量子有限オートマトンにおける決定不能問題

天野 正己 (Masami Amano)
京都大学工学部情報学科

岩間 一雄 (Kazuo Iwama)
京都大学大学院情報学研究科

Abstract

Our model in this paper is a 1.5-way quantum finite automaton which can move its head 0 or +1 position but not −1 position. It is shown that the most fundamental decision question, the emptiness problem, is not solvable for this model. Note that the emptiness problem is solvable for push-down automata and even for one-way nondeterministic stack automata.

1 Introduction

It has become an undoubted fact that the quantum mechanism gives us a certain kind of computational power which cannot be achieved by the conventional mechanism. However, although we have wonderful reports supporting this idea [Gro96, Sho94], there is still a lot of unclearness about what is real and concrete reason for such a miracle power and how hard it is to enjoy this power in designing algorithms. In their recent paper [KW97], Kondacs and Watrous gave a good hint against those questions: They introduced 2-way quantum finite automata, 2QFA's, and proved that a nonregular language, $\{a^n b^n \mid n \geq 1\}$, can be accepted by this model in linear time. (Although the same language can be recognized by 2-way probabilistic finite automata [Fre81] but it needs exponential expected time [DS89].) They also defined 1-way QFA's (1QFA's) which they prove can recognize only a proper subset of regular languages. Since finite automata are much simpler than the general computation model (i.e., Turing machines), it is obviously easier to understand concrete merits and demerits of the quantum mechanism.

More recently, Ambainis and Freivalds studied 1QFA's in more detail [AF98]. Their results are again interesting from the above viewpoint: (i) If the maximum error rate is bounded by a small value then 1QFA's cannot surpass the power of 1-way reversible finite automata and (ii) In some cases, we can design 1QFA's the number of whose states is exponentially less than conventional ones. More than that, they gave the following observation about the power of 2QFA's: The definition in [KW97] includes the head position into quantum state and hence the number of quantum states grows unlimitedly with the growing size of the input. Ambainis and Freivalds suggest that this could provide QFA's with unreasonably high power and at the same time could make the machine more complicated and more difficult to implement.

Their suggestion is probably true: In this paper it is shown that the real power of 2QFA's can actually be much higher than it seems. Our main result says that the emptiness problem, which asks whether a given machine accepts the empty set, is not solvable for 2QFA's. Moreover, it turns out that we do not need the 2-way head-move for this result but the 1.5-way head-move (the head can move 0 or +1 position to the right but not −1 position) is enough. Since all 1.5QFA's in this paper does not have a cycle of transitions without moving the head (called an ε-cycle), the theorem also holds for such 1.5QFA's which run obviously in linear time. The idea of this extension is to exploit the basic nature of quantum machines, i.e., the built-in parallelism of their computation.

The emptiness problem is probably the easiest one among popular decision questions such as the equivalence problem and the disjointness problem. The emptiness problem is solvable for push-down automata [HU79] and furthermore for 1-way nondeterministic stack automata (1SA's) [HU79]. 1SA's may not be linear time. It is known that all languages recognized by 1SA's are also recognized by deterministic linear-space TM's [HU68], but its proof is not that easy and the difference of power between these two models is far from trivial. Our unsolvability result means that 1.5QFA's can accept some languages that cannot be recognized by any 1SA's; it is quite reasonable to conclude that 1.5QFA's are very powerful in some cases.

It should be noted that this does not necessarily mean that 1.5QFA's are always powerful. We can probably prove that some regular languages cannot be recognized by 1.5QFA's by using the same technique as in [KW97]. However, it seems that such adversary languages have some common properties in their syntax, which can be avoided if we allow desirable encoding of input strings. In the proof of unsolvability, it is almost free to use such encoding techniques, which could make it easier to investigate the inherent power of machines without depending too much on the syntax of languages.

2 Definitions

2.1 2-way Quantum Finite Automata

Our definitions of QFA's are exactly the same as [KW97]. A 2-way QFA (2QFA) is given as $M = (Q, \Sigma, \delta, q_0, Q_{acc}, Q_{rej})$, where Q is a finite set of *states*, Σ is a finite set of *input symbols*, $q_0 \in Q$ is the *initial state*, and $Q_{acc} \subset Q$ and $Q_{rej} \subset Q$ are the sets of *accepting* and *rejecting states*, respectively. $Q_{non} = Q - (Q_{acc} \cup Q_{rej})$ is called the set of *non-halting states*. $\cent$ and $\$$ ($\notin \Sigma$) are the *left* and the *right end-markers*. $\Gamma = \Sigma \cup \{\cent, \$\}$ is called the set of *tape symbols*. $\delta : Q \times \Gamma \times Q \times \{-1, 0, 1\} \to C$ is called the *state transition function*.

A pair of a head position i and a state p, (i, p), is called a *configuration*. Therefore, a 2QFA M on a tape x of length n has $n|Q|$ different configurations, denoted by $C_n = Q \times Z_n$. A *superposition* of M is any norm 1 element of the Hilbert space $\mathcal{H}_n = l_2(C_n)$. For each $c \in C_n$, $|c\rangle$ denotes the unit vector with value 1 at c and 0 elsewhere. For a superposition $|\psi\rangle = \Sigma_{c \in C_n} \alpha_c |c\rangle$, α_c is the *amplitude* of the configuration c in $|\psi\rangle$. For each $q, q' \in Q$, $\delta \in \Gamma$ and $d \in \{-1, 0, 1\}$, $\delta(q, \sigma, q', d)$ shows the amplitude with which M in state q and reading σ will change to state q' and move its head d position right. Given a tape x, δ determines the *time-evolution operator* U_δ^x on $l_2(C_n)$ as follows

$$U_\delta^x |q, k\rangle = \sum_{q', d} \delta(q, x(k), q', d) |q', k + d(mod|x|)\rangle$$

where $x(k)$ is the input symbol at the kth cell. $U_\delta^x|q,k\rangle$ is naturally extended to $U_\delta^x|\psi\rangle$ by linearity. U_δ^x must be unitary and if so, $M$ is said to be *well-formed.*

The computation of $M$ on $x$ begins in superposition $|q_0,1\rangle$. Then U_δ^x is applied step by step. After each step, the current superposition is *observed.* Our *observable* is $Q_{acc} \oplus Q_{rej} \oplus Q_{non}$. If "accept" or "reject" is observed, the computation halts. It is said that *$x$ is accepted by $M$* iff "accept" is observed with probability greater than 1/2. $L(M)$ denotes the set of all the tapes accepted by $M$ and is called the *language accepted by $M$.*

To design well-formed 2QFA's, we can use the following approach: Suppose that we have designed a linear operator $V_\sigma : l_2(Q) \to l_2(Q)$ for each $\sigma \in \Gamma$ and a function $D : Q \to \{-1, 0, 1\}$. Let $\langle q'|V_\sigma|q\rangle$ denote the coefficient of $|q'\rangle$ in $V_\sigma|q\rangle$. Then define the transition function $\delta$ of $M$ as

$$\delta(q,\sigma,q',d) = \begin{cases} \langle q'|V_\sigma|q\rangle & \text{if } D(q') = d \\ 0 & \text{if } D(q') \neq d. \end{cases}$$

It is shown in [KW97] that $M$ is well-formed if and only if

$$\sum_{q'} \overline{\langle q'|V_\sigma|q_1\rangle}\langle q'|V_\sigma|q_2\rangle = \begin{cases} 1 & q_1 = q_2 \\ 0 & q_1 \neq q_2 \end{cases}$$

for each $\sigma \in \Gamma$.

2.2 1.5-way Quantum Finite Automata

A 1.5-way quantum finite automaton is a 2QFA such that $\delta$ is defined as $\delta : Q \times \Gamma \times Q \times \{0,1\} \to C$, and is denoted simply by a QFA from now on. Namely, QFA's cannot move their head $-1$ position but can move 0 or $+1$ position. A sequence of states $q_{i_1}, q_{i_2}, \cdots, q_{i_m}$ is called an ε-cycle if (i) for each q_{i_j}, $1 \leq j \leq m$, $q_{i_j} \in Q_{non}$ and $D(q_{i_j}) = 0$, (ii) $q_{i_1} = q_{i_m}$ and (iii) there is a tape symbol $\sigma \in \Gamma$ such that the coefficient of $|q_{i_{j+1}}\rangle$ in $V_\sigma|q_{i_j}\rangle$ is not zero for all $1 \leq i \leq m-1$. If a QFA $M$ does not include an ε-cycle, then $M$ clearly halts within a linear time. All QFA's in this paper do not include ε-cycles.

Fig. 1 shows an example of a QFA, denoted by M_0. This figure gives a so-called *state-transition diagram.* For example, the transition from q_0 (that is the initial state) means

$$V_{¢}|q_0\rangle = \sqrt{0.4}|q_1\rangle + \sqrt{0.4}|q_2\rangle + \sqrt{0.2}|q_3\rangle.$$

Accepting states are $q_{18}$ and $q_{19}$. Rejecting states are q_3, q_7, q_{13}, q_{16} and $q_{17}$. As for the function $D$, $D(q_i) = 0$ if q_i is an accepting or a rejecting state and $D(q_i) = 1$ otherwise. Following the practice [KW97, AF98], we leave many transitions (e.g., transitions from q_7) undefined. Those transitions may be arbitrary and it is not hard to define them so that the resulting operator will be unitary.

Suppose that we observe M_0 after the first step. Then "reject" (i.e., by q_3) is observed with probability 0.2 and if that happens M_0 halts. $\{q_1, q_2\}$ is observed with probability $(\sqrt{0.4})^2 + (\sqrt{0.4})^2 = 0.8$ and if that happens, the amplitude of each state changes from $\sqrt{0.4}$ to $\frac{\sqrt{0.4}}{\sqrt{0.8}} = \sqrt{0.5}$. This amplitude does not change until $\sqrt{0.5}|q_{18}\rangle + \sqrt{0.5}|q_{19}\rangle$ is reached unless M_0 drops into q_7, q_{13}, q_{16} or $q_{17}$. If M_0 reads $ab$ or $ba$ in the first two steps, then it goes to $\sqrt{0.5}|q_7\rangle + \sqrt{0.5}|q_9\rangle$. If "reject" (by q_7) is observed (with probability 0.5) then M_0 stops. Otherwise M_0 continues its computation from $\frac{\sqrt{0.5}}{\sqrt{0.5}}|q_9\rangle = |q_9\rangle$.

Suppose that the tape $x = x_1x_2x_3x_4$ ($x_i \in \{a,b\}$) is given to this QFA. Then one can see that: (i) If $x_1 = x_2$ and $x_3 = x_4$ then M_0 halts in q_3 with probability 0.2, and if that does not happen then M_0 reaches $\sqrt{0.5}|q_{18}\rangle + \sqrt{0.5}|q_{19}\rangle$. Therefore the probability that "accept" is observed is 0.8 in total. (ii) If $x_1 = x_2$ and $x_3 \neq x_4$ then that probability is 0.4. (iii) If $x_1 \neq x_2$ and $x_3 = x_4$ then it is again 0.4. (iv) If $x_1 \neq x_2$ and $x_3 \neq x_4$ then the probability is 0. Thus this QFA accepts the language $\{x_1x_2x_3x_4 \mid x_i \in \{a,b\},\ x_1 = x_2 \text{ and } x_3 = x_4\}$.

2.3 One-Register Machines

To prove the unsolvability, we shall use the unsolvability of the halting problem for one-register machines. A *one-register machine* (*RM*) consists of the finite control with states $p_0, \cdots, p_{K-1}$ and a single register that can hold an (arbitrarily large) integer. Let $i$ be 2 or 3. For each state, one of the following three instructions is specified:

(1) Multiply the current value, $R$, in the register by $i$ and move to state p_j. This instruction is denoted by (MUL-$i$, p_j).

(2) Divide $R$ by $i$ and move to p_j, denoted by (DIV-$i$, p_j)

(3) Test if $R$ is divisible by $i$. If so, move to p_{j_1} and move to p_{j_2} otherwise. This instruction is denoted by (TEST-$i$, p_{j_1}, p_{j_2}).

p_0 is always the initial state and $p_{K-1}$ is always the only one halting state. Without loss of generality, we can assume that when instruction (2) is executed, the current value $R$ is divisible by $i$. For technical reason we also assume that the instruction associated with the initial state p_0 is always (MUL-2, p_1), which again does not lose generality. The value $R$ does not change when (3) is executed. It is known that RM's are equivalent to Turing machines:

Proposition 1 [Min66]. The halting problem for RM's that start their computation with the initial register value one is not solvable.

3 Main Theorem

The emptiness problem is to decide whether $L(M) = \phi$ for a given automaton $M$. If the emptiness problem is unsolvable for a class of automata, several other problems are also unsolvable for that class of automata including the equivalence problem ($L(M_1) = L(M_2)$?) and the disjointness problem ($L(M_1) \cap L(M_2) = \phi$?). Also an immediate corollary of Theorem 1 is that the emptiness problem is unsolvable for 2QFA's.

Theorem 1. The emptiness problem for QFA's is unsolvable.

Proof. It is shown that there is an algorithm (a Turing machine that always halts) which translates any RM $X$ into a QFA M_X such that $L(M_X) = \phi$ iff $X$ starting with $R = 1$ does not halt. Suppose that the given $X$ has K states, p_0 through $p_{K-1}$. Recall that each state is associated with one of the following six instructions (MUL-2, p_j), (MUL-3, p_j), (DIV-2, p_j), (DIV-3, p_j), (TEST-2, p_{j_1}, p_{j_2}) and (TEST-3, p_{j_1}, p_{j_2}). Since the translation itself is not complicated, it will be enough to only give a detailed description of the target machine M_X. In the following, we first explain what kind of language M_X should accept and then describe how to construct M_X to that goal.

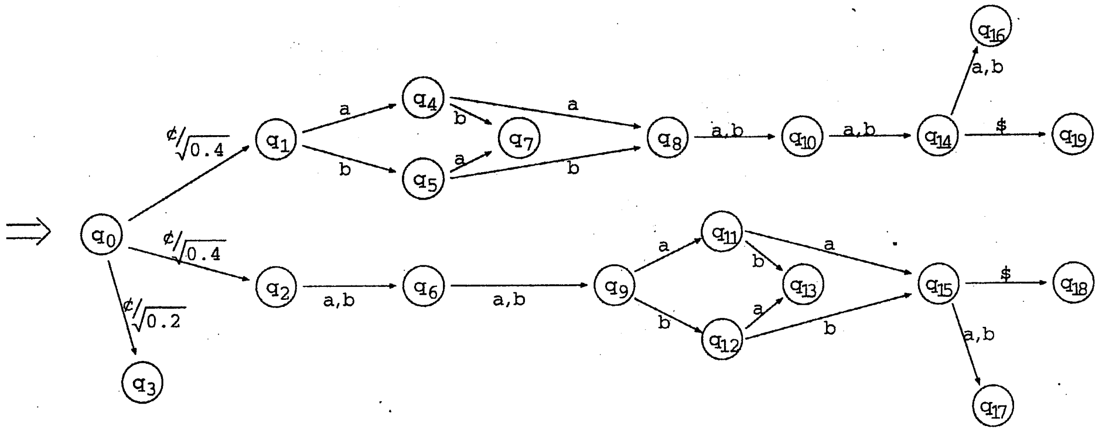

Fig. 1

3.1 The Language to be accepted by M_X

Let L_X be the following language which is determined by the RM X and whose alphabet is $\{p_0, \cdots, p_{K-1}, p_K, p_{K+1}, \sharp, 0\}$ (recall that p_0 through p_{K-1} are X's states). A sequence z contained in L_X has to be of the following form. Its objective is to show a sequence of configurations that change step by step in the course of X's computation:

$$\not{c} z\$ = \not{c}\sharp p_0 0 p_K \sharp p_1 00 p_{K+1} \sharp \cdots \sharp p_i 0 \cdots 0 p_j \sharp \cdots \sharp p_{K-1} 0 \cdots 0 p_l \sharp \$$$

A subsequence surrounded by two $\sharp$'s is called a *block*. The number of 0's in each block shows the value R. In more detail, z is in L_X iff the following three conditions are met:

(1) The first and second blocks must be $p_0 0 p_K$ and $p_1 00\ p_{K+1}$, respectively, regardless of X, where p_K and p_{K+1} are new symbols neither of which appears elsewhere again.

(2) Let $\sharp p_{i_1} 0^{j_1}\ p_{i_2} \sharp p_{i_3}\ 0^{j_2}\ p_{i_4} \sharp$ be any neighboring two blocks excepting the first and the second ones. Then (i) p_{i_1} must be equal to p_{i_4}, and (ii) the relation between the numbers of 0's in these two blocks, j_1 and j_2, and the relation between p_{i_1} and p_{i_3} must be valid with respect to the instruction associated with p_{i_1} (if the instruction is (MUL-2, p_j), for example, then j_2 must be $2 \cdot j_1$ and p_{i_3} must be p_j).

(3) The last block must start with p_{K-1}, i.e., the only one halting state of the RM X.

3.2 Submachines M_1 and M_2

The whole machine M_X consists of two major submachines M_1 and M_2 as shown in Fig. 2. From its initial state q_0, M_X splits into M_1, M_2 and a rejecting state, $q_{0,rej}$, with amplitudes $\sqrt{0.4}$, $\sqrt{0.4}$ and $\sqrt{0.2}$, respectively, just like the example in Fig. 1. Then M_1 tests (i) whether the first two blocks are proper using submachine M_{10} and (ii) whether the $(2i+1)$th and $(2i+2)$th blocks are proper in the sense of (2) above for each $i \geq 1$. To do so, M_1 uses submachines $M_1(p_0), M_1(p_1), \cdots, M_1(p_{K-1})$. M_2 is similar. It checks whether the first block is proper and then whether the $(2i)$th and $(2i+1)$th blocks are properly related. Both M_1 and M_2 also check whether the last block includes the X's halting state p_{K-1}.

Remark 1. M_1 branches into $M_1(p_0)$, $M_1(p_1)$, $\cdots$, $M_1(p_{K-1})$ by reading $p_0, p_1, \cdots; p_{K-1}$, respectively. After testing the properness of two neighboring blocks, M_1 must rejoin into a single state before branching into $M_1(p_0)$, $M_1(p_1)$, $\cdots, M_1(p_{K-1})$ again to test the next two blocks. The existence of p_{i_4} that is the same as p_{i_1} described in (2) above allows us to design this portion of M_1 using unitary operators.

As will be shown later, M_1 reaches "acceptance", i.e., "acceptance" is observed, with probability one (or with probability 0.4 considering the whole machine M_X), if the tape z passes the above test. If z does not pass the test, M_1 can reach "acceptance" with probability roughly $1/N$ or less, as shown later, where we can set N as an arbitrarily large integer. This is the same for M_2. One can see that if the tape x is in L_X, or represents a proper halting sequence of X's configurations, then x passes both tests of M_1 and M_2. That means the whole machine M_X reaches "acceptance" with probability 0.8, i.e., x is accepted. If the tape x is not in L_X then it does not pass at least one of the two tests, which means M_X reaches "acceptance" with probability at most $0.4 + \frac{1}{N} < 0.5$, namely x is not accepted by M_X. Thus M_X recognizes L_X.

3.3 Submachines M_{10} and M_{20}

Fig. 3 illustrates the submachine M_{10} which checks the first and second blocks. The machine is easy since all it has to do is to check whether the beginning portion of the input is equal to the fixed string. As for the notation, $V_{\neg\{\not{c}\}}|q_0\rangle$ means $V_\sigma|q_0\rangle$ for all $\sigma \notin \{\not{c}\}$. All rejecting states but the following exceptions have "rej" in their subscripts. Other rejecting states are $s_{1,j}$ and $t_{1,j}$ for $1 \leq j \leq N-1$. Note that M_{10} includes a split into $\sqrt{N}$ paths from $q_{1,8}$ and the quantum Fourier transform from $r_{1,j}$. We actually do not need those gadgets for the above purpose but we introduced them to adjust that portion to other parts of M_X, which one can see later.

The behavior of M_{10} is similar to the example in section 2.2. If "rejection" by $q_{0,rej}$ is not observed and the beginning portion of z is proper, then M_{10} reaches $|q_{1,8}\rangle$ with amplitude $\sqrt{0.5}$. (More precisely speaking, M_X reaches $\sqrt{0.5}|q_{1,8}\rangle + |\psi\rangle$ where ψ is some superposition of M_2's states such that $\|\ |\psi\rangle\ \| = \sqrt{0.5}$. In this expression, we have omitted the head position since it is not important.) Subsequently, M_{10} goes through the Fourier

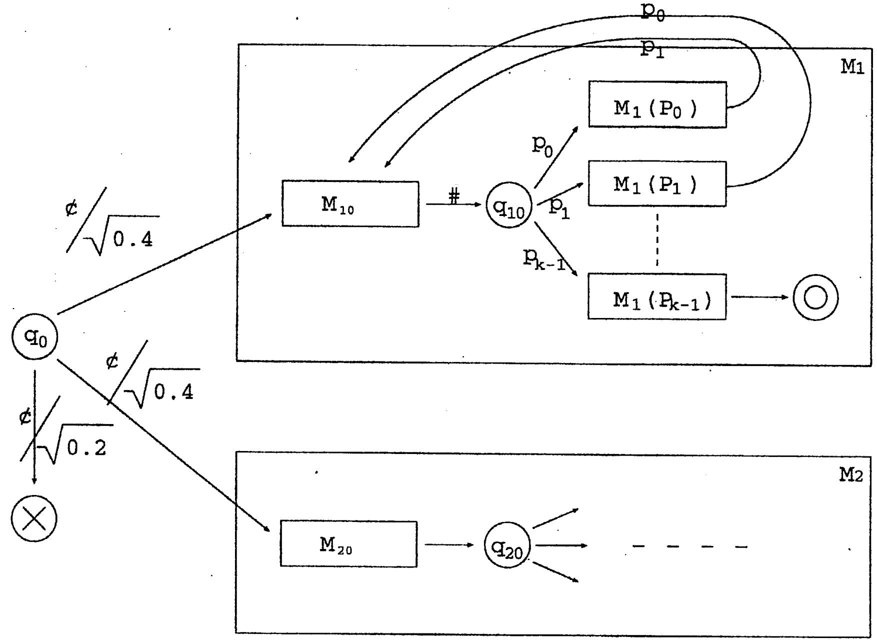

Fig. 2

transform and one can calculate that it reaches $0|s_{1,1}\rangle + 0|s_{1,2}\rangle+\cdots+0|s_{1,N-1}\rangle+\sqrt{0.5}|s_{1,N}\rangle = \sqrt{0.5}|s_{1,N}\rangle$. Then $M_{10}$ goes to $\sqrt{0.5}|q_{1,start}\rangle$. M_{20}'s behavior is similar.

3.4 Submachines $M_1(p_x)$ and $M_2(p_x)$

If the input string is proper, $M_{10}$ reads some p_x, $0 \leq x \leq K-1$, in state $q_{1,start}$ and branches to submachine $M_1(p_x)$. Recall that p_x is associated with one of the six instructions. Suppose that $p_x \neq p_{K-1}$ and the instruction for p_x is (MUL-2, p_y). Then the transition of $M_1(p_x)$ is as shown in Fig. 4. (The transition for other five cases are omitted.) If $p_x = p_{K-1}$ then $M_1(p_{K-1})$ has only to check whether the current block is the last one, whose transition is given in Fig. 5. In what follows, we overview the case that Fig. 4 applies. All the other cases are quite similar.

Suppose that p_x is associated with (MUL-2, p_y). Here we can use the idea of the 2QFA in [KW97], denoted by $M_{KW}$, which accepts $\{a^n b^n \mid n \geq 1\}$. Reading ¢, $M_{KW}$ splits into $N$ paths with amplitude $1/\sqrt{N}$. Along the $j$th path, $M_{KW}$ operates as follows: if $M_{KW}$ reads $a$, its tape head remains stationary for $j$ steps and then moves right. If $M_{KW}$ reads $b$, the head remains stationary for $N-j+1$ steps and then it moves right. Now suppose that $M_{KW}$ is given the input $a^{n_1}b^{n_2}$. Then it turns out that any two distinct computation paths will reach the \$ symbol at the same time if and only if $n_1 = n_2$. To check this simultaneousness, we can use the Fourier transform. Let us look at Fig. 3 again: If the machine is in $\frac{1}{\sqrt{N}}|r_{1,1}\rangle + \frac{1}{\sqrt{N}}|r_{1,2}\rangle+\cdots+\frac{1}{\sqrt{N}}|r_{1,N}\rangle$, i.e., if all the $N$ paths reaches $r_{1,1},\cdots, r_{1,N}$ at the same time, then the machine reaches $|s_{1,N}\rangle$ whose amplitude is 1.0. If only one path reaches, say, $r_{1,1}$ at some time (i.e., the machine is in $\frac{1}{\sqrt{N}}|r_{1,1}\rangle + |\psi\rangle$ where $|\psi\rangle$ does not include $|r_{1,2}\rangle,\cdots,|r_{1,N}\rangle$), then it reaches $\frac{1}{N}\exp(\frac{2\pi i}{N}\cdot 1)|s_{1,1}\rangle + \cdots + \frac{1}{N}\exp(\frac{2\pi i}{N}\cdot(N-1))|s_{1,N-1}\rangle+\frac{1}{N}\exp(\frac{2\pi i}{N}\cdot N)|s_{1,N}\rangle+|\psi'\rangle$ at the next step, i.e., the amplitude of $|s_{1,N}\rangle$ is very small. Recall that all $s_{1,1}$ through $s_{1,N-1}$ are rejecting states.

We can use the same idea to recognize the slightly different language $\{0^n 1 0^n \mid n \geq 1\}$. Actually, the situation is better than before since we do not need the reverse move of the head that was mandatory in $M_{KW}$ when the head crosses the boundary of $a$'s and $b$'s. Namely, $\{0^n 1 0^n \mid n \geq 1\}$ can be recognized by a 1.5QFA. To extend it furthermore into the machine that recognizes $\{0^n 1 0^{2n} \mid n \geq 1\}$ is a straightforward exercise, and that is exactly what we want to do in designing $M_1(p_x)$.

Let us look at Fig. 4 more in detail. (i) The machine starts in $q_{1,start}$ and branches to $M_1(p_x)$ by reading p_x and at the same time splits into $N$ paths. Here, $\{p_*\}$ means $\{p_0,\cdots,p_{K-1}\}$. (ii) Then $M_1(p_x)$ is supposed to read 0's, p_u, $\sharp$, p_w, and again 0's in this order. Suppose that $p_w \neq p_{K-1}$. Then $M_1(p_x)$ "counts" the numbers of the first 0's and the second 0's in each path by the method previously explained. In the middle of this action, it falls into rejecting states if $p_w \neq p_y$. (see (11) in the figure). Note that p_u may be arbitrary. (iii) Subsequently the machine should read p_x again (otherwise falls into rejecting states in (15)) and leaves $M_1(p_x)$, i.e., converges to $|r_{1,j}\rangle$ in $(14-a)$. Note that the $N$ paths are not converged at this moment yet. In the next step the Fourier transform is applied in $(16-a)$ (which already appeared in Fig. 3 but is repeated here).

Claim 1. If everything is good, then the $N$ computation paths arrive at $r_{1,1},\cdots,r_{1,N}$ at the same time. That means M_1 reaches $\sqrt{0.5}|s_{1,N}\rangle$. Otherwise, if the numbers of the first 0's and the second 0's are not proper for example, then each of the $N$ computation paths arrives at $s_{1,N}$ at all different steps with amplitude $1/N$ each time.

If $p_w = p_{K-1}$ and if this $p_{K-1}$ is the proper successor of p_x, then the second block must be the final one. Therefore $M_1(p_x)$ goes to a different routine at $(10-b)$. In this case, the Fourier transform is applied in $(16-b)$ and if everything is good then it reaches $\sqrt{0.5}|t_{1,N}\rangle$ instead of $\sqrt{0.5}|s_{1,N}\rangle$. In the next step it reaches $\sqrt{0.5}|q_{1,acc}\rangle$, where $q_{1,acc}$ is one of the four accepting states of M_X. (The others are $q_{2,acc}$ in M_2 that is the counterpart of $q_{1,acc}$, $q_{1,11,acc}$ in $M_1(p_{K-1})$ and $q_{2,11,acc}$ in $M_2(p_{K-1})$.) We omit the description of $M_2(p_x)$, which is almost the same as $M_1(p_x)$ (but of course we need new states).

There is one thing we should be careful for. There are two instructions (TEST-2, p_{j_1}, p_{j_2}) and (TEST-3, p_{j_1}, p_{j_2}) which require us to check whether the number of 0's is divisible by two and three, respectively. Checking if it is divisible by two has no problem but checking if it is divisible by three needs care. Intuitively, this can be done using three states, say p_0, p_1 and p_2. If the number is divisible, then the machine will be always in p_0. Otherwise, if not divisible, then the machine may be p_1 *or* p_2; it is not possible to define transition from those two states to a single state by the same symbol.

3.5 Analysis

Let $z = \sharp B_1 \sharp B_2 \sharp B_3 \sharp \cdots \sharp B_i \sharp \cdots \sharp B_{2m} \sharp$, where B_i is the ith block. Suppose first that all B_i's are proper and that M_X does not stop at $q_{0,rej}$. Then, for all $1 \leq i \leq m-1$, submachine M_1 always reaches $\sqrt{0.5}|s_{1,N}\rangle$ after it reads B_{2i}. Then M_1 reaches $\sqrt{0.5}|t_{1,N}\rangle$ after it reads $B_{2m}$, and then reaches $\sqrt{0.5}|q_{1,acc}\rangle$ finally. Submachine M_2 always reaches $\sqrt{0.5}|s_{2,N}\rangle$ also after it reads B_{2i+1} for all $0 \leq i \leq m-1$ and finally $B_{2m}$ is read by $M_2(p_{K-1})$, which reaches $\sqrt{0.5}|q_{2,11,acc}\rangle$ eventually. Thus the probability that "acceptance" is observed is $(1-0.2) \cdot ((\sqrt{0.5})^2 + (\sqrt{0.5})^2) = 0.8$ and $z$ is accepted by M_X. If $z$ includes an odd number of blocks, then nothing differs excepting that the roles of M_1 and M_2 are switched. As a result, we can conclude that if $z$ is in L_X then $z$ is accepted by M_X.

Now suppose that B_1 through $B_{i-1}$ are proper but B_i is not for some $i \geq 1$. There are two cases:

(1) B_i is improper regardless of the number of 0's; for example, its syntax is different from $p_x 0 \cdots 0 p_y$ or p_x or p_y is not what is supposed to be. Then one can see that at least one of M_1 and M_2 can detect that improperness and all the $N$ paths fall into rejecting states. That means the overall probability that "reject" is observed is at least $0.2 + (1-0.2) \cdot (\sqrt{0.5})^2 = 0.6$, namely, $z$ is rejected.

(2) The number of 0's are inconsistent between $B_{i-1}$ and B_i. Then either M_1 or M_2, say, M_1, which checks $B_{i-1}$ and B_i, can detect it as follows. Recall that M_1 splits into the $N$ paths each of which has amplitude $\sqrt{0.5}/\sqrt{N}$. As described in Claim 1, each path, the jth path, reaches $\sqrt{0.5}(\frac{1}{N}\exp(\frac{2\pi i}{N}j \cdot 1)\ |s_{1,1}\rangle + \frac{1}{N}\exp(\frac{2\pi i}{N}j \cdot 2)|s_{1,2}\rangle + \cdots + \frac{1}{N}\exp(\frac{2\pi i}{N}j \cdot N)|s_{1,N}\rangle)$ at different time. Recall that $s_{1,1}$ through $s_{1,N-1}$ are all rejecting states. Hence, when the fastest path reaches there, the probability that "reject" is observed is $(\sqrt{0.5}\frac{1}{N})^2$ (for $s_{1,1}$) $+ (\sqrt{0.5}\frac{1}{N})^2$ (for $s_{1,2}$) $+ \cdots + (\sqrt{0.5}\frac{1}{N})^2$ (for $s_{1,N-1}$) $= (N-1)\cdot(\sqrt{0.5}\frac{1}{N})^2 = 0.5\frac{N-1}{N^2}$. If "reject" is not observed, then the amplitude of each remaining path is increased by $\frac{1}{\sqrt{1-0.5\frac{N-1}{N^2}}}$ times. Therefore when the second fastest path reaches the same point, the probability that "reject" is observed is that

$$(N-1)\cdot(\sqrt{0.5}\frac{\frac{1}{N}}{\sqrt{1-0.5\frac{N-1}{N^3}}})^2 = 0.5(\frac{N-1}{N^2-0.5(N-1)}).$$

It then follows that the probability that "reject" is observed for the fastest path or the second fastest path is

$$0.5\frac{N-1}{N^2}+(1-0.5\frac{N-1}{N^2})0.5(\frac{N-1}{N^2-0.5(N-1)}) = 2\times(0.5\frac{N-1}{N^2}).$$

Namely, the probability is increased by $0.5\frac{N-1}{N^2}$ when the second fastest path arrives. It is not hard to see that the same amount of probability is also added when the third fastest path reaches there and so on. As a result, when the $N$ paths reach there, the probability that "reject" is observed for at least one path, namely that M_1 halts so far is

$$0.5\frac{N-1}{N^2} \cdot N = 0.5\frac{N-1}{N}.$$

Since "reject" is observed with probability 0.2 before M_x branches to M, the overall probability that "reject" is observed so far is

$$0.2 + (1-0.2) \cdot 0.5 \cdot \frac{N-1}{N} = 0.2 + 0.4\frac{N-1}{N}.$$

If $N$ is sufficiently large, this value is greater than 0.5 and $z$ is rejected. Thus we can conclude that M_X recognizes L_X.

Finally we should mention how we have designed M_X so as to be unitary. The basic idea is to make M_X reversible everywhere but the portions of the Fourier transform. This concludes the proof of Theorem 1.

4 Concluding Remarks

Our study in this paper would reveal several interesting questions yet to be resolved: (i) Now we know that 1.5QFA's can accept considerably high-class languages up to at least those which cannot be accepted by one-way stack automata. Then what is a well-known class of languages that can contain all the languages accepted by 1.5QFA's? Furthermore, does the answer to this question differ much if 1.5 QFA's are replaced by 2QFA's? (ii) A more specific question is whether or not there is a class of one-way linear-time conventional machines (like alternating off-line TMs) which is at least as powerful as 1.5 QFA's without ε-cycles. (iii) We were not able to discuss negative aspects of 1.5QFA's in this paper. Our conjecture is that there are regular languages that cannot be accepted by any 1.5QFA's even if we allow ε-cycles. (iv) Our 1.5QFA's in this paper have a quite large error probability. It does not appear to be easy to reduce it significantly as long as depending on the current approach of computing conjunctiveness (see the example in Section 2.2).

Reference

[AF98] 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 (to appear)*, 1998.

[DS89] C. Dwork and L. Stockmeyer, "On the power of 2-way probabilistic finite automata," *Proceedings of the 30th IEEE Conference on Foundations of Computer Science*, 480-485, 1989.

[Fre81] R. Freivalds, "Probabilistic two-way machines," *LNCS*, 188, 33-45, 1981.

[Gro96] L. Grover, "A fast quantum mechanical algorithm for database search," *Proceedings of the 28th ACM Symposium on Theory of Computing*, 212-219, 1996.

[HU79] J. Hopcroft and J. Ullman, *An Introduction to Automata Theory, Languages and Computation,* Addison-Wesley, 1979.

[HU68] J. Hopcroft and J. Ullman, " Deterministic stack automata and the quotient operator," *JCSS,* 2:1, 1-12, 1968.

[KW97] A. Kondacs and J. Watrous, "On the power of quantum finite state automata," *Proceedings of the 38th IEEE Conference on Foundations of Computer Science,* 66-75, 1997.

[Min66] M. Minsky, *Computation: Finite and infinite machines,* Prentice-Hall, 1966.

[Sho94] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," *Proceedings of the 35th IEEE Symposium on Foundations of Computer Science,* 124-134, 1994.

$$¢\cdots\sharp p_x 000 p_* \sharp p_y 000000 p_x \sharp \cdots \$$$

or $¢\cdots\sharp p_x 000 p_* \sharp p_{K-1} 000000 p_x \sharp\$$

(1) $V_{p_x}|q_{1,start}\rangle = \frac{1}{\sqrt{N}}\Sigma_{j=0}^{N}|r_{1,j,0,p_x,1}\rangle$

(2) $V_{\neg\{p_*\}}|q_{1,start}\rangle = |q_{1,start,rej}\rangle$

(3) $V_0|r_{1,j,0,p_x,1}\rangle = |r_{1,j,2j,p_x,1}\rangle$

(4) $V_0|r_{1,j,k,p_x,1}\rangle = |r_{1,j,k-1,p_x,1}\rangle,\ 1 \leq k \leq 2j$

(5) $V_{p_*}|r_{1,j,0,p_x,1}\rangle = |r_{1,j,0,p_x,2}\rangle$

(6) $V_{p_{K+1}}|r_{1,j,0,p_x,1}\rangle = |r_{1,j,0,p_x,2}\rangle$

(7) $V_{\neg\{0,p_{K+1},p_*\}}|r_{1,j,0,p_x,1}\rangle = |r_{1,j,0,p_x,1,rej}\rangle$

(8) $V_\sharp|r_{1,j,0,p_x,2}\rangle = |r_{1,j,0,p_x,3}\rangle$

(9) $V_{\neg\{\sharp\}}|r_{1,j,0,p_x,2}\rangle = |r_{1,j,0,p_x,2,rej}\rangle$

(10 − a) $V_{p_y}|r_{1,j,0,p_x,3}\rangle = |r_{1,j,0,p_x,4}\rangle$ (if $p_y \neq p_{K-1}$)

(10 − b) $V_{p_{K-1}}|r_{1,j,0,p_x,3}\rangle = |r_{1,j,0,p_x,5}\rangle$ (if $p_y = p_{K-1}$)

(11) $V_{\neg\{p_y\}}|r_{1,j,0,p_x,3}\rangle = |r_{1,j,0,p_x,3,rej}\rangle$

(12) $V_0|r_{1,j,0,p_x,f}\rangle = |r_{1,j,N-j+1,p_x,f}\rangle,\ f = 4,5$

(13) $V_0|r_{1,j,k,p_x,f}\rangle = |r_{1,j,k-1,p_x,f}\rangle,\ 1 \leq k \leq N-j+1$

(14 − a) $V_{p_x}|r_{1,j,0,p_x,4}\rangle = |r_{1,j}\rangle$

(14 − b) $V_{p_x}|r_{1,j,0,p_x,5}\rangle = |r_{1,j,1}\rangle$

(15) $V_{\neg\{0,p_x\}}|r_{1,j,0,p_x,f}\rangle = |r_{1,j,0,p_x,f,rej}\rangle$

(16 − a) $V_\sharp|r_{1,j}\rangle = \frac{1}{\sqrt{N}}\Sigma_{l=1}^{N}\exp(\frac{2\pi i}{N}jl)|s_{1,l}\rangle,\ 1 \leq j \leq N$

(16 − b) $V_\sharp|r_{1,j,1}\rangle = \frac{1}{\sqrt{N}}\Sigma_{l=1}^{N}\exp(\frac{2\pi i}{N}jl)|t_{1,l}\rangle,\ 1 \leq j \leq N$

(17 − a) $V_{\neg\{\sharp\}}|r_{1,j}\rangle = |r_{1,j,rej}\rangle$

(17 − b) $V_{\neg\{\sharp\}}|r_{1,j,1}\rangle = |r_{1,j,1,rej}\rangle$

(18) $V_\sharp|s_{1,N}\rangle = |q_{1,start}\rangle$

(19 − a) $V_\$|t_{1,N}\rangle = |q_{1,acc}\rangle$

(19 − b) $V_{\neg\{\$\}}|t_{1,N}\rangle = |t_{1,rej}\rangle$

(20) $D(r_{1,j,0,p_x,m}) = +1,\ 1 \leq m \leq 5$

(21) $D(r_{1,j,0,p_x,m,rej}) = 0$

(22) $D(r_{1,j,k,p_x,m}) = 0, k \neq 0,\ m = 1,4,5$

(23) $D(q_{1,start}) = +1$

(24) $D(s_{1,l}) = 0,\ 1 \leq l \leq N$

(25) $D(t_{1,l}) = +1,\ 1 \leq l \leq N$

(26) $D(r_{1,j}) = +1,\ 1 \leq j \leq N$

(27) $D(r_{1,j,rej}) = 0$

(28) $D(r_{1,j,1}) = +1$

(29) $D(r_{1,j,1,rej}) = 0$

(30) $D(q_{1,acc}) = 0$

(31) $D(t_{1,rej}) = 0$

Fig. 4

$$¢\sharp p_1 0 p_K \sharp p_2 00 p_{K+1} \sharp \cdots \$$$

$V_¢|q_0\rangle = \sqrt{0.4}|q_{1,0}\rangle + \sqrt{0.4}|q_{2,0}\rangle + \sqrt{0.2}|q_{0,rej}\rangle$
$V_{\neg\{¢\}}|q_0\rangle = |q_{0,rej}\rangle$
$V_\sharp|q_{1,0}\rangle = |q_{1,1}\rangle$
$V_{\neg\{\sharp\}}|q_{1,0}\rangle = |q_{1,0,rej}\rangle$
$V_{p_1}|q_{1,1}\rangle = |q_{1,2}\rangle$
$V_{\neg\{p_1\}}|q_{1,1}\rangle = |q_{1,1,rej}\rangle$
$V_0|q_{1,2}\rangle = |q_{1,3}\rangle$
$V_{\neg\{0\}}|q_{1,2}\rangle = |q_{1,2,rej}\rangle$
$V_{p_K}|q_{1,3}\rangle = |q_{1,4}\rangle$
$V_{\neg\{p_K\}}|q_{1,3}\rangle = |q_{1,3,rej}\rangle$
$V_\sharp|q_{1,4}\rangle = |q_{1,5}\rangle$
$V_{\neg\{\sharp\}}|q_{1,4}\rangle = |q_{1,4,rej}\rangle$
$V_{p_2}|q_{1,5}\rangle = |q_{1,6}\rangle$
$V_{\neg\{p_2\}}|q_{1,5}\rangle = |q_{1,5,rej}\rangle$
$V_0|q_{1,6}\rangle = |q_{1,7}\rangle$
$V_{\neg\{0\}}|q_{1,6}\rangle = |q_{1,6,rej}\rangle$
$V_0|q_{1,7}\rangle = |q_{1,8}\rangle$
$V_{\neg\{0\}}|q_{1,7}\rangle = |q_{1,7,rej}\rangle$
$V_{p_{K+1}}|q_{1,8}\rangle = \frac{1}{\sqrt{N}}\Sigma_{j=1}^{N}|r_{1,j}\rangle$
$V_{\neg\{p_{K+1}\}}|q_{1,8}\rangle = |q_{1,8,rej}\rangle$
$V_\sharp|r_{1,j}\rangle = \frac{1}{\sqrt{N}}\Sigma_{l=1}^{N}\exp(\frac{2\pi i}{N}jl)|s_{1,l}\rangle,\ 1 \leq j \leq N$
$V_{\neg\{\sharp\}}|r_{1,j}\rangle = |r_{1,j,rej}\rangle$
$V_\sharp|s_{1,N}\rangle = |q_{1,start}\rangle$

$D(q_{1,i}) = +1,\ 0 \leq i \leq 8$
$D(q_{1,i,rej}) = 0$
$D(r_{1,j}) = +1,\ 1 \leq j \leq N$
$D(r_{1,j,rej}) = 0$
$D(s_{1,l}) = 0,\ 1 \leq l \leq N$
$D(q_{1,start}) = +1$
$D(q_{0,rej}) = 0$

Fig. 3

$$¢\cdots\sharp p_{K-1} 0 \cdots 0 p_* \sharp\$$$

$V_{p_{K-1}}|q_{1,start}\rangle = |q_{1,9}\rangle$
$V_{\neg\{p_*\}}|q_{1,start}\rangle = |q_{1,start,rej}\rangle$
$V_0|q_{1,9}\rangle = |q_{1,9}\rangle$
$V_{P_*}|q_{1,9}\rangle = |q_{1,10}\rangle$
$V_{\neg\{0,p_*\}}|q_{1,9}\rangle = |q_{1,9,rej}\rangle$
$V_\sharp|q_{1,10}\rangle = |q_{1,11}\rangle$
$V_{\neg\{\sharp\}}|q_{1,10}\rangle = |q_{1,10,rej}\rangle$
$V_\$|q_{1,11}\rangle = |q_{1,11,acc}\rangle$
$V_{\neg\{\$\}}|q_{1,11}\rangle = |q_{1,10,rej}\rangle$

$D(q_{1,i}) = +1,\ i = 9,10,11$
$D(q_{1,i,rej}) = 0,\ i = 9,10,11$
$D(q_{1,11,acc}) = 0$

Fig. 5