Vol.2009-AL-127 No.4
2009/11/27

One-Way Quantum Finite Automata and Advice

(Preliminary Version)

Tomoyuki Yamakami[†1]

Abstract: We show containments and separations among language families defined by bounded-error one-way quantum finite automata whose computations are further helped by various types of "advice." Beside standard deterministic advice, we also study randomized advice and quantum advice. The presence of advice demands quite different approaches toward an analysis of the computational power of underlying one-way quantum finite automata. We discover new machine-independent characterizations and develop new proof techniques, which lead us to obtain the desired containments and separations.

Keywords: quantum finite automaton, reversible finite automaton, advice, randomized advice, quantum advice

1. Background, Motivations, and Challenges

Considerable attentions have been paid to a computational model of quantum finite automata, in hopes of achieving a much better understanding of quantum-mechanical computations. The notion of quantum finite automata[6),7)] was conceived as early as mid 1990s as a quantum-mechanical extension of probabilistic finite automata with a coin-flipping mechanism of determining next moves. With the current status of technology, prototypes of quantum computers are still limited in its operational ability. In case where the usage of memory space is severely limited, quantum finite automata may be an appropriate model for memoryless quantum computations. Simplicity of such a model has, since its introduction, helped investigate the behavioral characteristics of quantum computations.

Of various types of quantum finite automata, we are focused on *measure-many one-way quantum finite automata* (or 1qfa's, in short), each of which scans each tape cell by moving a tape head only in one direction (without stopping the tape head) and also performs a *(projection) measurement*, immediately after every head move until it scans the right endmarker. It is crucial to allow 1qfa's to err, because otherwise quantum finite automata are merely as powerful as one-way (deterministic) reversible finite automata (or 1rfa's, in short).

During an early period of study, a number of intriguing features of quantum finite automata have been revealed. As Kondacs and Watrous[6)] proved, for instance, a certain regular language is recognized by no 1qfa's with bounded-error probability. By Brodsky and Pippenger[2)], no bounded-error 1qfa recognizes languages accepted by minimal finite automata that lack a so-called *partial order condition* (see Section 5). Ambainis and Freivalds[1)] demonstrated that every language recognized by 1qfa's with success probability higher than 7/9 can be recognized even by 1rfa's. Moreover, quantum finite automata can be built more state-efficiently than, e.g., deterministic finite automata are[2)]. The model of quantum finite automaton has been further applied to, for instance, interactive proof systems[8)].

A notion of finite (state) automata equipped with supplemental information, known as *advice*, has been studied in a wide range of the literature. A piece of advice includes additional data, beside a standard input, which depends only on the input size[5)]. A series of recent studies[3),10)–13)] on one-way finite automata have revealed delicate roles of advice. Such advised-computational models have immediate connections to other fields, including one-way communication, random access coding, and two-player zero-sum games. A central question concerning advice is: how can we encode necessary information into a piece of advice before a computation starts and how can we decode and utilize such information stored inside the advice, as a computation proceeds step by step?

As a bold step, we shall examine the roles of advice, particularly given to bounded-error 1qfa's. An immediate advantage of taking such advice is the elimination of endmarkers. Note that our underlying model of 1qfa's require every input string to be surrounded by two endmarkers, ¢ (left endmarker) and \$ (right endmarker), given on an input tape. Earlier, Brodsky and Pippenger[2)] demonstrated that the left-endmarker can be eliminated without tampering the machine's computational power. By marking the last symbol of the input string by a piece of advice, we can also eliminate the right endmarker.

†1 Department of Information Science, University of Fukui

There are also numerous challenges on advice issues. The presence of advice often makes an analysis of underlying computations quite difficult and thus demands different kinds of proof techniques. For a quick example, a standard *pumping lemma*—a typical proof technique showing the non-regularity of a given language—is not quite serviceable to advised computations; therefore, we need to develop other tools (e.g., a swapping lemma[11]) for them. On a similar light, advice makes 1qfa's violate the aforementioned partial order condition criteria, making a proof technique of Kondacs and Watrous[6] inapplicable to a separation between regular languages and languages accepted by bounded-error advised 1qfa's. These difficulties motivate us to seek different kinds of proof techniques to show our desired separation result.

There is another type of advice studied for classical finite automata. Instead of giving a single advice string, we can probabilistically generate many advice strings and feed them to an underlying finite automaton at random so that they can elevate their computational ability. Such advice is known as *randomized advice*. An extreme case study[13] demonstrated that randomized advice provides unbounded-error probabilistic one-way finite automata with the full language-recognition power. Even one-way deterministic finite automata (1dfa's, in short) receive more benefits from randomized advice than deterministic advice[13]. A natural question that must arise is: can randomized advice also enhance a computational power of its underlying quantum computation more than deterministic advice does?

Quantum computation is capable of handling *quantum advice*, which is given as a pure quantum state in juxtaposition to original inputs. How resourceful can quantum advice be? For an effective use of quantum advice, however, we must consider a slightly non-conventional use of a 1qfa's input tape, authorizing an "alteration" of advice strings. This is necessary because, as we shall demonstrate later, quantum advice is merely reduced to randomized advice in computational power as far as an underlying machine does not alter any advice string. Allowing such a modification, we shall introduce a model of "rewritable" 1qfa's (see Section 7). This highlights a stark contrast between classical and quantum computations. While such an alteration does not affect one-way classical computation, one-way quantum computation can make use of the alteration even at the end of the computation.

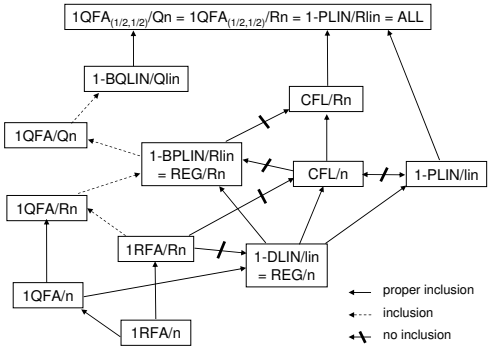

Fig. 1 A hierarchy of advised language families. Containments and separations associated with quantum finite automata are new in this paper.

In this paper, we shall address the issues discussed above and offer some reasonable answers to them. As shown in Fig. 1, we shall show new inclusions and separations of those advised language families to existing advised families. In the figure, "ALL" indicates the collection of all languages. To prove these new results, we shall give a new structural characterization of advised 1rfa's and also a new structural property of advised 1qfa's, which helps us separate, e.g., 1QFA/n from REG/n. These results are quite interesting on their own light.

2. Basic Terminology

We wish to present quick descriptions of notions and notations to read through this paper. Let $\mathbb{N}$ be the set of all *nonnegative integers*. For any pair $m, n \in \mathbb{N}$ with $m \leq n$, the *integer interval* $[m, n]_{\mathbb{Z}}$ denotes the set $\{m, m+1, m+2, \ldots, n\}$, and $[n]$ is shorthand for $[1, n]_{\mathbb{Z}}$. An *alphabet* Σ is a finite nonempty set and a *string* over Σ is a series of symbols taken from Σ. In particular, the *empty string* is always denoted λ. The *length* $|x|$ of a string x is the total number of symbols in x. For any string x and any number $n \in \mathbb{N}$, $Pref_n(x)$ means the substring consisting of the first n symbols of x whenever $|x| \geq n$.

A *one-tape two-way one-head off-line Turing machines* is a sextuple $(Q, \Sigma, \delta, q_0, Q_{acc}, Q_{rej})$, where Q is a finite set of inner states, Σ is an input

alphabet, δ is a *transition function*, q_0 ($\in Q$) is the *initial state*, Q_{acc} ($\subseteq Q$) is a set of *accepting states*, and Q_{rej} ($\subseteq Q$) is a set of *rejecting states*. Set Q_{non} to be $Q - (Q_{acc} \cup Q_{rej})$ for simplicity. For our convenience, write $\check{\Sigma}$ for $\Sigma \cup \{¢, \$\}$. We say that M *runs in linear time* if the longest computation path (even in a case of probabilistic computation) of M on every input x of length n is bounded from above by a certain fixed linearly-bounded function in n (see Tadaki et al.[10] for details). When δ is deterministic (probabilistic, resp.), we succinctly call M a *1DTM* (*1PTM*, resp.).

Finite (state) automata are a special case of those one-tape linear-time Turing machines, together with a restriction that the machines cannot alter tape contents. When M is a one-way deterministic (probabilistic, resp.) finite automaton that always move their tape heads rightward, we call M a *1dfa* (*1pfa*, resp.). Let REG, CFL, and DCFL denote, respectively, the family of *regular languages*, the family of *context-free languages*, and the family of *deterministic context-free languages*. See, e.g., a textbook[4] for their fundamental properties.

We say that a 1PTM M has *bounded error probability* if there exists a constant $\varepsilon \in [0, 1/2)$ satisfying that, for every input string x, either $\mathrm{Prob}_M[M(x) = 1] \geq 1 - \varepsilon$ or $\mathrm{Prob}_M[M(x) = 0] \geq 1 - \varepsilon$, where the probability is taken according to M's internal random process. Let 1-DLIN (1-BPLIN, 1-PLIN, resp.) denote the collection of all languages that are recognized by 1DTM (1PTM with bounded error, 1PTM with unbounded error, resp.) in time $O(n)$, where n is an input length.

To feed supplemental information, beside input strings, to one-tape machines, we use the "track" notation $\left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$ of Tadaki et al.[10].

An *advice function* is a function mapping $\mathbb{N}$ to Γ^*, where Γ is an alphabet, called an *advice alphabet*. The advised language family[⋆1] REG/n (1-DLIN/lin, resp.) of Tadaki et al.[10] is the collection of all languages L over certain alphabets Σ satisfying the following condition: there exist a 1dfa (a linear-time 1DTM, resp.) M, an advice alphabet Γ, and an advice function $h : \mathbb{N} \to \Gamma^*$ for which (i) for every length $n \in \mathbb{N}$, $|h(n)| = n$ ($|h(n)| = O(n)$, resp.) and (ii) for every string $x \in \Sigma^*$, $x \in L$ iff M accepts $\left[\begin{smallmatrix} x \\ h(|x|) \end{smallmatrix}\right]$. Similarly, CFL/$n$, 1-BPLIN/$lin$, and 1-PLIN/$lin$ are defined[?]. The next lemma, shown by Yamakami[13], gives a machine-independent characterization of languages in REG/n.

Lemma 2.1 *For any language S over an alphabet Σ, the following two statements are logically equivalent. Let $\Delta = \{(x, n) \in \Sigma^* \times \mathbb{N} \mid |x| \leq n\}$. (1) S is in* REG/n. *(2) There is an equivalence relation $\equiv_S$ over Δ such that (i) the total number of equivalence classes in $\Delta/\equiv_S$ is finite and (ii) for any index $n \in \mathbb{N}$ and any two strings $x, y \in \Sigma^*$ with $|x| = |y| \leq n$, the following holds: $(x, n) \equiv_S (y, n)$ iff $S(xz) = S(yz)$ for all strings z satisfying $|xz| = n$.*

In this paper, a *probability ensemble* means an infinite series $\{D_n\}_{n\in\mathbb{N}}$ of probability distributions, in which each D_n maps Γ^n to the unit real interval $[0, 1]$, where Γ is a given alphabet.

3. Usefulness of Advice

Since its introduction, the usefulness of advice has been demonstrated for various models of underlying computations. Following this line of study, we first pay our attention to bounded-error quantum computation that takes standard deterministic advice.

For our purpose, we shall give a brief description of quantum finite automata. In this paper, we consider only quantum finite automata with one-way head moves with bounded-error probability, provided that, at each step, they perform a (projection) measurement to check whether they enter halting states. Such automata are known as *measure-many one-way quantum finite automata* (or 1qfa's, in short).

Formally, a 1qfa M is a sextuple $(Q, \Sigma, \{U_\sigma\}_{\sigma\in\check{\Sigma}}, q_0, Q_{acc}, Q_{rej})$, where each *time-evolution operator* U_σ is a unitary operator acting on the Hilbert space $E_Q = span\{|q\rangle \mid q \in Q\}$ of dimension $|Q|$. The series $\{U_\sigma\}_{\sigma\in\check{\Sigma}}$ describe the *time evolution* of M. Let P_{acc}, P_{rej}, and P_{non} be respectively the projections of E_Q onto the subspaces $E_{acc} = span\{|q\rangle \mid q \in Q_{acc}\}$, $E_{rej} = span\{|q\rangle \mid q \in Q_{rej}\}$, and $E_{non} = span\{|q\rangle \mid q \in Q_{non}\}$. For any symbol $\sigma \in \check{\Sigma}$, we define a *transition operator* T_σ as $T_\sigma = P_{non}U_\sigma$. We expand this operator to its extended one T_x

⋆1 Damm and Holzer[3] took a different approach to advised computations; however, their definitions and ours are equivalent for, e.g., polynomial time-bounded computations.

for each fixed string $x = \sigma_1\sigma_2\cdots\sigma_n$ in $\check{\Sigma}^*$ by setting $T_x = T_{\sigma_n}T_{\sigma_{n-1}}\cdots T_{\sigma_2}T_{\sigma_1}$. Here is a helpful lemma concerning this extended operator T_x.

Lemma 3.1 *For any two quantum states $|\phi\rangle, |\phi'\rangle \in E_{non}$ and any string $x \in \check{\Sigma}^*$, $|||\phi\rangle - |\phi'\rangle||^2 - ||T_x(|\phi\rangle - |\phi'\rangle)||^2 \leq \frac{3}{2}[(|||\phi\rangle||^2 - ||T_x|\phi\rangle||^2) + (|||\phi'\rangle||^2 - ||T_x|\phi'\rangle||^2)]$.*

To describe precisely the *time-evolution* of M, let us consider a new Hilbert space $\mathcal{S}$ spanned by the basis vectors in $\ell_2(Q) \times \mathbb{R} \times \mathbb{R}$, where $\ell_2(Q) = \{|q\rangle \mid q \in Q\}$. We define a *norm* of an element $\psi = (|\phi\rangle, \gamma_1, \gamma_2)$ in $\mathcal{S}$ to be $||\psi|| = (|||\phi\rangle||^2 + |\gamma_1| + |\gamma_2|)^{1/2}$. With the space $\mathcal{S}$, we extend the aforementioned transition operator T_σ to $\hat{T}_\sigma$ by defining $\hat{T}_\sigma(|\phi\rangle, \gamma_1, \gamma_2) = (T_\sigma|\phi\rangle, \gamma_1 + ||P_{acc}U_\sigma|\phi\rangle||^2, \gamma_2 + ||P_{rej}U_\sigma|\phi\rangle||^2)$. For an arbitrary string $x = \sigma_1\sigma_2\cdots\sigma_n$ in $\check{\Sigma}^*$, we further define $\hat{T}_x$ as $\hat{T}_{\sigma_n}\hat{T}_{\sigma_{n-1}}\cdots\hat{T}_{\sigma_1}$. Notice that this extended operator $\hat{T}_x$ may not be a linear operator in general; however, it satisfies the following useful properties.

Lemma 3.2 *(1) For any two elements $\psi, \psi' \in \mathcal{S}$, it holds that $||\psi + \psi'|| \leq ||\psi|| + ||\psi'||$.*

(2) For any two elements $\psi, \psi' \in \mathcal{S}$ and any string $x \in \check{\Sigma}^*$, $||\hat{T}_x\psi - \hat{T}_x\psi'|| \leq \sqrt{2}||\psi - \psi'||$.

(3) For any two elements $\psi, \psi' \in \mathcal{S}$ and any string $x \in \check{\Sigma}^*$, let $\psi = (|\phi\rangle, \gamma_1, \gamma_2)$ and $\psi' = (|\phi'\rangle, \gamma_1', \gamma_2')$. Then, $||\hat{T}_x\psi - \hat{T}_x\psi'||^2 \geq ||\psi - \psi'||^2 - 3(|||\phi\rangle - |\phi'\rangle||^2 - ||T_x(|\phi\rangle - |\phi'\rangle)||^2)$.

Recall that an input to machines must be of the form $\mathfrak{c}x\$ = \sigma_1\sigma_2\cdots\sigma_{n+2}$ with $\sigma_1 = \mathfrak{c}$, $\sigma_{n+2} = \$$, and $x \in \Sigma^n$. The *acceptance probability* of M on x at step i $(1 \leq i \leq n+2)$, denoted $p_{acc}(i)$, is $||P_{acc}U_{\sigma_i}|\phi_{i-1}\rangle||^2$, where $|\phi_0\rangle = |q_0\rangle$ and $|\phi_i\rangle = T_{\sigma_i}|\phi_{i-1}\rangle$. The *acceptance probability* of M on x, denoted $p_{acc}(x)$, is $\sum_{i=1}^{n+2} p_{acc}(i)$. Similarly, we define the *rejection probabilities* $p_{rej}(i)$ and $p_{rej}(x)$ using P_{rej} instead of P_{acc}. Using these notations, it follows that $\hat{T}_{\mathfrak{c}x\$}(|q_0\rangle, 0, 0) = (|\phi_{n+2}\rangle, p_{acc}(x), p_{rej}(x))$. In Section 7, however, we shall expand this current definition of 1qfa's using a new device called *rewritable tape tracks*.

Let $a(n)$ and $b(n)$ be any function from $\mathbb{N}$ to the unit real interval $[0,1]$. We write $1\mathrm{QFA}_{(a(n),b(n))}$ for the collection of all languages L recognized by 1qfa's M with the following criteria: if $x \in L$ then M accepts x with probability at least $a(|x|)$, and if $x \notin L$ then M rejects x with probability at least $b(|x|)$. Finally, let 1QFA denote $\bigcup_{\epsilon>0} 1\mathrm{QFA}_{(1/2+\epsilon,1/2+\epsilon)}$.

Note that 1QFA is closed under complementation, inverse homomorphism, and word quotient[2] and that it is not closed under homomorphism[2].

Next, we introduce the notion of *advice* to 1qfa's. Similar to REG/n, the notation 1QFA/n indicates the collection of all languages L over alphabets Σ that satisfy the following condition: there are a 1qfa M, an error bound $\varepsilon \in [0, 1/2)$, and an advice function $h : \mathbb{N} \to \Gamma^*$ with an advice alphabet Γ such that (i) $|h(n)| = n$ for each length $n \in \mathbb{N}$, (ii) for every string $x \in L$, M accepts $[\ {}^{x}_{h(|x|)}\ ]$ with probability at least $1 - \varepsilon$, and (iii) for every $x \in \Sigma^* - L$, M rejects $[\ {}^{x}_{h(|x|)}\ ]$ with probability at least $1 - \varepsilon$. The last two requirements can be succinctly expressed as $\mathrm{Prob}_M[M([\ {}^{x}_{h(|x|)}\ ]) = L(x)] \geq 1 - \varepsilon$. It is important to note that this definition does not require the underlying 1qfa M to have bounded errors on *all* inputs.

An immediate benefit of using advice for 1qfa's is the elimination of endmarkers on their input tapes. Earlier, Brodsky and Pippenger[2] demonstrated that we can eliminate the left endmarker $\mathfrak{c}$. The use of advice further enables us to eliminate the right endmarker \$ as well.

Lemma 3.3 [endmarker lemma] *For any language $L \in 1\mathrm{QFA}/n$, there exist a 1qfa M, a constant $\varepsilon \in [0, 1/2)$, an advice alphabet Γ, and an advice function h such that (i) M's tape has no endmarkers, (ii) $|h(n)| = n$ for any length $n \in \mathbb{N}$, and (iii) for any string $x \in \Sigma^*$, $\mathrm{Prob}_M[M([\ {}^{x}_{h(|x|)}\ ]) = L(x)] \geq 1 - \varepsilon$.*

Similar to the well-known inclusion 1QFA $\subseteq$ REG[6], we can prove the following inclusion, whose proof can be obtained from Lemmas 2.1 and 5.3.

Proposition 3.4 $1\mathrm{QFA}/n \subseteq \mathrm{REG}/n$.

4. Reversible Finite Automata with Advice

Before investigating the roles of advice for 1QFA/n, we shall study a natural

subclass of 1QFA/n—an advised version of 1RFA. Since reversibility is one of the distinguishing features of quantum computation, an analysis of this particular family lays out a useful prelude to that of 1QFA/n in the subsequent section. Formally, a *one-way (deterministic) reversible finite automaton* (or 1rfa, in short) M is a 1dfa $(Q, \Sigma, \delta, q_0, Q_{acc}, Q_{rej})$ such that, for every inner state $q \in Q$ and every symbol $\sigma \in \Sigma$, there exists at most one inner state $q' \in Q$ satisfying $\delta(q', \sigma) = q$. Notice that, unlike 1dfa's, this definition demands *multiple* accepting states and *multiple* rejecting states.

The advised family 1RFA/n consists of all languages L over certain alphabets Σ that satisfy the following condition: there exist a 1rfa M and an advice function h such that (i) $|h(n)| = n$ for any length $n \in \mathbb{N}$ and (ii) $M([\ {}^{x}_{h(|x|)}\ ]) = L(x)$ for every string $x \in \Sigma^*$. Since 1rfa's are a restricted version of 1qfa's, it immediately holds that 1RFA/$n \subseteq$ 1QFA/n.

A machine-independent characterization of Lemma 2.1 turns out to be a useful tool in studying the computational complexity of languages in REG/n. A similar abstract treatment of languages in 1RFA/n is expected to be useful as well. As a key lemma of this section, we present such an abstract characterization.

Lemma 4.1 *Let S be any language over an alphabet Σ. The following two statements are equivalent. Let $\Delta = \{(x, n) \mid x \in \Sigma^*, n \in \mathbb{N}, |x| \leq n\}$. (1) S is in* 1RFA/n. *(2) There is an equivalence relation $\equiv_S$ over Δ such that (i) the set $\Delta/\equiv_S$ is finite, and (ii) for any length parameter $n \in \mathbb{N}$, any symbol $\sigma \in \Sigma$, and any two strings $x, y \in \Sigma^*$ with $|x| = |y| \leq n$, the following holds: (a) whenever $|x\sigma| \leq n$, $(x\sigma, n) \equiv_S (y\sigma, n)$ iff $(x, n) \equiv_S (y, n)$, and (b) if $(x, n) \equiv_S (y, n)$, then $S(xz) = S(yz)$ for all strings z with $|xz| = n$.*

Condition (a) in this lemma concerns the reversibility of 1rfa's. The proof of Lemma 4.1 is similar in nature to that[13] of Lemma 2.1 except for a treatment of the reversibility.

To see how useful this lemma is, we shall show that 1QFA is not included in 1RFA/n. This result can be viewed as a strength of bounded-error quantum computation over error-free one.

Theorem 4.2 1QFA $\not\subseteq$ 1RFA/n. *Thus,* 1RFA/$n \neq$ 1QFA/n.

5. Limitation of Advice for Quantum Computation

In Section 3, deterministic advice is used as a resource that helps its underlying 1qfa's gain more language-recognition power. There are also clear limitations on the use of such advice. One such limitation is that 1QFA/n is not large enough to contain REG.

To deal with 1QFA/n, we note that some of the well-known properties proven for 1QFA are not as useful as we hope them to be. The first of such properties is a criterion, known as a *partial order condition*[⋆1] of Brodsky and Pippenger[2], which every language in 1QFA must satisfy. As shown below, 1QFA/n unfortunately violates this criterion, making an analysis of languages in 1QFA/n quite different.

Example 5.1 1QFA/n does not satisfy the partial order condition criterion.

Kondacs and Watrous[6] first proved that REG $\not\subseteq$ 1QFA. We want to strengthen their result by proving a class separation between REG and 1QFA/n.

Theorem 5.2 REG $\not\subseteq$ 1QFA/n. *Thus,* 1QFA/$n \neq$ REG/n.

An argument of Kondacs and Watrous[6] for REG $\not\subseteq$ 1QFA used the separation language L_a, defined in Example 5.1. As pointed out by Brodsky and Pippenger[2], this result follows from the fact that L_a does not satisfy the aforementioned partial order condition. Therefore, by Example 5.1, their proof technique is not sufficient to prove the desired separation of REG $\not\subseteq$ 1QFA/n. There is another argument employed by Ambainis and Freivalds[1]. In their analysis of the computational behaviors of a 1qfa, Ambainis and Freivalds utilized a maximal subspace, closed under a T_σ operator. However, the presence of advice makes it difficult to employ a similar technique, since it requires an input size to change. We need to seek another way to prove the desired separation between REG and 1QFA/n. For the proof of Theorem 5.2, we shall give a key lemma, which describes a certain unique characteristic of languages computed by bounded-error

⋆1 We say that a language satisfies the partial order condition if its minimal 1dfa contains no two inner states $q_1, q_2 \in Q$ such that (i) there is a string z for which $\delta(q_1, z) \in Q_{acc}$ and $\delta(q_2, z) \notin Q_{acc}$ or vice versa, and (2) there are two nonempty strings x, y for which $\delta(q_1, x) = \delta(q_2, x) = q_2$ and $\delta(q_2, y) = q_1$.

1qfa's with advice.

We begin with a description of our key lemma. Following a standard convention, for any partial order $\leq$ defined on a finite set, we use the notation $x = y$ whenever $x \leq y$ and $y \leq x$; moreover, we write $x < y$ if $x \leq y$ and $x \neq y$. A finite sequence $(s_1, s_2, \ldots, s_m)$ is called a *strictly descending chain* of length m (with respect to $\leq$) if $s_{i+1} < s_i$ for any index $i \in [m-1]$.

For our convenience, we call a reflexive, symmetric, binary relation a *closeness relation*. For any closeness relation $\cong$, an $\cong$*-discrepancy set* S is a set such that, for any two elements $x, y \in S$, if x and y are different, then $x \ncong y$.

Lemma 5.3 [key lemma] *Let S be any language over an alphabet Σ. Let $\Delta = \{(x,n) \in \Sigma^* \times \mathbb{N} \mid |x| \leq n\}$. If $S \in$ 1QFA$/n$, then there exist two constants $c, d > 0$, an equivalence relation $\equiv_S$ over Δ, a partial order $\leq_S$ over Δ, and a closeness relation $\cong$ over Δ that satisfy the following seven conditions. Let $(x,n), (y,n) \in \Delta$, $z \in \Sigma^*$, and $\sigma \in \Sigma$ with $|x| = |y|$. (1) The set $\Delta/\equiv_S$ is finite. (2) If $(x,n) \cong (y,n)$, then $(x,n) \equiv_S (y,n)$. (3) If $|x\sigma| \leq n$, then $(x\sigma, n) \leq_S (x,n)$. (4) If $|xz| \leq n$, $(x,n) =_S (xz,n)$, $(y,n) =_S (yz,n)$, and $(xz,n) \cong (yz,n)$, then $(x,n) \equiv_S (y,n)$. (5) $(x,n) \equiv_S (y,n)$ iff $S(xz) = S(yz)$ for all $z \in \Sigma^*$ with $|xz| = n$. (6) Any strictly descending chain (with respect to $\leq_S$) in Δ has length at most c. (7) Any $\cong$-discrepancy subset of Δ has cardinality at most d.*

Theorem 5.2 is a direct consequence of Lemma 5.3. The lemma instantly guides us to Proposition 3.4 with help of Lemma 2.1. Our proof of Lemma 5.3 heavily depends on Lemma 3.2.

6. Randomized Advice and Automata

Unlike deterministic advice, *randomized advice* has been proven to endow an enormous power to one-way finite automata[13], where randomized advice refers to a *probability ensemble* $\{D_n\}_{n\in\mathbb{N}}$ consisting of an infinite series of probability distributions D_n over the advice strings Γ^n. For our notational simplicity, we use the same notation D_n for a random variable expressing strings $y \in \Gamma^n$ occurring with probability $D_n(y)$. Another notation $\left[\begin{smallmatrix} x \\ D_n \end{smallmatrix}\right]$ also denotes a random variable expressing a string $\left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$, provided that y ($\in \Gamma^n$) is chosen with probability $D_n(y)$. Yamakami[13] introduced advised language families REG$/Rn$, CFL$/Rn$, 1-BPLIN$/Rlin$, and 1-PLIN$/Rlin$ using randomized advice instead of deterministic advice. Analogous to the notations REG$/Rn$ and CFL$/Rn$ of Yamakami[13], 1QFA$/Rn$ indicates the collection of all languages L that satisfy the following condition: there are a 1qfa M, a constant $\varepsilon \in [0, 1/2)$, an advice alphabet Γ, and an advice probability ensemble $\{D_n\}_{n\in\mathbb{N}}$ ($D_n : \Gamma^n \to [0,1]$) such that, for every string $x \in \Sigma^*$, $\mathrm{Prob}_{M, D_{|x|}}[M(\left[\begin{smallmatrix} x \\ D_{|x|} \end{smallmatrix}\right]) = L(x)] \geq 1 - \varepsilon$.

We begin with a simple observation of how powerful random advice can be. Recall that ALL denotes the collection of all languages. By modifying the proof of the collapse result 1-PLIN$/Rlin$ = ALL[13], we can prove the following statement.

Lemma 6.1 1QFA$_{(1/2,1/2)}/Rn$ = ALL.

In Proposition 3.4, for deterministic advice, we have shown an inclusion of 1QFA$/n$ inside REG$/n$. When randomized advice is concerned, a similar inclusion still holds between 1QFA$/Rn$ and REG$/Rn$; however, its proof requires an analysis of success probability. Note that randomized advice does not automatically commute the inclusions between langauge families.

Proposition 6.2 1QFA$/Rn \subseteq$ REG$/Rn$.

For comparison, let us introduce another advised family 1RFA$/Rn$ using the 1rfa model instead of the 1qfa model. More precisely, 1RFA$/Rn$ is the collection of all languages L satisfying the following condition: there exist a 1rfa M, an error bound $\varepsilon \in [0, 1/2)$, and a probability ensemble $\{D_n\}_{n\in\mathbb{N}}$ such that, for every string $x \in \Sigma^*$, $\mathrm{Prob}_{M, D_{|x|}}[M(\left[\begin{smallmatrix} x \\ D_{|x|} \end{smallmatrix}\right]) = L(x)] \geq 1 - \varepsilon$. It is obvious that 1RFA$/Rn \subseteq$ 1QFA$/Rn$.

We shall demonstrate that a use of randomized advice increases the language-recognition power of 1qfa's as well as 1rfa's.

Proposition 6.3 1QFA$/n \neq$ 1QFA$/Rn$ *and* 1RFA$/n \neq$ 1RFA$/Rn$.

The above proposition immediately follows from the next lemma, in which we claim a class separation between REG$/n$ and DCFL $\cap$ 1RFA$/Rn$.

Lemma 6.4 $\mathrm{DCFL} \cap \mathrm{1RFA}/Rn \not\subseteq \mathrm{REG}/n$.

7. Power of Quantum Advice

Beyond randomized advice, we shall discuss another type of advice, known as *quantum advice*. Through this section, we shall argue how to define and use such quantum advice on our model of one-way quantum finite automata.

7.1 Quantum Advice on Read-Only Tape Tracks

In the past literature, quantum advice has been discussed mostly in the context of polynomial-time computations (see, e.g., Nishimura and Yamakami[9]) as a series of (pure) quantum states that help quantum Turing machines. Associated with an advice alphabet Γ, let $|\phi_n\rangle$ denote a normalized quantum state in a Hilbert space of dimension $|\Gamma|^n$. Using a computational basis Γ^n, we can assume $|\phi_n\rangle$ to be a superposition of the form $\sum_{s\in\Gamma^n} \alpha_s|s\rangle$ with $\alpha_s \in \mathbb{C}$ such that $\sum_{s\in\Gamma^n} |\alpha_s|^2 = 1$. For our later convenience, the succinct notation $|[\begin{smallmatrix} x \\ \phi_n \end{smallmatrix}]\rangle$ indicates the quantum state $\sum_{s\in\Gamma^n} \alpha_s|t_{x,s}\rangle$, where $t_{x,s} = [\begin{smallmatrix} x \\ s \end{smallmatrix}]$, in computational basis $\{[\begin{smallmatrix} x \\ s \end{smallmatrix}] \mid s \in \Gamma^n\}$.

Unlike quantum Turing machines, our current model of 1qfa's with read-only tape tracks severely limits the potential power of quantum advice. Observe that, since advice strings in a quantum advice state are unaltered, quantum computations associated with different advice strings never interfere with one another. This observation leads to the following lemma.

Lemma 7.1 *Let A be any language over an alphabet Σ. The following two statements are equivalent. (1) $A \in \mathrm{1QFA}/Rn$. (2) There exist a 1qfa M with read-only input tape tracks, an advice alphabet Γ, a series $\Phi = \{|\phi_n\rangle\}_{n\in\mathbb{N}}$ of quantum advice over Γ^*, and a constant $\varepsilon \in [0, 1/2)$ satisfying that $\mathrm{Prob}_M[M([\begin{smallmatrix} x \\ \phi_{|x|} \end{smallmatrix}]) = A(x)] \geq 1 - \varepsilon$ for any input $x \in \Sigma^*$.*

Lemma 7.1 says that, if a 1qfa has only read-only tape tracks, then the usage of quantum advice is reduced to that of randomized advice. The lemma therefore leads us to an introduction of the following notion of "rewritable" tape tracks. Notice that a simple and natural extension of our 1qfa model is to allow a machine's tape head to modify advice strings. When we deal with certain types of classical finite automata, it is of no importance whether the heads can erase or even rewrite all symbols of given advice strings after scanning them. This is because one-tape two-way Turing machines that modify tape contents in linear time can be simulated by one-way finite automata[10),13)] (see Fig. 1). Quantum computation, on the contrary, draws a benefit from a modification of advice strings, although a one-way head move still hampers the machine's ability. Here, we modify our original 1qfa's so that they can access rewritable tape tracks and modify track contents at the time when their tape heads scan tape cells. For our convenience, we call such a modified machine a *rewritable 1qfa*.

To be more precise, for each symbol $\sigma \in \Sigma$, let U_σ be any unitary transform acting on the space $E_{Q,\Gamma} = span\{|q\rangle|\tau\rangle \mid q \in Q, \tau \in \Gamma\}$, instead of $span\{|q\rangle \mid q \in Q\}$ used in Sections 3-6. Similarly, three projection operators P_{acc}, P_{rej}, and P_{non} can be also modified using $E_{Q,\Gamma}$. For each fixed index $i \in [n]$, $U_\sigma^{(i)}$ acts on $E_n = span\{|q\rangle|y\rangle \mid q \in Q, y \in \Gamma^n\}$ and, by applying U_σ, it modifies only the content of the ith tape cell as well as M's inner state. We define $T_\sigma^{(i)} = P_{non}U_\sigma^{(i)}$. Let $x = x_1x_2\cdots x_n$ be any string in $\check{\Sigma}^*$, and an extended operator $T_x = T_{x_n}^{(n)} \cdots T_{x_2}^{(2)}T_{x_1}^{(1)}$ acts on E_n. A rewritable 1qfa on the input x starts with the (initial) quantum state $|q_0\rangle|\phi_n\rangle$, where $|\phi_n\rangle$ is a quantum advice state in $span\{|z\rangle \mid z \in \Gamma^n\}$. The acceptance probability $p_{acc}(x)$ of M on x is the sum, over all $i \in [n]$, of $||(P_{acc}U_{x_i}^{(i)})T_{x_{i-1}}^{(i-1)} \cdots T_{x_2}^{(2)}T_{x_1}^{(1)}|q_0\rangle|\phi_n\rangle||^2$. The rejection probability $p_{rej}(x)$ of M on x is similarly defined.

The use of rewritable tape tracks makes it possible to reduce the number of applications of (projection) measurement. In short, measure-many quantum finite automata are "equivalent" in essence to measure-once ones in our setting.

Lemma 7.2 *For any rewritable 1qfa M with quantum advice $\{|\phi_n\rangle\}_{n\in\mathbb{N}}$, there exist another rewritable 1qfa N and another quantum advice $\{|\phi'_n\rangle\}_{n\in\mathbb{N}}$ such that (i) N applies a measurement only once just after scanning an entire input and (ii) N's acceptance probability equals that of M on any input x.*

7.2 Rewritable Quantum Finite Automata

As shown in Lemma 7.2, a use of rewritable tape tracks simplifies the behavioral

descriptions of 1qfa's by reducing the number of measurements. Furthermore, we shall show that rewritable tape tracks make 1qfa's easier to handle.

To emphasize our use of "quantum" advice, we use a special notation 1QFA/Qn to designate the family of all languages recognized using quantum advice by rewritable 1qfa's with bounded-error probability. The error bounds given here can be relaxed as follows. This relaxation is useful in constructing desired 1qfa's for given target languages.

Lemma 7.3 *Let $L \subseteq \Sigma^*$. Assume that a rewritable 1qfa M satisfies the following condition: there exist two constants ε_0 and ε_1 and a series $\{|\phi_n\rangle\}_{n\in\mathbb{N}}$ of quantum advice states such that (i) $0 \leq \varepsilon_0 < \varepsilon_1 \leq 1$, (ii) for any $x \in L$, M accepts $\left[\begin{smallmatrix} x \\ \phi_n \end{smallmatrix}\right]$ with probability at least ε_1, and (iii) for any $x \in L$, M accepts $\left[\begin{smallmatrix} x \\ \phi_n \end{smallmatrix}\right]$ with probability at most ε_0. Then, $L \in$ 1QFA/Qn.*

An immediate consequence of Lemma 7.2 is a closure property of 1QFA/Qn under Boolean operators. In contrast, this property is not yet known to hold for 1QFA.

Proposition 7.4 *The language family* 1QFA/Qn *is closed under complementation, union, and intersection.*

The power of rewritable 1qfa's with quantum advice is exemplified in the next proposition. The language family 1-BQLIN is the collection of all languages recognized by one-tape two-way one-head off-line quantum Turing machines, where all the (classically-viewed) computation paths of the machines must terminate within linearly many steps[10]. Using linear-size quantum advice, similar to 1-PLIN/$Rlin$, we can expand 1-BQLIN to its advised version 1-BQLIN/$Qlin$.

Proposition 7.5 REG/$Rn \subseteq$ 1QFA/$Qn \subseteq$ 1-BQLIN/$Qlin$.

References

1) A. Ambainis and R. Freivalds. 1-way quantum finite automata: strengths, weaknesses, and generalizations. In *Proc. of the 39th Annual Symposium on Foundations of Computer Science*, pp.332–342, 1998.
2) A. Brodsky and N. Pippenger. Characterizations of 1-way quantum finite automata. *SIAM J. Comput.* 31 (2002) 1456–1478.
3) C. Damm and M. Holzer. Automata that take advice. In *Proc. 20th Symposium on Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, Vol.969, pp.149–152, Springer, 1995.
4) J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. (Second Edition) Addison Wesley, 2001.
5) R. M. Karp and R. Lipton. Turing machines that take advice. *LEnseignement Mathématique*. 2nd series, Vol.28 (1982) 191–209.
6) A. Kondacs and J. Watrous. On the power of quantum finite state automata. In *Proc. 38th Annual Symposium on Foundations of Computer Science*, pp.66–75, 1997.
7) C. Moore and J. Crutchfield. Quantum automata and quantum languages. *Theoretical Computer Science* 237 (2000) 275–306.
8) H. Nishimura and T. Yamakami. An application of quantum finite automata to interactive proof systems. *Journal of Computer and System Sciences*, 75 (2009) 255-269. An extended abstract appeared in the *Proceedings of the 9th International Conference on Implementation and Application of Automata*, Lecture Notes in Computer Science, Vol.3317, pp.225–236, Springer, 2004.
9) H. Nishimura and T. Yamakami. Polynomial-time quantum computation with advice. *Inf. Process. Lett.* 90 (2004) 195–204.
10) K. Tadaki, T. Yamakami, and J. Lin. Theory of one tape linear time Turing machines. *Theoretical Computer Science*, 411 (2010) 22–43. A preliminary version appeared in *Proc. 30th SOFSEM Conference on Current Trends in Theory and Practice of Computer Science*, Lecture Notes in Computer Science, Vol.2932, pp.335–348, Springer, 2004. See also arXiv:cs/0310046.
11) T. Yamakami. Swapping lemmas for regular and context-free languages. Manuscript, 2008. See arXiv:0808.4122.
12) T. Yamakami. Immunity and pseudorandomness of context-free languages. Manuscript, 2009. See arXiv:0902.0261.
13) T. Yamakami. The roles of advice to one-tape linear-time Turing machines and finite automata. In *Proc. 20th International Symposium on Algorithms and Computation*, Lecture Notes in Computer Science, Vol.5878, pp.933–942, Springer, 2009.
14) A. C. Yao. Probabilistic complexity: Towards a unified measure of complexity. In *Proc. of the 18th IEEE Annual Symp. on Foundation of Computer Science*, pp.222–227, 1977.