{t+1}' \cdots U_m' \left|\phi_x, 0^l, 0^m\right\rangle \right\| =$$

$$\left\| U_t U_{t+1}' \cdots U_m' \left|\phi_x, 0^l, 0^m\right\rangle - U_t' U_{t+1}' \cdots U_m' \left|\phi_x, 0^l, 0^m\right\rangle \right\| =$$

$$\left\| U_t \left|\phi_{t+1}'\right\rangle - U_t' \left|\phi_{t+1}'\right\rangle \right\|,$$

where $|\phi_{t+1}'\rangle = U_{t+1}' \cdots U_m' \left|\phi_x, 0^l, 0^m\right\rangle$. By the definition of the transformations U_i', $|\phi_{t+1}'\rangle = \left|\phi_x, 0^l, a\right\rangle$ with $a = |0, \ldots, 0, x_{t+1}, \ldots, x_m\rangle$. Hence, by equation (1), $\| U_t |\phi_{t+1}'\rangle - U_t' |\phi_{t+1}'\rangle \| \leq 2\sqrt{\epsilon}$, and the claimed result follows. ∎

Now we can extract all the bits of x by computing $|\psi\rangle = U_1 \ldots U_m \left|\phi_x, 0^l, 0^m\right\rangle$ and measuring the m answer bits $a_1, \ldots, a_m$. The following claim says that we succeed with high probability.

Claim 4.2 $\mathrm{Prob}(a \neq x) \leq 4m\sqrt{\epsilon}$.

Proof: Let $|\psi'\rangle = U_1' \ldots U_m' \left|\phi_x, 0^l, 0^m\right\rangle = \left|\phi_x, 0^l, x\right\rangle$. From the claim above, we know that $\| |\psi\rangle - |\psi'\rangle \| \leq 2m\sqrt{\epsilon}$. When we measure the answer bits of $|\psi'\rangle$, we get x with probability 1. Moreover, from the following fact, the probability of observing x on measuring $|\psi\rangle$ cannot differ from this by very much.

Fact 4.1 *Suppose* $\| |\psi_1\rangle - |\psi_2\rangle \| \leq \delta$. *Let* $\mathcal{O}$ *be a measurement with possible results* Λ, *and* $\mathcal{D}_i$ *the classical distributions over* Λ *that result from applying* $\mathcal{O}$ *to* $|\psi_i\rangle$. *Then* $\| \mathcal{D}_1 - \mathcal{D}_2 \|_1 \stackrel{\text{def}}{=} \Sigma_{a \in \Lambda} |\mathcal{D}_1(a) - \mathcal{D}_2(a)| \leq 2\delta$.

Hence, the probability that $a \neq x$ is at most $4m\sqrt{\epsilon}$. ∎

Therefore, we get x with probability at least $1 - 4m\sqrt{\epsilon} \geq 1 - \frac{4m}{8m} = \frac{1}{2}$. It then follows from Holevo's Theorem [9] that $n \geq \Omega(m)$.

Now we deal with *probabilistic* quantum encoding, where we can encode a string $x \in \{0,1\}^m$ as a probabilistic mixture of pure states. It is well known that we can always *purify* the system, i.e., we can adjoin ancilla bits to the encoding, such that the result is a pure state. Now, as before, we may apply the decoding transformations U_i and retrieve all the encoded bits: for every x, there are ideal transformations $U_i' = U_i'(x)$ that behave almost as U_i (in the same sense as above) and the same argument again gives us the lower bound on n. ∎

Combining the two lemmas above, we get Theorem 1.2. We remark that we may extend this lower bound to general $p > 1/2$, by appropriately generalizing Lemma 4.1 above.

4.1 Serial encodings

We note that Theorem 1.2 holds even in a slightly more general scenario, when the decoding functions are allowed to depend on the string encoded.

Definition 4.1 $f : \{0,1\}^m \times R \mapsto \mathcal{C}^{2^n}$ *serially encodes* m *classical bits into* n *qubits with* p *success, if for any* $i \in [1..n]$ *and* $b_{[i+1,n]} = b_{i+1} \cdots b_n \in \{0,1\}^{n-i}$, *there is a measurement* $\mathcal{O}_{i, b_{[i+1,n]}}$ *that returns* 0 *or* 1 *and has the property that*

$$\forall b \in \{0,1\}^m : \ \mathrm{Prob}(\ \mathcal{O}_{i,b_{[i+1,n]}} |f(b,r)\rangle \ = \ b_i\ ) \geq p.$$

I.e., we allow the decoding functions to depend on the suffix $b_{i+1} \cdots b_m$ of the string b for recovering the value of the ith bit b_i. The lower bound for quantum random access codes of the previous section also holds for serial encodings.

Theorem 4.1 *Any quantum serial encoding of* m *bits into* n *qubits with constant success probability* $p > \frac{1}{2}$ *has* $n \geq \Omega(\frac{m}{\log m})$.

Proof: On careful examination, we see that for the proof of Theorem 1.2 to work in this case as well, all we need to check is that for all $i \in [1..n]$,

$$\left\| U_i \left|\phi_x, 0^l, a_i\right\rangle - U_i' \left|\phi_x, 0^l, a_i\right\rangle \right\|^2 \leq 2\epsilon,$$

where $a_i = |0, \ldots, 0, x_{i+1}, \ldots, x_m\rangle$. Although the transformations U_i may now depend on the bits already decoded, the above bound is easily verified, since a_i contains the required suffix of the encoded word x. ∎

5 The lower bound for 1-way quantum finite automata

In this section, we give the details of the proof of Theorem 1.3. The first two parts of Theorem1.3 are easy. Figure 3 shows a DFA with $2n+3$ states for the language L_n.

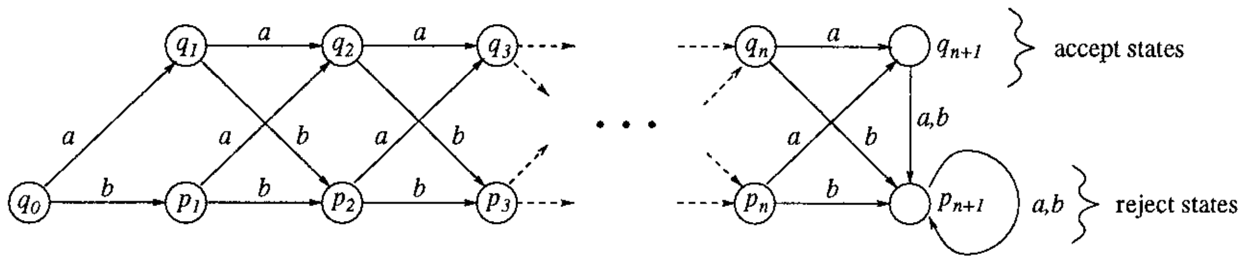

Figure 3: *A DFA that accepts the language* $L_n = \{wa \mid w \in \{a,b\}^*, |w| \leq n\}$.

Also, since each L_n is a finite language, there is a 1-way *reversible* finite automaton (as defined in Section 5.1), and hence a 1-way QFA that accepts it. What then remains to be shown is the lower bound on the size of a 1-way QFA accepting the language.

Intuitively, since a 1-way QFA is allowed to read input symbols only once, a QFA for L_n necessarily "records" the last symbol read in its state, and since it is required to be reversible, it is forced to "remember" *all* the symbols read until it is clear whether the input is in the language or not. Thus, we expect the state of the automaton after n input symbols to be an *encoding* of the n symbols. It is not difficult to see that in the case of a 1-way reversible automaton that accepts the language L_n, the encoding is such that all the n input symbols can be recovered with certainty. Thus, such an automaton has at least 2^n states. However, for reasons stated below, it is not clear in the case of a *general* 1-way QFA that the state encodes the input symbols in a "faithful" manner.

- Firstly, a 1-way QFA is allowed to make *partial decisions* (i.e., it is allowed to accept or reject an input with some probability before reading all its symbols). We show in Section 5.3 that partial decisions can be "deferred" for r steps at a cost of only an $O(r)$ factor increase in the size of the automaton. We call the resulting automaton an *r-restricted* QFA. Since no input of length more than $n+1$ belongs to L_n, this means that partial decisions are not very useful in building "small" automata for the language, and that we can limit our study to that of n-restricted QFAs.

- Secondly, and more seriously, the encoding defined by the automaton is such that each input symbol is accessible via a measurement only when all the symbols following it are known, and by trying to learn the later symbols we might destroy the encoding.

  This problem is exactly the one Theorem 4.1 solves. We can thus conclude that the number of qubits required to represent a state of the automaton is $\Omega(n/\log n)$, which gives us the lower bound stated in Theorem 1.3.

Before presenting the formal proof for the lower bound, we define 1-way QFAs precisely in the next section. We then show how a restricted QFA for the language L_n yields a serial encoding of n classical bits into a state of the automaton. Theorem 4.1 then immediately gives a size lower bound of $2^{\Omega(n/\log n)}$ for restricted QFAs. We then extend this lower bound to general QFAs in Section 5.3.

5.1 Technical preliminaries

A 1-way quantum finite automaton (QFA) is a theoretical model for a quantum computer with finite memory. It has a finite set of basis states Q, which consists of three parts: accepting states, rejecting states and non-halting states. The sets of accepting, rejecting and non-halting basis states are denoted by $Q_{\rm acc}, Q_{\rm rej}$ and $Q_{\rm non}$, respectively. One of the states, q_0, is distinguished as the starting state.

Inputs to a QFA are words over a finite alphabet Σ. We shall also use the symbols '¢' and '\$' that do not belong to Σ to denote the left and the right end marker, respectively. The set $\Gamma = \Sigma \cup \{¢, \$, \}$ denotes the working alphabet of the QFA. For each symbol $\sigma \in \Gamma$, a 1-way QFA has a corresponding unitary transformation U_σ on the space $\mathbb{C}^Q$. A 1-way QFA is thus defined by describing $Q, Q_{\rm acc}, Q_{\rm rej}, Q_{\rm non}, q_0, \Sigma$, and U_σ for all $\sigma \in \Gamma$. We will often refer to 1-way QFAs as simply QFAs, since we do not consider any other type of QFAs in this paper.

At any time, the state of a QFA is a superposition of basis states in Q. The computation starts in the superposition $|q_0\rangle$. Then transformations corresponding to the left end marker '¢,' the letters of the input word x and the right end marker '\$' are applied in succession to the state of the automaton, unless a transformation results in acceptance or rejection of the input. A transformation corresponding to a symbol $\sigma \in \Gamma$ consists of two steps:

1. First, U_σ is applied to $|\psi\rangle$, the current state of the automaton, to obtain the new state $|\psi'\rangle$.

2. Then, $|\psi'\rangle$ is measured with respect to the observable $E_{\rm acc} \oplus E_{\rm rej} \oplus E_{\rm non}$, where $E_{\rm acc} = {\rm span}\{|q\rangle \mid q \in Q_{\rm acc}\}$, $E_{\rm rej} = {\rm span}\{|q\rangle \mid q \in Q_{\rm rej}\}$, $E_{\rm non} = {\rm span}\{|q\rangle \mid q \in Q_{\rm non}\}$. The probability of observing E_i is equal to the squared norm of the projection of $|\psi'\rangle$ onto E_i. On measurement, the state of the automaton "collapses" to the projection onto the space observed, i.e., becomes equal to the projection, suitably normalized to a unit superposition.

   If we observe $E_{\rm acc}$ (or $E_{\rm rej}$), the input is accepted (or rejected). Otherwise, the computation continues, and the next transformation, if any, is applied.

We regard these two steps together as reading the symbol σ.

A QFA M is said to *accept* (or *recognize*) a language L with probability $p > \frac{1}{2}$ if it accepts every word in L with probability at least p, and rejects every word not in L with probability at least p.

A *reversible finite automaton* (RFA) is a QFA such that, for any $\sigma \in \Gamma$ and $q \in Q$, $U_\sigma |q\rangle = |q'\rangle$ for some $q' \in Q$. In other words, the operator U_σ is a permutation over the basis states; it maps each basis state to a basis state, not to a superposition over several states.

The *size* of a finite automaton is defined as the number of (basis) states in it. The "space used by the automaton" refers to the number of (qu)bits required to represent an arbitrary automaton state.

5.2 The lower bound for restricted QFAs

Define an *r-restricted* 1-way QFA for a language L as a 1-way QFA that recognizes the language with probability $p > \frac{1}{2}$, and which halts with non-zero probability before seeing the right end marker only *after* it has read r letters of the input. We first show a lower bound on the size of n-restricted 1-way QFAs that accept L_n.

Let M be any n-restricted 1-way QFA accepting L_n with constant probability $p > \frac{1}{2}$. The following claim formalizes the intuition that the state of M after n symbols of the input have been read is an encoding of the input string.

Claim 5.1 *There is a serial encoding of n bits into $\mathbb{C}^Q$, and hence into* $\lceil \log |Q| \rceil$ qubits, *where Q is the set of basis states of the QFA M.*

Proof: Let Q be the set of basis states of the QFA M, and let $Q_{\rm acc}$ and $Q_{\rm rej}$ be the set of accepting and rejecting states respectively. Also, let U_σ be the unitary operator of M corresponding to the symbol $\sigma \in \{a, b, ¢, \$\}$. Let $E_{\rm acc}$, $E_{\rm rej}$ and $E_{\rm non}$ be defined as in Section 5.1.

We define an encoding $f : \{a,b\}^n \to \mathbb{C}^Q$ of n-bit strings into unit superpositions over the basis states of the QFA M by letting $|f(x)\rangle$ be the state of the automaton M after the input string $x \in \{a,b\}^n$ has been read. We assert that f is a serial encoding.

To show that f is indeed such an encoding, we exhibit a suitable measurement for the ith bit of the input for every $i \in [1..n]$. Let, for $y \in \{a,b\}^{n-i}$, $V_i(y) = U_\$ U_y^{-1}$, where U_y stands for the identity operator if y is the empty word, and for $U_{y_{n-i}} U_{y_{n-i-1}} \cdots U_{y_1}$ otherwise. The ith measurement then consists of first applying the unitary transformation $V_i(x_{i+1} \cdots x_n)$ to $|f(x)\rangle$, and then measuring the resulting superposition with respect to $E_{\rm acc} \oplus E_{\rm rej} \oplus E_{\rm non}$. (Note that the measurement for the ith bit assumes the knowledge of all the successive bits $x_{i+1}, \ldots, x_n$ of the input.) Since for words with length at most n, containment in L_n is decided by the last letter, and because such words are accepted or rejected by the n-restricted QFA M with probability at least p *only after the entire input has been read*, the probability of observing $E_{\rm acc}$ if $x_i = a$, or $E_{\rm rej}$ if $x_i = b$, is at least p. Thus, f defines a serial encoding, as claimed. ■

Theorem 4.1 now immediately implies that $\lceil \log |Q| \rceil = \Omega(n/\log n)$ and thus $|Q| = 2^{\Omega(n/\log n)}$, where Q is as in the claim above.

5.3 Extension to general QFAs

It only remains to show that the lower bound on the size of restricted QFAs obtained above implies a lower bound on the size of general QFAs accepting L_n. We do this by showing that we can convert *any* 1-way QFA to an r-restricted 1-way QFA which is only $O(r)$ times as large as the original QFA. It follows that the $2^{\Omega(n/\log n)}$ lower bound on number of states of n-restricted 1-way QFAs recognizing L_n continues to hold for general 1-way QFAs for L_n, exactly as stated in Theorem 1.3.

The idea behind the construction of a restricted QFA, given a general QFA, is to carry the halting parts of the superposition of the original automaton as "distinguished" non-halting parts of the state of the new automaton till at least r more symbols of the input have been read since the halting part was generated or until the right end marker is encountered, and then mapping them to accepting or rejecting subspaces appropriately.

Lemma 5.1 *Let M be a 1-way QFA with S states recognizing a language L with probability p. Then there is an r-restricted 1-way QFA M' with $O(rS)$ states that recognizes L with probability p.*

Proof: Let M be a 1-way QFA with Q as the set of basis states, $Q_{\rm acc}$ as the set of accepting states, $Q_{\rm rej}$ as the set of rejecting states, and q_0 as the starting state. Let M' be the automaton with basis state set

$$Q \;\cup\; (Q_{\rm acc} \times \{0, 1, \ldots, r+1\} \times \{\text{acc}, \text{non}\}) \;\cup$$
$$(Q_{\rm rej} \times \{0, 1, \ldots, r+1\} \times \{\text{rej}, \text{non}\}).$$

Let $Q_{\rm acc} \cup (Q_{\rm acc} \times \{0, 1, \ldots, r+1\} \times \{\text{acc}\})$ be its set of accepting states, let $Q_{\rm rej} \cup (Q_{\rm rej} \times \{0, 1, \ldots, r+1\} \times \{\text{rej}\})$ be the set of rejecting states, and let q_0 be the starting state. If, for a state $q \in Q$, there is a transition

$$|q\rangle \;\mapsto\; \sum_{q'} \alpha_{q'} \, |q'\rangle$$

in M on symbol σ, then in M', we have the following transitions. On the '\$' symbol, we have the same transition, and on $\sigma \neq \$$, we have

$$|q\rangle \;\mapsto\; \sum_{q' \notin Q_{\rm acc} \cup Q_{\rm rej}} \alpha_{q'} \, |q'\rangle + \sum_{q' \in Q_{\rm acc} \cup Q_{\rm rej}} \alpha_{q'} \, |q', 0, \text{non}\rangle .$$

The transitions from the states not originally in M are given by the following rules. On the '\$' symbol,

$$|q, i, \text{non}\rangle \;\mapsto\; \begin{cases} |q, i, \text{acc}\rangle & \text{if } q \in Q_{\rm acc} \text{ and } i \leq r \\ |q, i, \text{rej}\rangle & \text{if } q \in Q_{\rm rej} \text{ and } i \leq r \end{cases}$$

and on a symbol $\sigma \in \{a, b\}$,

$$|q, i, \text{non}\rangle \mapsto \begin{cases} |q, i+1, \text{non}\rangle & \text{if } i < r \\ |q, i+1, \text{acc}\rangle & \text{if } q \in Q_{\text{acc}} \text{ and } i = r \\ |q, i+1, \text{rej}\rangle & \text{if } q \in Q_{\text{rej}} \text{ and } i = r \end{cases}$$

The rest of the transitions may be defined arbitrarily, subject to the condition of unitarity.

It is not difficult to verify that M' is an r-restricted 1-way QFA (of size $O(rS)$) accepting the same language as M, and with the same probability. ■

5.4 Some remarks

We observe that the size $O(n)$ versus size $\Omega(2^n)$ separation between DFAs and 1-way QFAs is the worst possible if we restrict ourselves to languages that can be accepted by 1-way QFAs with probability of correctness that is high enough (at least 7/9). Such languages include all *finite* regular languages, since these can be accepted by 1-way RFAs. This follows from the result of Ambainis and Freivalds [2] that any language accepted by a QFA with high enough probability can be accepted by a 1-way RFA which is at most exponentially bigger than the minimal DFA accepting the language. However, it is not clear that this is also the largest separation in the case of languages that are accepted by 1-way QFAs with smaller probability of correctness.

Another open problem involves the blow up in size while simulating a 1-way probabilistic finite automata (PFA) by a 1-way QFA. The only known way for doing this is by simulating the PFA by a 1-way DFA and then simulating the DFA by a QFA. Both simulating a PFA by a DFA [1, 8, 12] and simulating a DFA by a QFA (this paper) can involve exponential or nearly exponential increase in size. This means that the straightforward simulation of a probabilistic automaton by a QFA (described above) could result in a doubly-exponential increase in the size. However, we do not know of any examples where both transforming a PFA into a DFA and transforming a DFA into a QFA cause big increases of size. Better simulations of probabilistic automata by QFAs may well be possible.

In general, it is not known how to simulate a probabilistic coin-flip by a purely quantum-mechanical algorithm if space is limited. For example, the only known simulation of $S(n)$-space probabilistic Turing machines by $S(n)$-space quantum Turing machines can create quantum Turing machines running in expected time of $2^{2^{S(n)}}$ [14]. Finding better simulations or proving that they do not exist is another interesting direction to explore.

Acknowledgements

We would like to thank Ike Chuang for showing us the 3-into-1 quantum encoding. We also would like to thank Dorit Aharonov, Ike Chuang, Michael Nielsen, Steven Rudich and Avi Wigderson for many interesting discussions.

References

[1] A. Ambainis. The complexity of probabilistic versus deterministic finite automata. *Proceedings of the International Symposium on Algorithms and Computation (ISAAC'96), Lecture Notes in Computer Science* **1178**, 1996, pp. 233–239.

[2] A. Ambainis and R. Freivalds. 1-way quantum finite automata: strengths, weaknesses and generalizations. *Proceedings of the 39th IEEE Conference on Foundations of Computer Science,* 1998, pp. 332–341.

[3] C. Bennett, E. Bernstein, G. Brassard and U. Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing* **26(5)**, 1997, pp. 1510–1523.

[4] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM Journal on Computing* **26(5)**, 1997, pp. 1411–1473.

[5] I.L. Chuang. Personal communication, 1997.

[6] G.D. Cohen. A nonconstructive upper bound on covering radius. *IEEE Transactions on Information Theory* **IT-29(3)**, 1983, pp. 352–353.

[7] T.M. Cover and J.A. Thomas. *Elements of information theory.* Wiley, New York, 1991.

[8] R. Freivalds. On the growth of the number of states in result of determinization of probabilistic finite automata. *Automatic Control and Computer Sciences* **13(3)**, 1982, pp. 39–42.

[9] A.S. Kholevo. Some estimates of the information transmitted by quantum communication channels. *Problems of Information Transmission* **9**, 1973.

[10] A. Kondacs and J. Watrous. On the power of quantum finite state automata. *Proceedings of the 38th IEEE Conference on Foundations of Computer Science,* 1997, pp. 66–75.

[11] C. Moore and J. Crutchfield. Quantum automata and quantum grammars. Santa-Fe Institute Working Paper 97-07-062, 1997. Also available at `http://xxx.lanl.gov/archive/quant-ph/9707031`.

[12] M.O. Rabin. Probabilistic automata. *Information and Control* **6**, 1963, pp. 230–245.

[13] U. Vazirani. On the power of quantum computation. *Philosophical Transactions of the Royal Society of London, Series A* **356**, 1998, pp. 1759–1768.

[14] J. Watrous. Relationships between quantum and classical space-bounded complexity classes. *Proceedings of the 13th IEEE Conference on Computational Complexity,* 1998, pp. 210–227.