The class of languages recognizable by 1-way quantum finite automata is not closed under union ⋆

Māris Valdats

Institute of Mathematics and Computer Science, University of Latvia, Raiņa bulv. 29, Rīga, Latvia⋆⋆
sd70066@lanet.lv

Abstract. In this paper we develop little further the theory of quantum finite automata (QFA). There are already few properties of QFA known, that deterministic and probabilistic finite automata do not have e.g. they cannot recognize all regular languages. In this paper we show, that class of languages recognizable by QFA is not closed under union, even not under any Boolean operation, where both arguments are significant.

1 Introduction

In recent years quantum computing is developing very quickly. Almost all classical computational models already have their quantum analogues. Quantum finite automata is probably the simplest of them and this paper is about them. Here we will not repeat basic facts, but as an introduction to quantum finite automata (QFA) would recommend you these papers: [CM 97,AF 98]. There are a lot of explanations and even examples. Here we will recall only the definition and main results so far.

1.1 Definition

Definition 1.1. *A QFA is a tuple* $M = (Q; \Sigma; V; q_0; Q_{acc}; Q_{rej})$ *where* Q *is a finite set of states,* Σ *is an input alphabet,* V *is a transition function,* $q_0 \in Q$ *is a starting state, and* $Q_{acc} \subseteq Q$ *and* $Q_{rej} \subseteq Q$ *are sets of accepting and rejecting states* $(Q_{acc} \cap Q_{rej} = \emptyset)$*. The states in* Q_{acc} *and* Q_{rej}*, are called* halting states *and the states in* $Q_{non} = Q - (Q_{acc} \cup Q_{rej})$ *are called* non halting states. κ *and* \$ *are symbols that do not belong to* Σ*. We use* κ *and* \$ *as the left and the right endmarker, respectively. The* working alphabet *of* M *is* $\Gamma = \Sigma \cup \{\kappa; \$\}$.

The transition function V *is a mapping from* $\Gamma \times l_2(Q)$ *to* $l_2(Q)$ *such that, for every* $a \in \Gamma$*, the function* $V_a : l_2(Q) \to l_2(Q)$ *defined by* $V_a(x) = V(a, x)$ *is a unitary transformation.*

⋆ This paper also available in Latvian - contact author

⋆⋆ Research supported by Grant No.96.0282 from the Latvian Council of Science

The computation of a QFA starts in the superposition $|q_0\rangle$. Then transformations corresponding to the left endmarker κ, the letters of the input word x and the right endmarker \$ are applied. The transformation corresponding to $a \in \Gamma$ consists of two steps.

1. First, V_a is applied. The new superposition ψ' is $V_a(\psi)$ where ψ is the superposition before this step.

2. Then, ψ' is observed with respect to $E_{acc}, E_{rej}, E_{non}$ where $E_{acc} = span\{|q\rangle : q \in Q_{acc}\}$, $E_{rej} = span\{|q\rangle : q \in Q_{rej}\}$, $E_{non} = span\{|q\rangle : q \in Q_{non}\}$. It means, that if the system's state before measurement was

$$\psi' = \sum_{q_i \in Q_{acc}} \alpha_i |q_i\rangle + \sum_{q_j \in Q_{rej}} \beta_j |q_j\rangle + \sum_{q_k \in Q_{non}} \gamma_k |q_k\rangle$$

then measurement accepts ψ' with probability $\Sigma \alpha_i^2$, rejects with probability $\Sigma \beta_j^2$ and continues process with probability $\Sigma \gamma_k^2$ with system having state $\psi = \Sigma \gamma_k |q_k\rangle$.

We regard these two transformations as reading a letter a. We use V'_a to denote the transformation consisting of V_a followed by projection to E_{non}. This is the transformation mapping ψ to the non-halting part of $V_a(\psi)$. We use V'_w to denote the product of transformations $V'_w = V'_{a_n} V'_{a_{n-1}} \ldots V'_{a_2} V'_{a_1}$, where a_i is the i-th letter of the word w. Also we use ψ_y to denote the non-halting part of QFA's state after reading the left endmarker κ and the word $y \in \Sigma^*$. From the notation follows, that $\psi_w = V'_{\kappa w}(|q_0\rangle)$.

We will say, that automaton recognizes language L with probability p ($p > \frac{1}{2}$) if automaton accepts any word $x \in L$ with probability $\geq p$ and rejects any word $x \notin L$ with probability $\geq p$.

1.2 Main results so far

It has been shown [KW 97], that class of languages, recognizable by QFA is a proper subset of regular languages. Also it has been shown (Theorems **??** and **??** taken from [ABFK 99]) , that classes of languages recognizable by QFA with different probabilities differs.

Theorem 1.1. *Let's denote hierarchy of languages* $L_n = a_1^* a_2^* a_3^* a_4^* ... a_n^*$. *Then language* L_n *can be recognized with probability greater than* $\frac{1}{2} + \frac{1}{4n}$ *but not with greater than* $\frac{1}{2} + \frac{3}{\sqrt{n-1}}$.

Theorem 1.2. *Let* L *be a language and* M *be its minimal automaton. Assume that there is a word* x *such that* M *contains states* q_1, q_2 *satisfying:*

1. $q_1 \neq q_2$,
2. *If* M *starts in the state* q_1 *and reads* x, *it passes to* q_2,
3. *If* M *starts in the state* q_2 *and reads* x, *it passes to* q_2, *and*
4. q_2 *is neither "all-accepting" state, nor "all-rejecting" state.*

   Then L *cannot be recognized by a 1-way quantum finite automaton with probability* $7/9 + \varepsilon$ *for any fixed* $\varepsilon > 0$.

   If we add one more condition

5. *There is a word y such that if M starts in q_2 and reads y, it passes to q_1, then L cannot be recognized by any 1-way quantum finite automaton.*

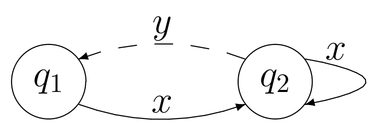

Fig. 1. Conditions of theorem 1.2, condition 5 - with dotted line

Theorem 1.1 is proved in [ABFK 99], theorem 1.2 is proved in [AF 98]

All recently known regular languages that are not recognizable by QFA have these properties 1-5. The first thing we will do in next chapter, is construct a language, that is not recognizable by a QFA, and has not the property 5.

There are also a lot of results [AF 98,K 98] about number of states needed for a QFA to recognize different languages. It can be exponentially less than even for probabilistic automata but for reversible automata (a special type of quantum automata) it can be also exponentially more than for deterministic automata.

It is yet unknown, what is the class of languages, recognizable by QFA.

2 Main results

Let's define a language $L_1 = a^*bb^*a(b^*ab^*a)b^* + a^*$. Its minimal automaton G_1 is

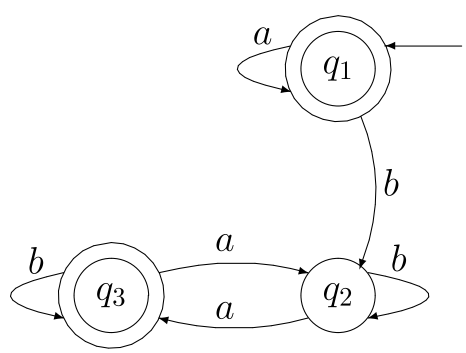

Fig. 2. Automaton G_1

States q_1 and q_2 of automaton G_1 and the word b fulfills conditions 1-4 of theorem 1.2 but condition 5 is not fulfilled.

Theorem 2.1. *Language L_1 is not recognizable by a QFA.*

Proof. As it is long and technical, it is presented in appendix.

Now let's consider 2 other languages L_2 and L_3. For variety they will be recognizable by QFA. So they are:

$L_2 = (aa)^* bb^* a(b^* ab^* a)b^* + (aa)^*$

$L_3 = aL_2 = a(aa)^* bb^* a(b^* ab^* a)b^* + a(aa)^*$

More easy is to look at their minimal automatons G_2 and G_3 (Fig.3 and Fig.4) They differ only with a starting state. That is the only thing, where their quantum analogs K_2 and K_3 are going to differ, too.

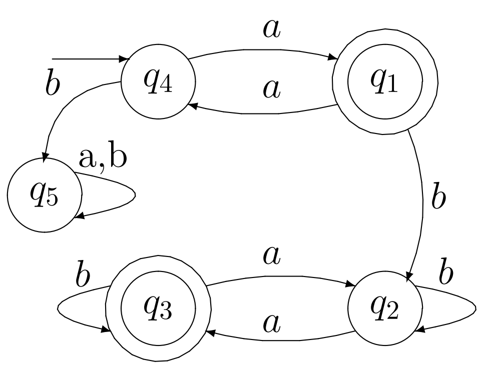

Fig. 3. Automaton G_2

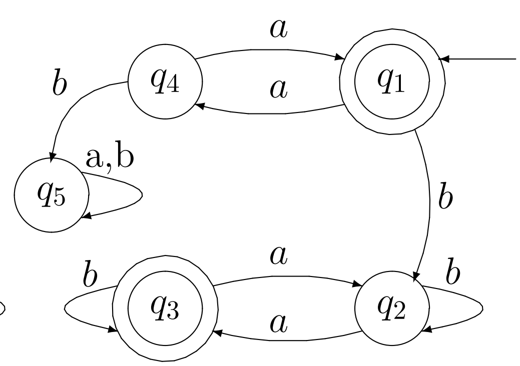

Fig. 4. Automaton G_3

So, the automaton K_2 will consist of 8 states: q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, where $Q_{non} = \{q_1, q_2, q_3, q_4\}$, $Q_{acc} = \{q_5, q_8\}$, $Q_{rej} = \{q_6, q_7\}$.

The unitary transform matrixes V_κ, V_a, V_b and $V_\$$ are:

$$V_\kappa = \begin{pmatrix} \sqrt{\frac{2}{3}} & \sqrt{\frac{1}{3}} & 0 & 0 & 0 & 0 & 0 & 0 \\ \sqrt{\frac{1}{3}} & -\sqrt{\frac{2}{3}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\sqrt{\frac{2}{3}} & \sqrt{\frac{1}{3}} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{\frac{2}{3}} & \sqrt{\frac{1}{3}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, V_a = \begin{pmatrix} 0&0&0&1&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \end{pmatrix}$$

$$V_b = \begin{pmatrix} 0 & 0 & 0 & 0 & \sqrt{\frac{1}{2}} & \sqrt{\frac{1}{2}} & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \sqrt{\frac{1}{2}} & 0 & 0 & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & 0 \\ \sqrt{\frac{1}{2}} & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, V_\$ = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The starting state for K_2 is q_1, for K_3 it is q_4. Now we will look only at K_2. For K_3 it is similar.

State q_1 in G_2 corresponds to $\psi_1 = \sqrt{\frac{2}{3}}\,|q_1\rangle + \sqrt{\frac{1}{3}}\,|q_2\rangle$ in K_2

State q_2 in G_2 corresponds to $\psi_2 = \sqrt{\frac{1}{3}}\,|q_2\rangle$ in K_2

State q_3 in G_2 corresponds to $\psi_3 = \sqrt{\frac{1}{3}}\,|q_3\rangle$ in K_2

State q_4 in G_2 corresponds to $\psi_4 = \sqrt{\frac{2}{3}}\,|q_4\rangle + \sqrt{\frac{1}{3}}\,|q_3\rangle$ in K_2

1. After reading the left endmarker κ automaton is in state ψ_1 or $V'_\kappa(|q_1\rangle) = \psi_1$, also starting state of G_2 is q_1.
2. If by reading letter a automaton G_2 passes from q_1 to q_4 or back, then automaton K_2 state changes from ψ_1 to ψ_4 or back.
3. If automaton K_3 is in state q_4 and receives letter b then it rejects input with probability $\frac{2}{3}$ so we have no special interest what happens further (and it is correct, because G_2 is now in "all rejecting" state q_5).
4. If automaton G_2 is in state q_1 and receives letter b it passes to q_3, if automaton K_2 is in state ψ_1 and receives letter b it passes to state $\frac{1}{\sqrt{3}}\,|q_2\rangle + \frac{1}{\sqrt{3}}\,|q_5\rangle + \frac{1}{\sqrt{3}}\,|q_6\rangle$ and after measurement accepts input with probability $\frac{1}{3}$, rejects input with the same probability $\frac{1}{3}$, or continues in state ψ_2.
5. If by reading letter a automaton G_2 passes from q_2 to q_3 or back, then automaton's K_2 state changes from ψ_2 to ψ_3 or back. By reading letter b G_2 passes from q_2 to q_2 and from q_3 to q_3. Also K_2 - if it is in ψ_2 or ψ_3 and receives b it does not change its state.
6. If automaton receives the right endmarker in state ψ_1 then input is accepted with probability $\frac{2}{3}$.
7. If automaton receives the right endmarker in state ψ_2 then input is rejected with probability $\frac{1}{3}$ and as it was rejected with same probability so far, the total probability to reject input is $\frac{2}{3}$.
8. If automaton receives the right endmarker in state ψ_3 then input is accepted with probability $\frac{1}{3}$ and as it was accepted with same probability so far, the total probability to reject input is $\frac{2}{3}$.
9. If automaton receives the right endmarker in state ψ_4 then input is rejected with probability $\frac{2}{3}$.

In these 9 points we wanted to show, that automaton K_2 performs computation the same way as G_1. While automaton G_2 is in one of its states $q_1, \ldots, q_4$, K is in a corresponding state $\psi_1, \ldots, \psi_4$. AutomatonK_2 accepts input with probability $\frac{2}{3}$ iff it receives right endmarker \$ in one of states ψ_1 or ψ_3, corresponding whom q_1 and q_3 are the only accepting states in G_1. So we can conclude, that K_2 accepts language L_2 with probability $\frac{2}{3}$.

What are languages L_1, L_2 and L_3 informally?

L_3 consists of all words which start with <u>odd</u> number of letters a and after first letter b (if there is such) there is odd number of letters a.

L_2 consists of all words which start with <u>even</u> number of letters a and after first letter b (if there is such) there is odd number of letters a.

L_1 consists of all words which start with <u>any</u> number of letters a and after first letter b (if there is such) there is odd number of letters a.

So, it is almost evident, that $L_1 = L_2 \bigcup L_3$.

Corollary 2.1. *There are two languages L_2 and L_3 which are recognizable by QFA (with probability $\frac{2}{3}$), union of whom $L_1 = L_2 \bigcup L_3$ is not recognizable by QFA.*

Corollary 2.2. *The class of languages recognizable by QFA is not closed under union.*

As $L_2 \bigcap L_3 = \emptyset$ then also $L_1 = L_2 \Delta L_3$. So the class of languages recognizable by QFA is not closed under symmetric difference. From this and from the fact, that this class is closed under complement easy follows:

Corollary 2.3. *The class of languages recognizable by QFA is not closed under any binary Boolean operation, where both arguments are significant.*

3 Some more details

In previous section we found two languages L_2 and L_3 recognizable by QFA with probability $\frac{2}{3}$, union of whom is not recognizable by any QFA. What if we increase the probability?

Theorem 3.1. *If 2 languages L_1 and L_2 are recognizable by QFA with probabilities p_1 and p_2 and $\frac{1}{p_1} + \frac{1}{p_2} < 3$, then $L = L_1 \bigcup L_2$ is also recognizable by QFA with probability $\frac{2p_1p_2}{p_1+p_2+p_1p_2}$.*

In case if $p_1, p_2 > \frac{2}{3}$ the condition holds.

Proof. We have automaton K_1, which accepts L_1 with probability p_1 and automaton K_2, which accepts L_2 with probability p_2. We will make automaton K which will work like this:

1. Runs K_1 with probability $\frac{p_2}{p_1+p_2+p_1p_2}$,
2. Runs K_2 with probability $\frac{p_1}{p_1+p_2+p_1p_2}$,

3. Accepts input with probability $\frac{p_1p_2}{p_1+p_2+p_1p_2}$.

To make such an automaton we just have to make tensor product $K_1 \otimes K_2 \otimes K_3$ where K_3 consists of only one "all accepting" state, and modify a little its V_κ matrix. When we have done it, we have:

1. $w \in L_1$ and $w \in L_2 \longrightarrow$ input is accepted with probability
$$\frac{p_2}{p_1+p_2+p_1p_2} * p_1 + \frac{p_1}{p_1+p_2+p_1p_2} * p_2 + \frac{p_1p_2}{p_1+p_2+p_1p_2} * 1 = 1$$
2. $w \in L_1$ and $w \notin L_2 \longrightarrow$ input is accepted with probability at least
$$\frac{p_2}{p_1+p_2+p_1p_2} * p_1 + \frac{p_1p_2}{p_1+p_2+p_1p_2} * 1 = \frac{2p_1p_2}{p_1+p_2+p_1p_2}$$
3. $w \notin L_1$ and $w \in L_2 \longrightarrow$ input is accepted with probability at least
$$\frac{p_1}{p_1+p_2+p_1p_2} * p_2 + \frac{p_1p_2}{p_1+p_2+p_1p_2} * 1 = \frac{2p_1p_2}{p_1+p_2+p_1p_2}$$
4. $w \notin L_1$ and $w \notin L_2 \longrightarrow$ input is rejected with probability at least
$$\frac{p_2}{p_1+p_2+p_1p_2} * p_1 + \frac{p_1}{p_1+p_2+p_1p_2} * p_2 = \frac{2p_1p_2}{p_1+p_2+p_1p_2}$$

So automaton K recognizes L with probability at least
$$\frac{2p_1p_2}{p_1+p_2+p_1p_2} = \frac{1}{2} + \frac{3-(\frac{1}{p_1}+\frac{1}{p_2})}{4(1+\frac{1}{p_1}+\frac{1}{p_2})} > \frac{1}{2}$$

All this has also a nice geometric interpretation. We are going to build a linear function f from probabilities x_1, x_2 to probability x such, that $f(p_1,p_2) \geq \frac{1}{2}+\varepsilon$, $f(p_1,0) \geq \frac{1}{2}+\varepsilon$, $f(0,p_2) \geq \frac{1}{2}+\varepsilon$, $f(1-p_1,1-p_2) \leq \frac{1}{2}-\varepsilon$. Geometrically we consider a plane x,y where each word w is located in a point (x,y), where x is probability that K_1 accepts w and y is the probability, that K_2 accepts w.

S_1 is the place, where lies all words, that do not belong to L.

S_2 is the place, where lies all words, that belong to L.

If we can (Fig.5) separate these two parts with a line $ax+by=c$ then we can construct automaton "$K = aK_1 + bK_2$" with c as isolated cut point. If we can not (Fig.6), then this method doesn't help. And as it was shown higher, sometimes none of other methods can help, too.

Case when $p_1 = p_2 = \frac{2}{3}$ (Fig.7) is the limit case. If any of the probabilities were a little bit greater then this method would help.

Sometimes it may be, that there are no words w such, that K_1 or K_2 would reject with probability $1-t$ or greater. Then (Fig.8) you can see, that now it is easier, to make such a line, so condition $\frac{1}{p_1}+\frac{1}{p_2} < 3$ can be weakened (the probabilities in Fig.8 are the same as in Fig.6). In the limit case, when rejecting probabilities are only p_1 and p_2, S_1 is the point $(1-p_1, 1-p_2)$ (Fig.9). So with

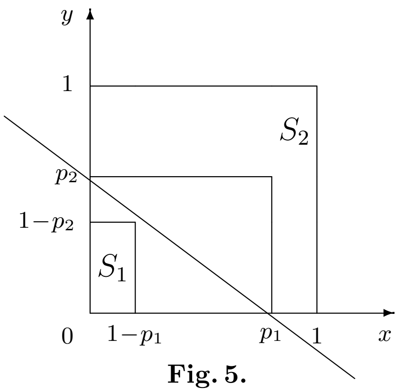

Fig. 5.

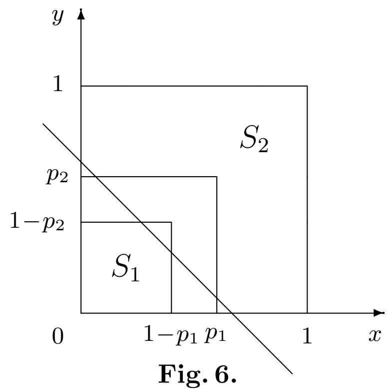

Fig. 6.

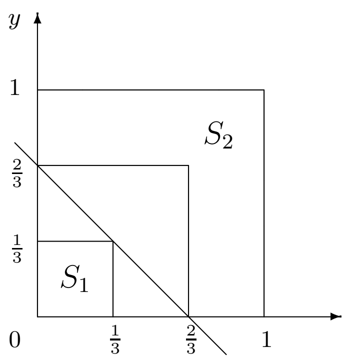

Fig. 7.

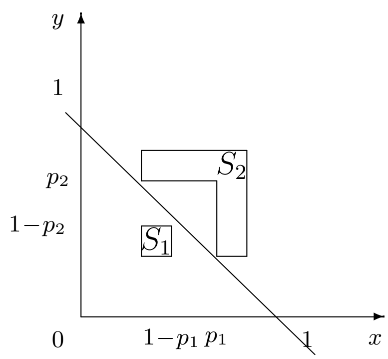

Fig. 8.

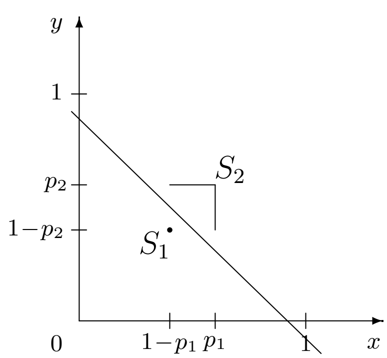

Fig. 9.

any p_1 and p_2 you can separate S_1 from S_2 with a line, from what follows you can always construct $K = K_1 \bigcup K_2$.

Now it is clear, that languages L_2 and L_3 defined in chapter 2, cannot be recognized with probability greater than $\frac{2}{3}$ so the construction presented there is best possible.

4 Appendix - proof of theorem 1.2

In this proof we are going to use one classical result from [BV 97], so as it has very little connection with all other proof, we are going to present it here, in the beginning.

Lemma 4.1. *If ψ and ϕ are two states of quantum system and $\|\psi - \phi\| < \varepsilon$ then total variation distance between probability distributions generated by measurements on ψ and ϕ are less than 2ε.*

Proof. Let's denote

$$\varphi = \frac{1}{2}(\psi + \phi) = \sum_i \alpha_i \left| q_i \right\rangle$$

and

$$\pi = \frac{1}{2}(\psi - \phi) = \sum_i \gamma_i \left| q_i \right\rangle \ , \ \|\pi\| < \frac{\varepsilon}{2}$$

The total variation distance between two probability distributions $P = \sum p_i \left| q_i \right\rangle$ and $R = \sum r_i \left| q_i \right\rangle$ is defined as

$$\Delta = \sum_i |p_i - r_i|$$

As $\psi = \varphi + \pi$ and $\phi = \varphi - \pi$ then total variation distance is

$$\Delta = \sum_i |\|\alpha_i + \gamma_i\|^2 - \|\alpha_i - \gamma_i\|^2| =$$

$$= \sum_i |2\alpha_i \gamma_i^* + 2\alpha_i^* \gamma_i| \leq 4 \sum_i |\alpha_i||\gamma_i|$$

Now using Cauchy inequality we get

$$\Delta \leq 4\sqrt{\sum_i |\alpha_i|^2 * \sum_i |\gamma_i|^2} = 4\|\varphi\|\|\pi\| < 2\|\varphi\|\varepsilon$$

and as $\|\varphi\| \leq 1$ then $\Delta < 2\varepsilon$.

This lemma shows the intuitively clear fact, that close states are accepted with close probabilities. In our proof we are going to use it in such a form, that difference between acceptance probabilities (and also rejection probabilities) of states ψ and ϕ where $\|\psi - \phi\| < \varepsilon$ is less than 2ε.

Let's say, that there is such QFA K, which recognizes the same language as G with a fixed probability $\frac{1}{2}+\varepsilon$. First thing we will do is decompose its state space E_{non} into 2 components $E_{non} = E_1 \oplus E_2$. In E_1 we will put all vectors ψ with such a property: if automaton K starts in ψ then the probability, that input is accepted or rejected while reading any word $w \in \Sigma^*$ is 0. Or $\forall w \in \Sigma^*\ \|\psi\| = \|V'_w(\psi)\|$. E_2 will contain all vectors orthogonal to E_1.

More formally we will do it this way:

$$
\begin{aligned}
&E^0 = E_{non}\\
&E^1 = \{\psi \mid \psi \in E^0\ \&\ V_a(\psi) \in E^0\ \&\ V_b(\psi) \in E^0\}\\
&E^2 = \{\psi \mid \psi \in E^1\ \&\ V_a(\psi) \in E^1\ \&\ V_b(\psi) \in E^1\}\\
&E^3 = \{\psi \mid \psi \in E^2\ \&\ V_a(\psi) \in E^2\ \&\ V_b(\psi) \in E^2\}\\
&\ldots\\
&E^{j+1} = \{\psi \mid \psi \in E^j\ \&\ V_a(\psi) \in E^j\ \&\ V_b(\psi) \in E^j\}\\
&E_1 = \bigcap_{j=0}^{+\infty} E^j \qquad E_2 = E \ominus E
\end{aligned}
$$

At first we will notice, that $E^{j+1} \subseteq E^j$, so $\dim E^{j+1} \leq \dim E^j$. If $\dim E^j = \dim E^{j+1}$ then $E^j = E^{j+1} = E^{j+2} = ...$, hence $\forall j \geq n\ E^j = E^n$, where $n = \dim E_{non}$, or n is just the number of states in Q_{non}. So as well we can define $E_1 = \bigcap_{j=0}^{n} E^j$. This means, that for each state ψ not in E_1 there is a word of length n, which projects part of ψ to Q_{acc} or Q_{rej}. As in E_1 there are no projections, then $V'_a(\psi) = V_a(\psi)$ and $V'_b(\psi) = V_b(\psi)$ if $\psi \in E_1$, so V'_a and V'_b are unitary in E_1. And as product of 2 unitary matrixes is also unitary, so V'_w is unitary in E_1 for all $w \in \Sigma^*$. From the definition of E_1 follows, that $\forall \psi \in E_1 \Rightarrow V'_a(\psi) \in E_1$ and $V'_b(\psi) \in E_1$. As unitary transformations transforms orthogonal vectors to orthogonal, we can conclude, that

$$\forall \psi \in E_2 \Rightarrow V_a(\psi) \in E_2 \oplus E_{acc} \oplus E_{rej}, V_b(\psi) \in E_2 \oplus E_{acc} \oplus E_{rej}$$

therefore

$$\forall \psi \in E_2 \Rightarrow V'_a(\psi) \in E_2, V'_b(\psi) \in E_2$$

So we can say, that computation is performed in E_1 and E_2 independently.

Lemma 4.2. *For every $\psi \in E_2$ and every δ there is such a word $w \in \Sigma^*$, that $\|V'_w(\psi)\| < \delta$ or in other words* $\inf\{\|V'_w(\psi)\| \mid \psi \in E_2, w \in \Sigma^*\} = 0$.

Proof. For each vector $\psi \in E_2$ let's denote $M_\psi = min\{\|V'_w(\psi)\| \mid w \in \Sigma^n\}$ and $M = \{M_\psi \mid \psi \in E_2, \|\psi\| \leq 1\}$ where n is still the number of states in Q_{non}. It means, that for each ψ we find a word w with length n reading which automaton would make maximum projections. It is clear, that $M_\psi < 1$, otherwise ψ would be in E_1. We denote $S = \sup(M)$. As set $\{\psi \mid \psi \in E_2\ \|\psi\| \leq 1\}$ is closed, so is M. Hence $S \in M$ and so $S < 1$. Now the proof is easy. For each $\psi \in E_2$ we can construct word $w \in \Sigma^{kn}$ such, that $\|V'_w(\psi)\| \leq S^k \|\psi\| \to 0$ when $k \to \infty$.

We'll say, that state ψ_1 is reachable from state ψ_2, if there is a sequence of words $\{w_i\}$ such, that

$$\lim_{i\to\infty} \|V'_{w_i}(\psi_2) - \psi_1\| = 0$$

Let's put $\delta_i = \|V'_{w_i}(\psi_2) - \psi_1\|$, now $\delta_i \to 0$, when $i \to \infty$. Let's look at sequence of vectors

$$\psi_1, U(\psi_1), U^2(\psi_1), U^3(\psi_1), \ldots$$

where $U = V'_{w_i}$. As all they are inside finite space???, and they are infinitely many, then I can find a pair of them as close to one another as I wish, say

$$\|U^k(\psi_1) - U^m(\psi_1)\| < \delta_i,\ k < m$$

Then

$$\|U^k(\psi_1 - U^{m-k}(\psi_1))\| < \delta_i$$

and also

$$\|\psi_1 - U^{m-k}(\psi_1)\| < \delta_i$$

because unitary transformation doesn't change the length of vector. So now we have $\|U(\psi_2) - \psi_1\| = \delta_i$ and $\|\psi_1 - U^{m-k}(\psi_1)\| < \delta_i$. By triangle inequality we can conclude, that

$$\|U(\psi_2) - U^{m-k}(\psi_1)\| < 2\delta_i$$

or $\|\psi_2 - U^{m-k-1}(\psi_1)\| < 2\delta_i$. What does it mean? If we denote $u_i = w_i^{m-k-1}$ (m and k may be different for each w_i) then

$$\lim_{i\to\infty} \|V'_{u_i}(\psi_1) - \psi_2\| \le \lim_{i\to\infty} 2\delta_i = 0$$

or reachability is symmetric.

It is also very easy to prove, that reachability is transitive. It follows directly from the fact, that transformations are continuous.

To prove the transitivity of reachability we even did not need the unitarity of transformations, we used only their continuity, so reachability is transitive in E_{non}, and symmetric in E_1, where the transformations are unitary. So it is equivalence in E_1.

Let's denote the state after reading left endmarker $\psi_0 = \psi_I + \psi_{II}$, where $\psi_I \in E_1$ and $\psi_{II} \in E_2$. Also after reading any word $w \in \Sigma^*$, the state is $V'_w(\psi_0) = V'_w(\psi_I) + V'_w(\psi_{II})$, where $V'_w(\psi_I) \in E_1$ and $V'_w(\psi_{II}) \in E_2$. Let's denote R the class of all reachable states from starting state ψ_I. Also let's denote $A(\psi)$ the probability to accept input, if automaton in state ψ receives right endmarker \$, and p_w the probability, that it has accepted input, while reading word κw. So the probability that automaton accepts word w is $p_w + A(\psi_w)$.

We begin with reading word w such, that $\|V'_w(\psi_{II})\| < k$, where k is very small. We can easily assume, that automaton G_1 after reading w is in state q_2, if it is not, then instead of w we can take wb or wa if it is in q_1 or q_3.

$$\psi_w = V'_w(\psi_0) = V'_w(\psi_I) + V'_w(\psi_{II}) = \psi_w^1 + \psi_w^2,\ \|\psi_w^2\| < k$$

In further calculation we can omit existence of ψ_w^2, and assume, that $\psi_w = \psi_w^1$, and $\forall u \in \Sigma^*\ p_w = p_{wu}$ because probability changes ψ_2 can make, are too small, when the difference between acceptance and rejection probabilities has to be at least 2ε.

Now we will divide R into 3 subsets.
$R_1 = \{\psi \mid \frac{1}{2} + \varepsilon \leq p_w + A(\psi) \leq 1\}$
$R_2 = \{\psi \mid \frac{1}{2} - \varepsilon < p_w + A(\psi) < \frac{1}{2} + \varepsilon\}$
$R_3 = \{\psi \mid 0 \leq p_w + A(\psi) \leq \frac{1}{2} - \varepsilon\}$

Lemma 4.3. *R_2 is empty.*

Proof. Let there be $\psi \in R_2$, we denote $max(\frac{1}{2} + \varepsilon - \|\psi\|, \|\psi\| - \frac{1}{2} + \varepsilon) = 2k$. As ψ is reachable from ψ_w, then there is word u, that $\|V'_u(\psi_w) - \psi\| < k$.

Then by lemma 4.1 $|A(\psi) - A(\psi_{wu})| < 2k$. So as $\frac{1}{2} - \varepsilon + 2k \leq p_w + A(\psi) \leq \frac{1}{2} + \varepsilon - 2k$ then $\frac{1}{2} - \varepsilon < p_{wu} + A(\psi_{wu}) < \frac{1}{2} + \varepsilon$, so the automaton accepts word wu with probability between $\frac{1}{2} - \varepsilon$ and $\frac{1}{2} + \varepsilon$ - contradiction.

If automaton is in state $\psi \in R_1$ and receives right endmarker \$, it accepts input. If automaton is in state $\psi \in R_3$ and receives right endmarker \$, it rejects input.

After reading letter a automaton must change its state from state, where it accepts input (R_1) to state, where it doesn't accept it (R_3), and vice versa, reading of letter b should not change anything. More formally

$$\forall \psi \in R_1 \Rightarrow V'_a(\psi) \in R_3, V'_b(\psi) \in R_1$$

$$\forall \psi \in R_3 \Rightarrow V'_a(\psi) \in R_1, V'_b(\psi) \in R_3$$

Now we have 2 choices:

1. $\psi_I \in R_1$. Let's look at states ψ_{bw} and ψ_{bwa}, where word w is chosen, to make $V'_{bw}(\psi_{II})$ negligible, and contains pair number of a-s (we can always find such). From this our choice follows, that $\psi_{bw} \in R_1$ and $\psi_{bwa} \in R_3$, so probability to accept word bw is greater than probability to accept bwa, at least for 2ε what is not correct, because bwa belongs to language but bw does not.
2. $\psi_I \in R_3$. The same problem. Let's look at states ψ_{abw} and ψ_{abwa}, where word w is chosen, to make $V'_{abw}(\psi_{II})$ negligible, and contains pair number of a-s (we can always find such). From this our choice follows, that $\psi_{abw} \in R_1$ and $\psi_{abwa} \in R_3$, so probability to accept word abw is greater than probability to accept $abwa$, at least for 2ε what is not correct, because $abwa$ belongs to language but abw does not.

So we have found, that automaton K does not recognize some words correctly, so it does not recognize language L_1. Now the proof is finished.

References

CM 97. C. Moore, J. Crutchfield. Quantum automata and quantum grammars. Santa-Fe Institute Working Paper 97-07-062, 1997. Also quant-ph/9707031.

AF 98. Andris Ambainis and Rūsiņš Freivalds. 1-way quantum finite automata: strengths, weaknesses and generalizations. *Proc. 39th FOCS,* 1998, p. 332–341. Also quant-ph/9802062.

BV 97. Ethan Bernstein, Umesh Vazirani, Quantum complexity theory. *SIAM Journal on Computing,* 26:1411-1473, 1997.

ABFK 99. Andris Ambainis, Richard Bonner, Rūsiņš Freivalds, Arnolds Ķikusts. Probabilities to accept languages by quantum finite automata. *COCOON'99* Also quant-ph/9904066.

K 98. Arnolds Ķikusts. A small 1-way quantum finite automaton. quant-ph/9810065.

KW 97. Attila Kondacs and John Watrous. On the power of quantum finite state automata. In *Proc. 38th FOCS*, 1997, p. 66–75.