State succinctness of two-way finite automata with quantum and classical states

S. Zheng[1,3] D. Qiu[1,2,4] J. Gruska[3] L. Li[1] P. Mateus[2]

[1]Department of Computer Science, Sun Yat-sen University, Guangzhou 510006, China

[2]SQIG–Instituto de Telecomunicações, Departamento de Matemática,
Instituto Superior Técnico, Universidade de Lisboa,
Av. Rovisco Pais 1049-001, Lisbon, Portugal

[3]Faculty of Informatics, Masaryk University, Brno, Czech Republik

[4]The State Key Laboratory of Computer Science, Institute of Software,
Chinese Academy of Sciences, Beijing 100080, China

Abstract

Two-way quantum automata with quantum and classical states (2QCFA) were introduced by Ambainis and Watrous in 2002. In this paper we study state succinctness of 2QCFA. For any $m \in \mathbb{Z}^+$ and any $\epsilon < 1/2$, we show that:

1. there is a promise problem $A^{eq}(m)$ which can be solved by a 2QCFA with one-sided error ϵ in a polynomial expected running time with a constant number (that depends neither on m nor on ε) of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states, whereas the sizes of the corresponding *deterministic finite automata* (DFA), *two-way nondeterministic finite automata* (2NFA) and polynomial expected running time *two-way probabilistic finite automata* (2PFA) are at least $2m+2$, $\sqrt{\log m}$, and $\sqrt[3]{(\log m)/b}$, respectively;
2. there exists a language $L^{twin}(m) = \{wcw | w \in \{a,b\}^*\}$ over the alphabet $\Sigma = \{a,b,c\}$ which can be recognized by a 2QCFA with one-sided error ϵ in an exponential expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states, whereas the sizes of the corresponding DFA, 2NFA and polynomial expected running time 2PFA are at least 2^m, $\sqrt{m}$, and $\sqrt[3]{m/b}$, respectively;

where b is a constant.

Keywords: Computing models, Quantum finite automata, State complexity, Succinctness

1 Introduction

An important way to get a deeper insight into the power of various quantum resources and features for information processing is to explore the power of various quantum variations of the basic models of classical automata. Of a special interest and importance is to do that for various quantum variations of classical finite automata because quantum resources are not cheap and quantum operations are not easy to implement. Attempts to find out how much one can do with very little of quantum resources and consequently with the simplest quantum variations of classical finite automata are therefore of a particular interest. This paper is an attempt to contribute to such line of research.

There are two basic approaches how to introduce quantum features to classical models of finite automata. The first one is to consider quantum variants of the classical *one-way (deterministic) finite automata* (1FA or 1DFA) and the second one is to consider quantum variants of the classical *two-way finite automata* (2FA or 2DFA). Already the very first attempts to introduce such models, by Moore and Crutchfields [23] and Kondacs and Watrous [16] demonstrated that in spite of the fact that in the classical case, 1FA and 2FA have the same recognition power, this is not so for their quantum variations (in case only unitary operations and projective measurements are considered as quantum operations). Moreover, already the first important model of *two-way quantum finite automata* (2QFA), namely that introduced by Kondacs and Watrous, demonstrated that very natural quantum variants of 2FA are much too powerful - they can recognize even some *non-context free languages* and are actually not really finite in a strong sense [16]. It started to be therefore of interest to introduce and explore some "less quantum" variations of 2FA and their power [1, 2, 3, 9, 7, 20, 21, 22, 24, 26, 30, 31, 38, 39, 40].

A very natural "hybrid" quantum variations of 2FA, namely, *two-way quantum automata with quantum and classical states* (2QCFA) were introduced by Ambainis and Watrous [3]. Using this model they were able to show, in an elegant way, that an addition of a single qubit to a classical model can enormously increase the power of automata. A 2QCFA is essentially a classical 2FA augmented with a quantum memory of constant size (for states in a fixed Hilbert space) that does not depend on the size of the (classical) input. In spite of such a restriction, 2QCFA have been shown to be more powerful than *two-way probabilistic finite automata* (2PFA) [3].

State complexity and succinctness results are an important research area of the classical automata theory, see [41], with a variety of applications. Once quantum versions of classical automata were introduced and explored, it started to be of large interest to find out, also through succinctness results, a relation between the power of classical and quantum automata models. This has turned out to be an area of surprising outcomes that again indicated that the relations between the classical and the corresponding quantum automata models are intriguing. For example, it has been shown, see [2, 4, 5, 6, 19], that for some languages 1QFA require exponentially less states than classical 1FA, but for some other languages it can be in an opposite way.

Because of the simplicity, elegance and interesting properties of the 2QCFA model, as well as its natural character, it seems to be both useful and interesting to explore the state complexity and succinctness results of 2QCFA and this we will do in this paper.

In the first part of this paper, 2QCFA are recalled formally and some basic notations are also given. Then we will prove a state succinctness result of 2QCFA on an infinite family of promise problems. For any $m \in \mathbb{Z}^+$ let $A^{eq}_{yes}(m) = \{w \in \{a,b\}^* | w = a^m b^m\}$ and $A^{eq}_{no}(m) = \{w \in \{a,b\}^* | w \neq a^m b^m \ and \ |w| \geq m\}$. For any $\epsilon < 1/2$ (ϵ is always a nonnegative number in this paper), we will prove that the promise problem $A^{eq}(m) = (A^{eq}_{yes}(m), A^{eq}_{no}(m))$ can be solved by a 2QCFA with one-sided error ϵ in a polynomial expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ (the base of logarithm is always 2 in this paper) classical states, whereas sizes of the corresponding DFA, 2DFA and 2NFA are at least $2m+2$, $\sqrt{\log m}$ and $\sqrt{\log m}$, respectively. We also show that for any $m \in \mathbb{Z}^+$, any 2PFA that solves the promise problem $A^{eq}(m)$ with an error probability $\epsilon < 1/2$ and within polynomial expected running time has least $\sqrt[3]{(\log m)/b}$ states, where $b > 0$ is a constant. Finally, we show a state succinctness result of 2QCFA on an infinite family of languages. For any $m \in \mathbb{Z}^+$ and any $\epsilon < 1/2$, there exists a 2QCFA that recognizes the language $L^{twin}(m) = \{wcw | w \in \{a,b\}^*\}$ over the alphabet $\Sigma = \{a,b,c\}$ with one-sided error ϵ in an exponential expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states. We use a lower bound of the *communication complexity* to prove that any DFA recognizing the language $L^{twin}(m)$ has at least 2^m states. Next, we prove that the sizes of the corresponding 2DFA and 2NFA to recognize $L^{twin}(m)$ are at least $\sqrt{m}$. We also show that for any $m \in \mathbb{Z}^+$, any 2PFA recognizing $L^{twin}(m)$ with an error probability $\epsilon < 1/2$ and within polynomial expected running time has least $\sqrt[3]{m/b}$ states, where $b > 0$ is a constant.

We now outline the remainder of this paper. Definition of 2QCFA and some auxiliary lemmas are recalled in Section 2. In Section 3 we prove a state succinctness result for 2QCFA on an infinite family of promise problems. Then we show a state succinctness result of 2QCFA on an infinite family of languages in Section 4. Finally, Section 5 contains a conclusion and some open problems.

2 Preliminaries

In the first part of this section we formally recall the model of 2QCFA we will use. Concerning the basics of *quantum computation* we refer the reader to [13, 25], and concerning the basic properties of automata models, we refer the reader to [13, 14, 15, 27, 29, 32].

2.1 2QCFA

2QCFA were first introduced by Ambainis and Watrous [3], and then studied by Qiu, Yakaryilmaz and etc. [28, 38, 42, 43]. Informally, we describe a 2QCFA as a 2DFA which has an access to a quantum memory of a constant size (dimension), upon which it can perform quantum unitary transformations or projective measurements. Given a finite set of quantum states Q, we denote by $\mathcal{H}(Q)$ the Hilbert space spanned by Q. Let $\mathcal{U}(\mathcal{H}(Q))$ and $\mathcal{O}(\mathcal{H}(Q))$ denote sets of the unitary operators and projective measurements over $\mathcal{H}(Q)$, respectively.

Definition 1. A 2QCFA $\mathcal{A}$ is specified by a 9-tuple

$$\mathcal{A} = (Q, S, \Sigma, \Theta, \delta, q_0, s_0, S_{acc}, S_{rej}) \tag{1}$$

where:

1. Q is a finite set of quantum states;
2. S is a finite set of classical states;
3. Σ is a finite set of input symbols; Σ is then extended to the tape symbols set $\Gamma = \Sigma \cup \{ ¢, \$\}$, where $¢ \notin \Sigma$ is called the left end-marker and $\$ \notin \Sigma$ is called the right end-marker;
4. $q_0 \in Q$ is the initial quantum state;
5. $s_0 \in S$ is the initial classical state;
6. $S_{acc} \subset S$ and $S_{rej} \subset S$ satisfying $S_{acc} \cap S_{rej} = \emptyset$ are the sets of classical accepting and rejecting states, respectively.
7. Θ is the transition function of quantum states:

   $$\Theta : S \setminus (S_{acc} \cup S_{rej}) \times \Gamma \to \mathcal{U}(\mathcal{H}(Q)) \cup \mathcal{O}(\mathcal{H}(Q)). \tag{2}$$

   Thus, $\Theta(s, \gamma)$ is either a unitary transformation or a projective measurement.
8. δ is the following transition function of classical states.

   a) If $\Theta(s, \gamma) \in \mathcal{U}(\mathcal{H}(Q))$, then

   $$\delta : S \setminus (S_{acc} \cup S_{rej}) \times \Gamma \to S \times \{-1, 0, 1\}, \tag{3}$$

   which is similar to the transition function for 2DFA. $\delta(s, \gamma) = (s', d)$ means that when the classical state $s \in S$ is scanning $\gamma \in \Gamma$ then the state is changed to a state s', and the movement of the tape head is determined by d (moving right one cell if $d = 1$, left if $d = -1$, and being stationary if $d = 0$).

   b) If $\Theta(s, \gamma) \in \mathcal{O}(\mathcal{H}(Q))$, then $\Theta(s, \gamma)$ is a projective measurement with a set of possible eigenvalues $R = \{r_1, \cdots, r_n\}$ and the projectors set $\{P(r_i) : i = 1, \cdots, n\}$, where $P(r_i)$ denotes the projector onto the eigenspace corresponding to the eigenvalue r_i. In such a case

   $$\delta : S \setminus (S_{acc} \cup S_{rej}) \times \Gamma \times R \to S \times \{-1, 0, 1\}, \tag{4}$$

where $\delta(s,\gamma)(r_i) = (s',d)$ means that when the projective measurement result is r_i, then the classical state $s \in S$ is changed to s', and the movement of the tape head is determined by d.

Given an input w, a 2QCFA $\mathcal{A} = (Q, S, \Sigma, \Theta, \delta, q_0, s_0, S_{acc}, S_{rej})$ proceeds as follows: at the beginning, the tape head is positioned on the left end-marker ¢, the quantum initial state is $|q_0\rangle$, the classical initial state is s_0. In the next steps, if the current quantum state is $|\psi\rangle$, the current classical state is $s \in S \setminus (S_{acc} \cup S_{rej})$ and the current scanning symbol is $\sigma \in \Gamma$, then the quantum state $|\psi\rangle$ and the classical state s will be changed according to $\Theta(s,\sigma)$ as follows:

1. if $\Theta(s,\sigma)$ is a unitary operator U, then U is applied to the current quantum state $|\psi\rangle$ changing it into $U|\psi\rangle$, and $\delta(s,\sigma) = (s',d) \in S \times \{-1,0,1\}$ makes the current classical state s to become s', together with the tape head moving in terms of d. In case $s' \in S_{acc}$, the input is accepted, and in case $s' \in Q_{rej}$, the input rejected;
2. if $\Theta(s,\sigma)$ is a projective measurement, then the current quantum state $|\psi\rangle$ is changed to the quantum state $P_j|\psi\rangle / \|P_j|\psi\rangle\|$ with probability $\|P_j|\psi\rangle\|^2$ and in this case, $\delta(s,\sigma)$ is a mapping from the set of all possible results of the measurement to $S \times \{-1,0,1\}$. For instance, for the result r_j of the measurement, and $\delta(s,\sigma)(r_j) = (s_j, d)$, we have
   (a) if $s_j \in S \setminus (S_{acc} \cup S_{rej})$, new classical state is s_j and the head moves in the direction d;
   (b) if $s_j \in S_{acc}$, then the machine accepts the input and the computation halts;
   (c) and similarly, if $s_j \in S_{rej}$, then the machine rejects the input and the computation halts.

The computation will end if the resulting classical state is in $S_{acc} \cup S_{rej}$. Therefore, similarly to the definition of accepting and rejecting probabilities for 2QFA [16], the accepting and rejecting probabilities $Pr[\mathcal{A}\ \textit{accepts}\ w]$ and $Pr[\mathcal{A}\ \textit{rejects}\ w]$ for an input w are, respectively, the sums of all accepting probabilities and all rejecting probabilities before the end of the machine for computing input w.

Let $L \subset \Sigma^*$ and $\epsilon < 1/2$. A 2QCFA $\mathcal{A}$ recognizes L with one-sided error ϵ if

1. $\forall w \in L$, $Pr[\mathcal{A}\ \textit{accepts}\ w] = 1$, and
2. $\forall w \notin L$, $Pr[\mathcal{A}\ \textit{rejects}\ w] \geq 1 - \epsilon$.

2.2 Notations and auxiliary lemmas

In this subsection we review some additional notations related to 2QCFA [28]. For convenience, let $2QCFA_\epsilon$ denote the classes of all languages recognized by 2QCFA with a given error probability ϵ and $2QCFA_\epsilon(ptime)$ denote the classes of languages recognized in polynomial expected time by 2QCFA with a given error probability ϵ. Moreover, let $QS(\mathcal{A})$ and $CS(\mathcal{A})$ denote the numbers of quantum states and classical states of a 2QCFA $\mathcal{A}$ and let $T(\mathcal{A})$ denote the expected running time of 2QCFA $\mathcal{A}$. For a string w, the length of w is denoted by $|w|$.

Lemma 1 ([3])**.** *For any $\epsilon < 1/2$, there is a 2QCFA $\mathcal{A}(\epsilon)$ that accepts any $w \in L^{eq} = \{a^m b^m | m \in \mathbb{N}\}$ with certainty, rejects any $w \notin L^{eq}$ with probability at least $1-\epsilon$ and halts in the expected running time $\mathbf{O}(|w|^4)$, where w is the input.*

Remark 1. According to the proof of Lemma 1 in [3], for the above 2QCFA $\mathcal{A}(\epsilon)$, we further have $QS(\mathcal{A}(\epsilon)) = 2$, $CS(\mathcal{A}(\epsilon)) \in \mathbf{O}(\log \frac{1}{\epsilon})$.

Lemma 2 ([28])**.** *If $L_1 \in 2QCFA_{\epsilon_1}(2QCFA_{\epsilon_1}(ptime))$ and $L_2 \in 2QCFA_{\epsilon_2}(2QCFA_{\epsilon_2}(ptime))$, then $L_1 \cap L_2 \in 2QCFA_\epsilon(2QCFA_\epsilon(ptime))$, where $\epsilon = \epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$.*

Remark 2. According to the proof of Lemma 2 in [28], if a 2QCFA $\mathcal{A}_1$ recognizes L_1 with one-sided error ϵ_1 (in a polynomial expected time) and if a 2QCFA $\mathcal{A}_2$ recognizes L_2 with one-sided error ϵ_2 (in a polynomial expected time), then there is a 2QCFA $\mathcal{A}$ recognizes $L_1 \cap L_2$ (in polynomial expected time), where $QS(\mathcal{A}) = QS(\mathcal{A}_1) + QS(\mathcal{A}_2)$ and $CS(\mathcal{A}) = CS(\mathcal{A}_1) + CS(\mathcal{A}_2) + QS(\mathcal{A}_1)$.

Lemma 3 ([33, 34])**.** *Every n-state 2DFA can be simulated by a DFA with $(n+1)^{n+1}$ states.*

Lemma 4 ([8])**.** *Every n-state 2NFA can be simulated by a DFA with $2^{(n-1)^2+n}$ states.*

Definition 2. Let a language $L \subset \Sigma^*$ and $\epsilon < 1/2$, then a 2PFA $\mathcal{A}$ recognizes L with error probability ϵ if

(1) $\forall w \in L$, $Pr[\mathcal{A}\ accepts\ w] \geq 1-\epsilon$, and

(2) $\forall w \notin L$, $Pr[\mathcal{A}\ rejects\ w] \geq 1-\epsilon$.

A 2PFA $\mathcal{A}$ recognizes L if there is an $\epsilon < 1/2$ such that $\mathcal{A}$ recognizes L with the error probability ϵ.

Definition 3. Let $A, B \subseteq \Sigma^*$ with $A \cap B = \varnothing$, then a 2PFA $\mathcal{A}$ separates A and B [11] if there is some $\epsilon < 1/2$ such that

(1) $\forall w \in A$, $Pr[\mathcal{A}\ accepts\ w] \geq 1-\epsilon$, and

(2) $\forall w \in B$, $Pr[\mathcal{A}\ rejects\ w] \geq 1-\epsilon$.

Lemma 5 ([10])**.** *For every $\epsilon < 1/2$, $a > 0$ and $d > 0$, there exists a constant $b > 0$ such that, for any c, if a language L is recognized by a c-state 2PFA with an error probability ϵ and within time an^d, then L is recognized by some DFA with at most c^{bc^2} states, where $n = |w|$ is the length of the input.*

Lemma 6 ([11]). *Let $A, B \subseteq \Sigma^*$ with $A \cap B = \varnothing$. Suppose there is an infinite set I of positive integers and, for each $m \in I$, a set $W_m \subseteq \Sigma^*$ such that*

(1) $|w| \leq m$ for all $w \in W_m$,

(2) for every integer k, there is an m_k such that $|W_m| \geq m^k$ for all $m \in I$ with $m \geq m_k$, and

(3) for every $m \in I$ and every $w, w' \in W_m$ with $w \neq w'$, there are words $u, v \in \Sigma^*$ such that either $uwv \in A$ and $uw'v \in B$ or $uwv \in B$ and $uw'v \in A$.

Then no 2PFA separates A and B.

We recall some basic notations of *communication complexity*, and we refer the reader to [17, 18, 37] for more details. We will deal with the situation that there are only two communicating parties and with very simple tasks of computing two argument functions where one argument is known to one party and the other argument is known to the other party. We will completely ignore computational resources needed by the parties and we focuses solely on the amount of communication exchanged between the parties.

Let X, Y, Z be arbitrary finite sets. We consider a two-argument function $f : X \times Y \to Z$ and two communicating parties. Alice is given an input $x \in X$ and Bob is given an input $y \in Y$. They wish to compute $f(x, y)$.

The computation of the value $f(x, y)$ is done using a communication protocol. During the execution of the protocol, the two parties alternate roles in sending messages. Each of these messages is a string of bits. The protocol, based on the communication so far, specifies for each step whether the communication terminated (in which case it also specifies what is the output). If the communication has not terminated, the protocol specifies what message the sender (Alice or Bob) should send next, as a function of its input and of the communication so far. A communication protocol $\mathcal{P}$ computes the function f, if for every input pair $(x, y) \in A \times B$ the protocol terminates with the value $f(x, y)$ as its output.

We define the deterministic communication complexity of $\mathcal{P}$ as the worst case number of bits exchanged by the protocol. The deterministic communication complexity of a function f is the communication complexity of the best protocol that computes f, denoted by $D(f)$.

Lemma 7 ([17]). *If Alice and Bob each holds an n length string, $x, y \in \{a, b\}^n$ and the equality function, $EQ_n(x, y)$, is defined to be 1 if $x = y$ and 0 otherwise, then*

$$D(EQ_n) = n. \tag{5}$$

Remark 3. A trivial protocol for the equality function is that Alice sends her input x to Bob who computes $EQ_n(x, y)$. The number of bits exchanged is n clearly. Note that in some references (e.g., [18, 35]) it is required that both Alice and Bob know the final result. If so, one additional bit is required for Bob returning his result to Alice. Then $n + 1$ bits exchanged are necessary.

3 State succinctness of 2QCFA on promise problems

In this section, we will give an infinite family of promise problems[1] which can be solved by 2QCFA with one-sided error ϵ in a polynomial expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states.

A *promise problem* is a pair $A = (A_{yes}, A_{no})$, where A_{yes}, $A_{no} \subset \Sigma^*$ are disjoint sets of strings [36]. (Languages may be viewed as promise problems that obey the additional constraint $A_{yes} \cup A_{no} = \Sigma^*$.) For an alphabet $\Sigma = \{a, b\}$ and any $m \in \mathbb{Z}^+$, let $A^{eq}_{yes}(m) = \{a^m b^m\}$ and $A^{eq}_{no}(m) = \{w \in \{a, b\}^* | w \neq a^m b^m \ and \ |w| \geq m\}$. For any $\epsilon < 1/2$, we will prove that promise problems $A^{eq}(m) = (A^{eq}_{yes}(m), A^{eq}_{no}(m))$ can be solved by a 2QCFA with one-sided error ϵ in a polynomial expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states, whereas the sizes of the corresponding DFA, 2DFA and 2PFA grow without a bound.

In order to prove that the promise problem $A^{eq}(m)$ can be solved by 2QCFA, we first prove that a simpler promise problem can be solved by 2QCFA.

For any finite alphabet Σ and an $m \in \mathbb{Z}^+$, let $A_{yes}(m) = \{w \in \Sigma^* \mid |w| = m\}$ and $A_{no}(m) = \{w \in \Sigma^* \mid |w| \neq m \ and \ |w| \geq m/2\}$. For any $\epsilon < 1/2$, we will prove that there is a 2QCFA that can solve promise problem $A(m) = (A_{yes}(m), A_{no}(m))$ with one-sided error ϵ in a polynomial expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states. The language $L(m) = \{w \in \Sigma^* \mid |w| = m\}$ was showed to be recognized by a 7-state *one way quantum finite automata with restart* ($1QFA^{\circlearrowleft}$) with one-sided error ϵ in an exponential expected time by Yakaryilmaz and Cem Say [38]. In the same paper, they mentioned that $1QFA^{\circlearrowleft}$ can be simulated by 2QCFA easily. In following theorem we will prove in details that the promise problem $A(m)$ can be solved by a 2QCFA with one-sided error ϵ in a polynomial expected time.

Theorem 8. *For any $m \in \mathbb{Z}^+$ and any $\epsilon < 1/2$, there exists a 2QCFA $\mathcal{A}(m, \epsilon)$ which accepts any $w \in A_{yes}(m)$ with certainty, and rejects any $w \in A_{no}(m)$ with probability at least $1 - \epsilon$, where $QS(\mathcal{A}(m, \epsilon))$ is a constant and $CS(\mathcal{A}(m, \epsilon)) \in \mathbf{O}(\log \frac{1}{\epsilon})$. Furthermore, we have $T(\mathcal{A}(m, \epsilon)) \in \mathbf{O}(|w|^4)$, where w is the input.*

Proof. The main idea is as follows: we consider a 2QCFA $\mathcal{A}(m, \epsilon)$ with 2 quantum states $|q_0\rangle$ and $|q_1\rangle$. $\mathcal{A}(m, \epsilon)$ starts with the quantum state $|q_0\rangle$. When $\mathcal{A}(m, \epsilon)$ reads the left end-marker $\cent$, the state is rotated by the angle $\sqrt{2}m\pi$ and every time when $\mathcal{A}(m, \epsilon)$ reads a symbol $\sigma \in \Sigma^*$, the state is rotated by the angle $-\alpha = -\sqrt{2}\pi$ (notice that $\sqrt{2}m\pi = m\alpha$). When the

[1] A promise problem $A = (A_{yes}, A_{no})$ is solved by a 2QCFA $\mathcal{A}$ with one-sided error $\epsilon < 1/2$ if *(1)* $\forall w \in A_{yes}$, $Pr[\mathcal{A}$ *accepts* $w] = 1$, and *(2)* $\forall w \in A_{no}$, $Pr[\mathcal{A}$ *rejects* $w] \geq 1 - \epsilon$. A promise problem $A = (A_{yes}, A_{no})$ is solved by a 2PFA $\mathcal{A}$ with the error probability $\epsilon < 1/2$ if *(1)* $\forall w \in A_{yes}$, $Pr[\mathcal{A}$ *accepts* $w] \geq 1 - \epsilon$, and *(2)* $\forall w \in A_{no}$, $Pr[\mathcal{A}$ *rejects* $w] \geq 1 - \epsilon$. A promise problem $A = (A_{yes}, A_{no})$ is solved by a DFA (2DFA, 2NFA) $\mathcal{A}$ if *(1)* $\forall w \in A_{yes}$, $\mathcal{A}$ accepts w and *(2)* $\forall w \in A_{no}$, $\mathcal{A}$ rejects w.

```
Repeat the following ad infinitum:
1. Move the tape head to left end-marker, read the left end-marker $¢$, and perform $U_{¢}$ on $|q_0\rangle$.
2. Until the scanned symbol is the right end-marker $, do the following:
  (2.1) Perform $U_{-\alpha}$ on the current quantum state ($U_\alpha$ is defined in the proof of Theorem 8).
  (2.2) Move the tape head one square to the right.
3. Measure the quantum state. If the result is not $|q_0\rangle$, reject.
4. Repeat the following subroutine two times:
  (4.1) Move the tape head to the first input symbol.
  (4.2) Move the tape head one square to the right.
  (4.3) While the currently scanned symbol is not $¢$ or $, do the following:
      Simulate a coin flip. If the result is "head", move right. Otherwise, move left.
5. If both times the process ends at the right end-marker $, do:
    Simulate $k$ coin-flips and if all outcomes are "heads", accept.
```

Figure 1: Description of the behaviour of 2QCFA $\mathcal{A}(m,\epsilon)$. The choice of k will depend on ϵ.

right end-marker \$ is reached, $\mathcal{A}(m,\epsilon)$ measures the current quantum state. If the resulting quantum state is $|q_1\rangle$, the input string w is rejected. Otherwise, the process is repeated.

We now complete the description of $\mathcal{A}(m,\epsilon)$ as sketched in Figure 1. The quantum states of the automaton will be over the orthogonal base $\{|q_0\rangle, |q_1\rangle\}$ and will use the following two unitary transformations

$U_{¢}\|q_0\rangle = \cos m\alpha\|q_0\rangle + \sin m\alpha\|q_1\rangle$	$U_{-\alpha}\|q_0\rangle = \cos\alpha\|q_0\rangle - \sin\alpha\|q_1\rangle$
$U_{¢}\|q_1\rangle = -\sin m\alpha\|q_0\rangle + \cos m\alpha\|q_1\rangle$	$U_{-\alpha}\|q_0\rangle = \sin\alpha\|q_0\rangle + \cos\alpha\|q_1\rangle$

Lemma 9. *If the input $w \in A_{yes}(m)$, then the quantum state of $\mathcal{A}(m,\epsilon)$ will evolve with certainty into $|q_0\rangle$ after the loop **2**.*

Proof. If $w \in A_{yes}(m)$, then $|w| = m$. The quantum state after the loop **2** can be described as follows:

$$|q\rangle = (U_{-\alpha})^m U_{¢}|q_0\rangle = \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix}^m \begin{pmatrix} \cos m\alpha & -\sin m\alpha \\ \sin m\alpha & \cos m\alpha \end{pmatrix} |q_0\rangle \tag{6}$$

$$= \begin{pmatrix} \cos m\alpha & \sin m\alpha \\ -\sin m\alpha & \cos m\alpha \end{pmatrix} \begin{pmatrix} \cos m\alpha & -\sin m\alpha \\ \sin m\alpha & \cos m\alpha \end{pmatrix} |q_0\rangle = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} |q_0\rangle = |q_0\rangle. \tag{7}$$

□

Lemma 10. *If $w \in A_{no}(m)$, $|w| = n$, then $\mathcal{A}(m,\epsilon)$ rejects w after the step **3** with a probability at least $1/(2(m-n)^2+1)$.*

Proof. Starting with the state $|q_0\rangle$, $\mathcal{A}(m,\epsilon)$ changes its quantum state to $|q\rangle = (U_{-\alpha})^n U_{\cent}|q_0\rangle$ after the loop **2**, the quantum state can be described as follows:

$$|q\rangle = (U_{-\alpha})^n U_{\cent}|q_0\rangle = \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix}^n \begin{pmatrix} \cos m\alpha & -\sin m\alpha \\ \sin m\alpha & \cos m\alpha \end{pmatrix} |q_0\rangle \tag{8}$$

$$= \begin{pmatrix} \cos n\alpha & \sin n\alpha \\ -\sin n\alpha & \cos n\alpha \end{pmatrix} \begin{pmatrix} \cos m\alpha & -\sin m\alpha \\ \sin m\alpha & \cos m\alpha \end{pmatrix} |q_0\rangle \tag{9}$$

$$= \begin{pmatrix} \cos(m-n)\alpha & \sin(m-n)\alpha \\ \sin(m-n)\alpha & \cos(m-n)\alpha \end{pmatrix} |q_0\rangle = \cos((m-n)\alpha)|q_0\rangle + \sin((m-n)\alpha)|q_1\rangle. \tag{10}$$

The probability of observing $|q_1\rangle$ is $\sin^2(\sqrt{2}(m-n)\pi)$ in the step **3**. Without loss of generality, we assume that $m-n>0$. Let l be the closest integer to $\sqrt{2}(m-n)$. If $\sqrt{2}(m-n) > l$, then $2(m-n)^2 > l^2$. So we get $2(m-n)^2 - 1 \geq l^2$ and $l \leq \sqrt{2(m-n)^2-1}$. Therefore

$$\sqrt{2}(m-n) - l \geq \sqrt{2}(m-n) - \sqrt{2(m-n)^2-1} \tag{11}$$

$$= \frac{(\sqrt{2}(m-n) - \sqrt{2(m-n)^2-1})(\sqrt{2}(m-n) + \sqrt{2(m-n)^2-1})}{\sqrt{2}(m-n) + \sqrt{2(m-n)^2-1}} \tag{12}$$

$$= \frac{1}{\sqrt{2}(m-n) + \sqrt{2(m-n)^2-1}} > \frac{1}{2\sqrt{2}(m-n)}. \tag{13}$$

Because l is the closest integer to $\sqrt{2}(m-n)$, we have $0 < \sqrt{2}(m-n) - l < 1/2$. Let $f(x) = sin(x\pi) - 2x$. We have $f''(x) = -\pi^2\sin(x\pi) \leq 0$ when $x \in [0,1/2]$. That is to say, $f(x)$ is concave in $[0,1/2]$, and we have $f(0) = f(1/2) = 0$. So for any $x \in [0,1/2]$, it holds that $f(x) \geq 0$, that is, $\sin(x\pi) \geq 2x$. Therefore, we have

$$\sin^2(\sqrt{2}(m-n)\pi) = \sin^2((\sqrt{2}(m-n) - l)\pi) \tag{14}$$

$$\geq (2(\sqrt{2}(m-n) - l))^2 = 4(\sqrt{2}(m-n) - l)^2 \tag{15}$$

$$> 4(\frac{1}{2\sqrt{2}(m-n)})^2 = \frac{1}{2(m-n)^2} > \frac{1}{2(m-n)^2+1}. \tag{16}$$

If $\sqrt{2}(m-n) < l$, then $2(m-n)^2 < l^2$. So we get $2(m-n)^2 + 1 \leq l^2$ and $l \geq \sqrt{2(m-n)^2+1}$. We have

$$\sqrt{2}(m-n) - l \leq \sqrt{2}(m-n) - \sqrt{2(m-n)^2+1} \tag{17}$$

$$= \frac{(\sqrt{2}(m-n) - \sqrt{2(m-n)^2+1})(\sqrt{2}(m-n) + \sqrt{2(m-n)^2+1})}{\sqrt{2}(m-n) + \sqrt{2(m-n)^2+1}} \tag{18}$$

$$= \frac{-1}{\sqrt{2}(m-n) + \sqrt{2(m-n)^2+1}} < \frac{-1}{2\sqrt{2(m-n)^2+1}} \tag{19}$$

and this implies that

$$l-\sqrt{2}(m-n)>\frac{1}{2\sqrt{2(m-n)^2+1}}. \tag{20}$$

Because l is the closest integer to $\sqrt{2}(m-n)$, we have $0<l-\sqrt{2}(m-n)<1/2$. Therefore, we have

$$\sin^2(\sqrt{2}(m-n)\pi)=\sin^2((\sqrt{2}(m-n)-l)\pi) \tag{21}$$

$$=\sin^2((l-\sqrt{2}(m-n))\pi)\geq(2(l-\sqrt{2}(m-n)))^2 \tag{22}$$

$$=4(l-\sqrt{2}(m-n))^2>4(\frac{1}{2\sqrt{2(m-n)^2+1}})^2=\frac{1}{2(m-n)^2+1}. \tag{23}$$

So the lemma has been proved. □

Simulation of a coin flip in the steps **4** and **5** is a necessary component in the above algorithm. We will show that coin-flips can be simulated by a 2QCFA using two quantum states $|q_0\rangle$ and $|q_1\rangle$.

Lemma 11. *A coin flip in the algorithm can be simulated by a 2QCFA $\mathcal{A}(m,\epsilon)$ using two quantum states $|q_0\rangle$ and $|q_1\rangle$.*

Proof. Let us consider a projective measurement $M=\{P_0,P_1\}$ defined by

$$P_0=|q_0\rangle\langle q_0|, P_1=|q_1\rangle\langle q_1|, \tag{24}$$

whose classical outcomes will be denoted by 0 and 1, representing the "tail" and "head" of a coin flip, respectively. Hadamard unitary operator

$$H=\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \tag{25}$$

changes basis states as follows

$$|q_0\rangle\to|\psi\rangle=\frac{1}{\sqrt{2}}(|q_0\rangle+|q_1\rangle),\ \ |q_1\rangle\to|\phi\rangle=\frac{1}{\sqrt{2}}(|q_0\rangle-|q_1\rangle). \tag{26}$$

Suppose now that the machine starts with the state $|q_0\rangle$, changes its quantum state by H, and then measures the quantum state with M. Then we will get both result 0 or 1 with probability $\frac{1}{2}$. This is similar to a coin flip process. □

Lemma 12. *[3] If the length of the input string is n, then every execution of the loops 4 and* **5** *leads to the acceptance with a probability $1/(2^k(n+1)^2)$.*

Proof. The loop **4** performs two times of the random walk starting at location 1 and ending at location 0 (the left end-marker ¢) or at the location $n+1$ (the right end-marker \$). It is known from the probability theory that the probability of reaching the location $n+1$ is $1/(n+1)$ (see Chapter 14.2 in [12]). Repeating it twice and flipping k coins, we get the probability $1/(2^k(n+1)^2)$. □

After the step **5**, if $\mathcal{A}(m,\epsilon)$ accepts the input string, $\mathcal{A}(m,\epsilon)$ halts. Otherwise, the outcome of the last coin-flip is tail (it can happen in two cases: the outcome of the coin-flip is tail and the tape head reaches the left end-marker in the step **4**, and the outcome of the coin-flip is tail in step **5**.), and then the quantum state of $\mathcal{A}(m,\epsilon)$ must be $|q_0\rangle$ and $\mathcal{A}(m,\epsilon)$ starts a new iteration.

If we take $k = 1 + \lceil \log \frac{1}{\epsilon} \rceil$, then $\epsilon \geq 1/2^{k-1}$. Assume also that $|w| = n$. If $w \in A_{yes}(m)$, then the loop **2** always changes the quantum state $|q_0\rangle$ to $|q_0\rangle$, and $\mathcal{A}(m,\epsilon)$ never rejects after the measurement in the step **3**. After the loops **4** and **5**, the probability of $\mathcal{A}(m,\epsilon)$ accepting w is $1/(2^k(n+1)^2)$. Repeating the loops **4** and **5** for cn^2 times, the accepting probability is

$$Pr[\mathcal{A}(m,\epsilon)\ accepts\ w] = 1 - (1 - \frac{1}{2^k(n+1)^2})^{cn^2}, \tag{27}$$

and this can be made arbitrarily close to 1 by selecting the constant c appropriately.

Otherwise, if $|w| \in A_{no}(m)$, $\mathcal{A}(m,\epsilon)$ rejects the input after steps **2** and **3** with the probability

$$P_r > \frac{1}{2(m-n)^2+1} \tag{28}$$

according to Lemma 10. $\mathcal{A}(m,\epsilon)$ accepts the input after the loops **4** and **5** with the probability

$$P_a = 1/(2^k(n+1)^2) \leq \epsilon/(2(n+1)^2). \tag{29}$$

If we repeat the whole algorithm indefinitely, the probability of $\mathcal{A}(m,\epsilon)$ to reject the input w is

$$Pr[\mathcal{A}(m,\epsilon)\ rejects\ w] = \sum_{i\geq 0}(1-P_a)^i(1-P_r)^iP_r \tag{30}$$

$$= \frac{P_r}{P_a+P_r-P_aP_r} > \frac{P_r}{P_a+P_r} \tag{31}$$

$$> \frac{1/(2(n-m)^2+1)}{\epsilon/2(n+1)^2 + 1/(2(n-m)^2+1)} \tag{32}$$

$$= \frac{(n+1)^2/(2(n-m)^2+1)}{\epsilon/2 + (n+1)^2/(2(n-m)^2+1)} \tag{33}$$

Let $f(x) = \frac{x}{\epsilon/2+x} = 1 - \frac{\epsilon}{(\epsilon+2x)}$, then $f(x)$ is monotonously increasing in $(0,+\infty)$. By our assumption, we have $n = |w| \geq m/2$. So we have $(n+1)^2/(2(n-m)^2+1) > 1/2$. Therefore, we have

$$Pr[\mathcal{A}(m,\epsilon)\ rejects\ w] > \frac{1/2}{1/2+\epsilon/2} = \frac{1}{1+\epsilon} > 1-\epsilon. \tag{34}$$

If we assume that the input is w, then the step **1** takes $\mathbf{O}(1)$ time, the loop **2** and the step **3** take $\mathbf{O}(|w|)$ time, and the loops **4** and **5** take $\mathbf{O}(|w|^2)$ time. The expected number of the repetitions of the algorithm is $\mathbf{O}(|w|^2)$. Hence, the expected running time of $\mathcal{A}(m,\epsilon)$ is $\mathbf{O}(|w|^4)$. Obviously, $QS(\mathcal{A}(m,\epsilon)) = 2$. We just need $\mathbf{O}(k)$ classical states to simulate k coin-flips and calculate the outcomes. Therefore $CS(\mathcal{A}(m,\epsilon)) \in \mathbf{O}(\log \frac{1}{\epsilon})$. □

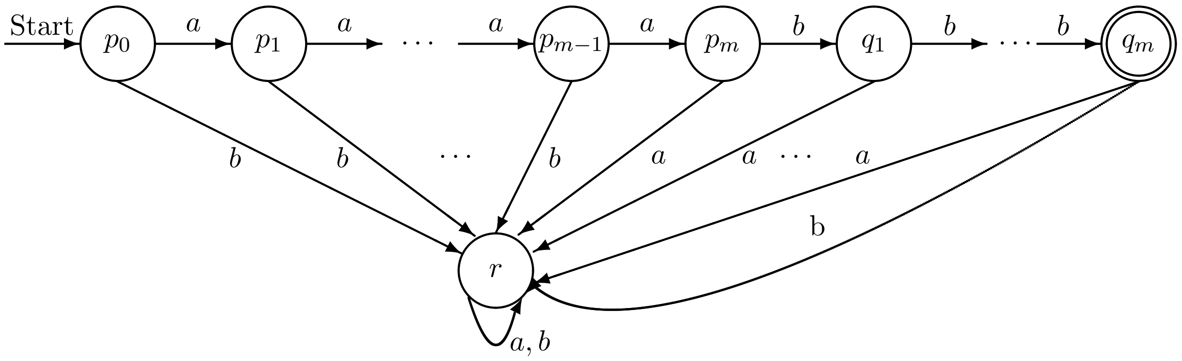

Figure 2: DFA $\mathcal{A}(m)$ to solve the promise problem $A^{eq}(m)$

Theorem 13. *For any $m \in \mathbb{Z}^+$ and any $\epsilon < 1/2$, there exists a 2QCFA $\mathcal{A}(m,\epsilon)$ which accepts any $w \in A^{eq}_{yes}(m)$ with certainty, and rejects any $w \in A^{eq}_{no}(m)$ with probability at least $1-\epsilon$, where $QS(\mathcal{A}(m,\epsilon))$ is a constant and $CS(\mathcal{A}(m,\epsilon)) \in \mathbf{O}(\log \frac{1}{\epsilon})$. Furthermore, we have $T(\mathcal{A}(m,\epsilon)) \in \mathbf{O}(|w|^4)$ where w is the input.*

Proof. Obviously, $A^{eq}(m) = L^{eq} \cap A(2m)$. According to Lemma 1, for any $\epsilon_1 > 0$, there is a 2QCFA $\mathcal{A}_1(\epsilon_1)$ that recognizes L^{eq} with the one-sided error ϵ_1, $QS(\mathcal{A}_1(\epsilon_1)) = 2$, $CS(\mathcal{A}_1(\epsilon_1)) \in \mathbf{O}(\log \frac{1}{\epsilon_1})$ and $T(\mathcal{A}_1(\epsilon_1)) \in \mathbf{O}(|w|^4)$. According to Theorem 8, for any $\epsilon_2 > 0$, there is a 2QCFA $\mathcal{A}_2(m,\epsilon_2)$ that solves the promise problem $A(2m)$ with the one-sided error ϵ_2, $QS(\mathcal{A}_2(m,\epsilon_2)) = 2$, $CS(\mathcal{A}_2(m,\epsilon_2)) \in \mathbf{O}(\log \frac{1}{\epsilon_2})$ and $T(\mathcal{A}_2(m,\epsilon_2)) \in \mathbf{O}(|w|^4)$. For any $\epsilon < 1/2$, let $\epsilon_1 = \epsilon/2$ and $\epsilon_2 = \epsilon/2$. According to Lemma 2, there is a 2QCFA $\mathcal{A}(m,\epsilon)$ that solves the promise problem $L^{eq} \cap A(2m)$ with the one-sided error ϵ, where $QS(\mathcal{A}(m,\epsilon)) = QS(\mathcal{A}_1(\epsilon_1)) + QS(\mathcal{A}_2(m,\epsilon_2)) = 4$, $CS(\mathcal{A}(m,\epsilon)) = CS(\mathcal{A}_1(\epsilon_1)) + CS(\mathcal{A}_2(m,\epsilon_2)) + QS(\mathcal{A}_1(\epsilon_1)) \in \mathbf{O}(\log \frac{1}{\epsilon})$ and $T(\mathcal{A}(m,\epsilon)) = T(\mathcal{A}_1(\epsilon_1)) + T(\mathcal{A}_2(m,\epsilon_2)) \in \mathbf{O}(|w|^4)$. Hence, the theorem has been proved. □

Remark 4. Actually, L_1 and L_2 must be languages in Lemma 2. But in Theorem 13, we used a promise problem $A(2m)$. It is easy to show that Lemma 2 still holds for promise problem $A(2m)$ and language L^{eq}. We used Lemma 2 to prove Theorem 13 in this section. However, we can prove Theorem 13 directly.

Obviously, there exists a DFA depicted in Figure 2 that solves the promise problem $A^{eq}(m)$ with $2m+2$ states.

Theorem 14. *For any $m \in \mathbb{Z}^+$, any DFA solving the promise problem $A^{eq}(m)$ has at least $2m+2$ states.*

Proof. Let us consider the string set $W = \{a^0, a^1, \cdots, a^m, a^m b^1, a^m b^2, \cdots, a^m b^m\}$, where a^0 is the empty string. Obviously, for any two different strings $w_i, w_j \in W$, we have $|w_i| \neq |w_j|$, and if $|w_i| < |w_j|$, then w_i is a prefix of w_j. For any string $x \in \Sigma^*$ and any $\sigma \in \Sigma$, let $\widehat{\delta}(s, \sigma x) = \widehat{\delta}(\delta(s,\sigma), x)$; if $|x| = 0$, $\widehat{\delta}(s, x) = s$ [15]. Assume that an n-state DFA $\mathcal{A}(m)$ solves promise problem $A^{eq}(m)$. We show that n cannot be less than $2m+2$.

Assume that s_0 is the initial state of $\mathcal{A}(m)$, and that there are two different strings $w_i, w_j \in W$ such that $\widehat{\delta}(s_0, w_i) = \widehat{\delta}(s_0, w_j)$. Without loss of generality, we assume that w_i is a prefix of w_j, so there is a string x such that $w_j = w_i x$, where $|x| \neq 0$. Let $\widehat{\delta}(s_0, w_i) = s$, we have $\widehat{\delta}(s, x) = \widehat{\delta}(s, x^*) = s$. Because w_i is a prefix of $a^m b^m$, there exists a string y such that $\widehat{\delta}(s_0, w_i y) = \widehat{\delta}(s, y) = s_{acc}$, where s_{acc} is an accepting state. This implies $\widehat{\delta}(s_0, w_i x^* y) = s_{acc}$. Therefore, there is a $k \in \mathbb{Z}^+$ such that $\widehat{\delta}(s_0, w_i x^k y) = s_{acc}$ and $w_i x^k y \in A^{eq}_{no}(m)$, which is a contradiction. Hence, for any two different strings $w_i, w_j \in W$ we have that $\widehat{\delta}(s_0, w_i) \neq \widehat{\delta}(s_0, w_j)$.

For any $w_i \in W$, $\widehat{\delta}(s_0, w_i)$ is a reachable state (i.e., there exists a string z such that $\widehat{\delta}(\widehat{\delta}(s_0, w_i), z)$ is an accepting state). Therefore, there must be at least one state that is not reachable, for example, $\widehat{\delta}(s_0, a^m b^{m+1})$. There is $2m+1$ elements in the set W and at least one that does not lead to a reachable state. So any DFA solving the promise problem $A^{eq}(m)$ has at least $2m+2$ states.

□

Theorem 15. *For any $m \in \mathbb{Z}^+$, any 2DFA, 2NFA and any polynomial expected running time 2PFA solving the promise problem $A^{eq}(m)$ has at least $\sqrt{\log m}$, $\sqrt{\log m}$ and $\sqrt[3]{(\log m)/b}$ states, where b is a constant.*

Proof. Assume that an n_1-state 2DFA $\mathcal{A}(m)$ solves the promise problem $A^{eq}(m)$. It is easy to prove that $n_1 \geq 3$. According to Lemma 3, there is a DFA that solves the promise problem $A^{eq}(m)$ with $(n_1+1)^{n_1+1}$ states. According to Theorem 14, we have

$$(n_1+1)^{n_1+1} \geq 2m+2 \Rightarrow (n_1+1)\log(n_1+1) > \log m + 1. \tag{35}$$

Because $n_1 \geq 3$, we get

$$n_1^2 > (n_1+1)\log(n_1+1) > \log m \Rightarrow n_1 > \sqrt{\log m}. \tag{36}$$

Assume that an n_2-state 2NFA $\mathcal{A}(m)$ solves the promise problem $A^{eq}(m)$. According to Lemma 4, there is a DFA that solves the promise problem $A^{eq}(m)$ with $2^{(n_2-1)^2+n_2}$ states. According to Theorem 14, we have

$$2^{(n_2-1)^2+n_2} \geq 2m+2 \Rightarrow (n_2-1)^2 + n_2 > \log m + 1 \tag{37}$$

$$\Rightarrow n_2^2 > \log m \Rightarrow n_2 > \sqrt{\log m}. \tag{38}$$

Assume that an n_3-state 2PFA $\mathcal{A}(m)$ solves the promise problem $A^{eq}(m)$ with the error probability $\epsilon < 1/2$ and within a polynomial expected running time. Suppose L is the language recognized by 2PFA $\mathcal{A}(m)$. For any $w \in A^{eq}_{yes}(m)$, we have $w \in L$, and for any $w \in A^{eq}_{no}(m)$, we have $w \notin L$. According to Lemma 5, there is a DFA $\mathcal{A}'(m)$ that recognizes L with $n_3^{bn_3^2}$ states, where $b > 0$ is a constant. For any $w \in A^{eq}_{yes}(m)$, $\mathcal{A}'(m)$ accepts w, and for any $w \in A^{eq}_{no}(m)$, $\mathcal{A}'(m)$ rejects w. Therefor, the promise problem $A^{eq}(m)$ can be solved by DFA $\mathcal{A}'(m)$. According to Theorem 14, we have

$$n_3^{bn_3^2} \geq 2m + 2 \Rightarrow bn_3^2 \log n_3 > \log m \tag{39}$$

$$\Rightarrow n_3^3 > (\log m)/b \Rightarrow n_3 > \sqrt[3]{(\log m)/b}. \tag{40}$$

□

4 State succinctness of 2QCFA

For the alphabet $\Sigma = \{a, b, c\}$ and any $m \in \mathbb{Z}^+$, let $L^{twin}(m) = \{wcw | w \in \{a, b\}^*, |w| = m\}$. For any $\epsilon < 1/2$, we will prove that $L^{twin}(m)$ can be recognized by a 2QCFA with one-sided error ϵ in an exponential expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states. The language $L^{twin} = \{wcw|w \in \{a, b\}^*\}$ over the alphabet $\Sigma = \{a, b, c\}$ was claimed to be recognized by a 2QCFA by Yakaryilmaz and Cem Say [38]. However, they did not give the details of such a 2QCFA. In the following, we will show such an automaton and its behavior in detail.

Theorem 16. *For any $\epsilon < 1/2$, there exists a 2QCFA $\mathcal{A}(\epsilon)$ which accepts any $w \in L^{twin}$ with certainty, rejects any $w \notin L^{twin}$ with the probability at least $1 - \epsilon$, and halts in exponential expected time, where $QS(\mathcal{A}(\epsilon))$=3 and $CS(\mathcal{A}(\epsilon)) \in \mathbf{O}(\log \frac{1}{\epsilon})$.*

Proof. Let us consider matrixes

$$A = \begin{pmatrix} 4 & 3 & 0 \\ -3 & 4 & 0 \\ 0 & 0 & 5 \end{pmatrix}, B = \begin{pmatrix} 4 & 0 & 3 \\ 0 & 5 & 0 \\ -3 & 0 & 4 \end{pmatrix}. \tag{41}$$

We now describe formally a 2QCFA $\mathcal{A}(\epsilon)$, that is described less formally in Figure 3, with 3 quantum states $\{|q_0\rangle, |q_1\rangle, |q_2\rangle\}$, with $|q_0\rangle$ being the initial state. $\mathcal{A}(\epsilon)$ has two unitary operators $U_a = \frac{1}{5}A$ and $U_b = \frac{1}{5}B$ given in Eq. (41). They can also be described as follows:

$U_a\|q_0\rangle = \frac{4}{5}\|q_0\rangle - \frac{3}{5}\|q_1\rangle$	$U_b\|q_0\rangle = \frac{4}{5}\|q_0\rangle - \frac{3}{5}\|q_2\rangle$
$U_a\|q_1\rangle = \frac{3}{5}\|q_0\rangle + \frac{4}{5}\|q_1\rangle$	$U_b\|q_1\rangle = \|q_1\rangle$
$U_a\|q_2\rangle = \|q_2\rangle$	$U_b\|q_2\rangle = \frac{3}{5}\|q_0\rangle + \frac{4}{5}\|q_2\rangle$

We now summarize some concepts and results from [3] that we will use to prove the theorem. For $u \in \mathbb{Z}^3$, we use $u[i]$ $(i = 1, 2, 3)$ to denote the ith entry of u. We define a function $f : \mathbb{Z}^3 \to \mathbb{Z}$ as

$$f(u) = 4u[1] + 3u[2] + 3u[3] \tag{42}$$

for each $u \in \mathbb{Z}^3$, and we define a set $K \subseteq \mathbb{Z}^3$ as

$$K = \{u \in \mathbb{Z}^3 : u[1] \not\equiv 0(mod\ 5), f(u) \not\equiv 0(mod\ 5), and\ u[2] \cdot u[3] \equiv 0(mod\ 5)\} \tag{43}$$

```
Check whether the input is of the form $xcy$ ($x, y \in \{a, b\}^*$). If not, reject.
Otherwise, repeat the following ad infinitum:
  1. Move the tape head to the first input symbol.
  2. Until the currently scanned symbol $\sigma$ is $c$, do the following:
    (2.1) Perform $U_\sigma$ on the quantum state.
    (2.2) Move the tape head one square to the right.
  3. Move the tape head to the last input symbol.
  4. Until the currently scanned symbol $\sigma$ is $c$, do the following:
    (4.1) Perform $U_\sigma^{-1}$ on the quantum state.
    (4.2) Move the tape head one square to the left.
  5. Measure the quantum state. If the result is not $|q_0\rangle$, reject.
  6. Move the tape head to the last input symbol and set $flag = 0$.
  7. While the currently scanned symbol is not ¢, do the following:
    (7.1) Simulate $k$ coin-flips. Set $flag = 1$ in case all results are not "heads".
    (7.2) Move the tape head one square to the left.
  8. If $flag = 0$, accept.
```

Figure 3: Informal description of the actions of a 2QCFA for L^{twin}. The choice of k will depend on ϵ.

Lemma 17 ([3]). *If $u \in K$, then $Au \in K$ and $Bu \in K$.*

Lemma 18 ([3]). *If an $u \in \mathbb{Z}^3$ is such that $u = Av = Bw$ for some $v, w \in \mathbb{Z}^3$, then $u \notin K$.*

Lemma 19. *If $u \in K$, there does not exist an $l \in \mathbb{Z}^+$ such that $Xu = \pm 5^l (1, 0, 0)^T$, where $X \in \{A, B\}$.*

Proof. Suppose there is an $l \in \mathbb{Z}^+$ such that $Xu = \pm 5^l (1, 0, 0)^T$. Assume that $X = A$ (the proof for $X = B$ is similar), then it holds

$$Xu = Au = \begin{pmatrix} 4 & 3 & 0 \\ -3 & 4 & 0 \\ 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} u[1] \\ u[2] \\ u[3] \end{pmatrix} = \begin{pmatrix} 4u[1] + 3u[2] \\ -3u[1] + 4u[2] \\ 5u[3] \end{pmatrix} = \pm \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} 5^l \tag{44}$$

$$\Rightarrow \begin{pmatrix} u[1] \\ u[2] \\ u[3] \end{pmatrix} = \pm \begin{pmatrix} 4 \cdot 5^{l-2} \\ 3 \cdot 5^{l-2} \\ 0 \end{pmatrix}. \tag{45}$$

Since $4u[1] + 3u[2] + 3u[3] = \pm(16 \cdot 5^{l-2} + 9 \cdot 5^{l-2}) = \pm 5^l$, we conclude $f(u) \equiv 0(mod\ 5)$. We get that $u \notin K$, which contradicts the fact that $u \in K$. Hence, the Lemma has been proved. □

Corollary 20. *Let*

$$u = X_k \cdots X_1 (1,0,0)^T, \tag{46}$$

where $X_i \in \{A, B\}$. Then $u = \pm 5^l (1,0,0)^T$ for no $l \in \mathbb{Z}^+$.

Proof. Clearly, $(1,0,0)^T \in K$. According to Lemma 17, $X_{k-1} \cdots X_1 (1,0,0)^T \in K$. According to Lemma 19, there does not exist $l \in \mathbb{Z}^+$ such that $u = \pm 5^l (1,0,0)^T$. □

Lemma 21. *Let*

$$u = Y_1^{-1} \cdots Y_k^{-1} (1,0,0)^T, \tag{47}$$

where $Y_i \in \{A, B\}$. Then $u = \pm \frac{1}{5^l} (1,0,0)^T$ for no $l \in \mathbb{Z}^+$.

Proof. Assume that there is an $l \in \mathbb{Z}^+$ such that $u = Y_1^{-1} \cdots Y_k^{-1} (1,0,0)^T = \pm \frac{1}{5^l} (1,0,0)^T$, then we get $Y_k \cdots Y_1 (1,0,0)^T = \pm 5^l (1,0,0)^T$. According to Corollary 20, such l does not exist. □

Lemma 22. *Let*

$$u = (5Y_1^{-1}) \cdots (5Y_m^{-1})(5^{-1}X_n) \cdots (5^{-1}X_1)(1,0,0)^T, \tag{48}$$

where $X_j, Y_j \in \{A, B\}$. If $m = n$ and $X_j = Y_j$ for $1 \leq j \leq n$, then $u[2]^2 + u[3]^2 = 0$. Otherwise, $u[2]^2 + u[3]^2 > 5^{-(n+m)}$.

Proof. If $m = n$ and $X_j = Y_j$ for $1 \leq j \leq n$, then we have

$$u = Y_1^{-1} \cdots Y_n^{-1} X_n \cdots X_1 (1,0,0)^T = (1,0,0)^T, \tag{49}$$

and thus $u[2]^2 + u[3]^2 = 0$.

Otherwise, note that $||u|| = 1$. $5^{-1}X_j$ and $5Y_j^{-1}$ are unitary for each j, and also note that $5^{(n+m)}u[i]$ $(i = 1, 2, 3)$ is an integer. It therefore suffices to prove that $u \neq \pm(1,0,0)^T$. $|u[1]| < 1$ implies $|u[1]| \leq 1 - 5^{-(n+m)}$, and therefore

$$u[2]^2 + u[3]^2 = 1 - u[1]^2 \geq 1 - (1 - 5^{-(n+m)})^2 > 5^{-(n+m)}. \tag{50}$$

We first prove the case that $n \geq m$. If $X_{n-j} = Y_{m-j}$ for $0 \leq j \leq m-1$, then

$$u = (5Y_1^{-1}) \cdots (5Y_m^{-1})(5^{-1}X_n) \cdots (5^{-1}X_1)(1,0,0)^T = 5^{-(n-m)} X_{n-m} \cdots X_1 (1,0,0)^T. \tag{51}$$

According to Corollary 20, for every $l \in \mathbb{Z}^+$,

$$u = 5^{-(n-m)} X_{n-m} \cdots X_1 (1,0,0)^T \neq \pm 5^{-(n-m)} 5^l (1,0,0)^T. \tag{52}$$

This implies that $u \neq \pm(1,0,0)^T$ if $l = n - m$.

Next suppose there exist an $i < m$ such that $X_{n-i} \neq Y_{m-i}$. Let k be the smallest integer such that $X_{n-k} \neq Y_{m-k}$, and without loss of generality suppose $X_{n-k} = A, Y_{m-k} = B$. Since $X_{n-j} = Y_{m-j}$ for $j < k$, we have

$$u = (5Y_1^{-1}) \cdots (5Y_m^{-1})(5^{-1}X_n) \cdots (5^{-1}X_1)(1,0,0)^T = 5^{-(n-m)}Y_1^{-1} \cdots Y_{m-k}^{-1}X_{n-k} \cdots X_1(1,0,0)^T. \tag{53}$$

For $u = (1,0,0)^T$, we get

$$u = 5^{-(n-m)}Y_1^{-1} \cdots Y_{m-k}^{-1}X_{n-k} \cdots X_1(1,0,0)^T = (1,0,0)^T \tag{54}$$

$$\Rightarrow X_{n-k} \cdots X_1(1,0,0)^T = 5^{(n-m)}Y_{m-k} \cdots Y_1(1,0,0)^T. \tag{55}$$

Obviously, $(1,0,0)^T \in K$. Let $v = X_{n-k-1} \cdots X_1(1,0,0)^T$ and $w = Y_{m-k-1} \cdots Y_1(1,0,0)^T$. According to Lemma 17, we have $v, w \in K$, $X_{n-k}v = Av \in K$, and $Y_{m-k}w = Bw \in K$. If $n = m$, by Lemma 18 this implies $Av \neq Bw$, which contradicts the Equation 55. If $n > m$, since $Bw \in K$, we get $5^{n-m}Bw \notin K$. Therefor, $Aw \neq 5^{n-m}Bw$, which contradicts the Equation 55. From that we conclude $u \neq (1,0,0)^T$. By similar reasoning we get that, $u \neq -(1,0,0)^T$.

Now we deal with the case $n < m$. If $X_{n-j} = Y_{m-j}$ for $0 \leq j \leq n-1$, then

$$u = (5Y_1^{-1}) \cdots (5Y_m^{-1})(5^{-1}X_n) \cdots (5^{-1}X_1)(1,0,0)^T = 5^{m-n}Y_1^{-1} \cdots Y_{m-n}^{-1}(1,0,0)^T. \tag{56}$$

According to Lemma 21, for every $l \in \mathbb{Z}^+$,

$$u = 5^{m-n}Y_1^{-1} \cdots Y_{m-n}^{-1}(1,0,0)^T \neq \pm 5^{m-n}5^{-l}(1,0,0)^T. \tag{57}$$

This implies that $u \neq \pm(1,0,0)^T$ if $l = m-n$.

Let us assume that there exists a $j < n$ such that $X_{n-j} \neq Y_{m-j}$. Let k be the smallest index such that $X_{n-k} \neq Y_{m-k}$. By similar reasoning as in the case $n \geq m$, we get $u \neq \pm(1,0,0)^T$. □

If the input w is not of the form xcy, $\mathcal{A}(\epsilon)$ rejects w immediately.

Lemma 23. *If the input $w = xcy$ and $x = y$, then the quantum state of $\mathcal{A}(\epsilon)$ will evolve into $|q_0\rangle$ with certainty after the loop* ***4***.

Proof. Let $x = x_1x_2 \ldots x_l = y = y_1y_2 \ldots y_l$ for some l. Starting with the state $|q_0\rangle$, $\mathcal{A}(\epsilon)$ changes its quantum state to $|\psi\rangle$ after the loop **4**, where

$$|\psi\rangle = U_{y_1}^{-1}U_{y_2}^{-1} \cdots U_{y_l}^{-1}U_{x_l} \cdots U_{x_2}U_{x_1}|q_0\rangle = U_{x_1}^{-1}U_{x_2}^{-1} \cdots U_{x_l}^{-1}U_{x_l} \cdots U_{x_2}U_{x_1}|q_0\rangle = |q_0\rangle. \tag{58}$$

□

Lemma 24. *If the input $w = xcy$ and $x \neq y$, then $\mathcal{A}(\epsilon)$ rejects w after the step* ***5*** *with the probability at least $5^{-(m+n)}$.*

Proof. Let $x = x_1x_2\cdots x_n$, $y = y_1y_2\cdots y_m$. Starting with the state $|q_0\rangle$, $\mathcal{A}(\epsilon)$ changes its quantum state after the loop 4 to:

$$|\psi\rangle = U_{y_1}^{-1}U_{y_2}^{-1}\cdots U_{y_m}^{-1}U_{x_n}\cdots U_{x_2}U_{x_1}|q_0\rangle. \tag{59}$$

Let $|\psi\rangle = \beta_0|q_0\rangle + \beta_1|q_1\rangle + \beta_2|q_2\rangle$. According to Lemma 22, $\beta_1^2+\beta_2^2 > 5^{-(n+m)}$. In the step **5**, the quantum state $|\psi\rangle$ is measured, $\mathcal{A}(\epsilon)$ then rejects w with the probability $p_r = \beta_1^2+\beta_2^2 > 5^{-(n+m)}$. □

Every execution of the steps **6**, **7** and **8** leads to an acceptance with the probability $2^{-k(n+m+1)}$.

After the step **8**, if $\mathcal{A}(\epsilon)$ accepts the input string, $\mathcal{A}(\epsilon)$ halts. Otherwise, the outcome of the last coin-flip is tail, and then the quantum state of $\mathcal{A}(\epsilon)$ must be $|q_0\rangle$ and $\mathcal{A}(\epsilon)$ starts a new iteration.

Let $k \geq \max\{\log 5, \log\frac{1}{\epsilon}\}$. Assume that the input is of the form $w = xcy$. If $x = y$, 2QCFA $\mathcal{A}(\epsilon)$ always changes its quantum state to $|q_0\rangle$ after the loop **4**, and $\mathcal{A}(\epsilon)$ never rejects the input after the measurement in the step **5**. After the steps **6**, **7** and **8**, the probability of $\mathcal{A}(\epsilon)$ accepting w is $2^{-k(n+m+1)}$. Repeating the whole iteration for $c2^{k(n+m+1)}$ times, the accepting probability is

$$Pr[\mathcal{A}(\epsilon) \ accepts \ w] = 1-(1-2^{-k(n+m+1)})^{c2^{k(n+m+1)}}, \tag{60}$$

and this can be made arbitrarily close to 1 by selecting constant c appropriately.

Otherwise, if $x \neq y$, then, according to Lemma 24, $\mathcal{A}(\epsilon)$ rejects the input after the step **5** with the probability

$$P_r > 5^{-(m+n)} \tag{61}$$

and, $\mathcal{A}(\epsilon)$ accepts the input after the steps **6**, **7** and **8** with the probability

$$P_a = 2^{-k(n+m+1)}. \tag{62}$$

If we repeat the whole iteration indefinitely, the probability of $\mathcal{A}(\epsilon)$ rejecting input w is

$$Pr[\mathcal{A}(\epsilon) \ rejects \ w] = \sum_{i\geq 0}(1-P_a)^i(1-P_r)^iP_r \tag{63}$$

$$= \frac{P_r}{P_a+P_r-P_aP_r} > \frac{P_r}{P_a+P_r} \tag{64}$$

$$> \frac{5^{-(m+n)}}{2^{-k(n+m+1)}+5^{-(m+n)}} \tag{65}$$

$$> \frac{1}{1+\epsilon} > 1-\epsilon. \tag{66}$$

If the input is w, then the step **1** takes $\mathbf{O}(1)$ time, the steps **2** and **3** take $\mathbf{O}(|w|)$ time, the loops **4** and **5** take $\mathbf{O}(|w|)$ time, the steps **6**, **7** and **8** take $\mathbf{O}(|w|)$ time. The expected number of iterations is $\mathbf{O}(2^{k|w|})$. Hence, the expected running time of $\mathcal{A}(\epsilon)$ is $\mathbf{O}(|w|2^{k|w|})$. Obviously, the 2QCFA $\mathcal{A}(\epsilon)$ has three quantum states. We just need $\mathbf{O}(k)$ classical states to simulate k coin-flips and calculate the outcomes, therefore $CS(\mathcal{A}(\epsilon)) \in \mathbf{O}(\log \frac{1}{\epsilon})$.

□

In Theorem 16, we have proved that L^{twin} can be recognized by 2QCFA. We will show that L^{twin} can not be recognized by 2PFA with any error probability $\epsilon < 1/2$. Thus L^{twin} is another witness of the fact that 2QCFA are more powerful than their classical counterparts 2PFA.

Theorem 25. *There is no 2PFA recognizing L^{twin} with error probability $\epsilon < 1/2$.*

Proof. Let $A = L^{twin}$ and $B = \overline{L^{twin}} = \Sigma^* \setminus A$. For each $m \in I$, let $W_m = \{w | w \in \{a,b\}^*$ *and* $|w| \leq m\}$. Obviously, $W_m \subseteq \Sigma^*$ satisfying conditions (1) and (2) of Lemma 6. For every $m \in I$ and every $w, w' \in W_m$ with $w \neq w'$, if we take $u = \lambda$ (the empty word) and $v = cw$, then $uwv = wcw \in A$ and $uw'v = w'cw \in B$. According to Lemma 6, there is no 2PFA separating A and B. Thus, there is no 2PFA recognizing L^{twin} and the Theorem has been proved. □

For any finite alphabet Σ and an $m \in \mathbb{Z}^+$, let $L(m) = \{w \in \Sigma^* \mid |w| = m\}$.

Lemma 26 ([38])**.** *For any $\epsilon < 1/2$, there exists a 7-state 1QFA$^{\circlearrowleft}$ $\mathcal{A}(m,\epsilon)$ which accepts any $w \in L(m)$ with certainty, and rejects any $w \notin L(m)$ with probability at least $1-\epsilon$. Moreover, the expected runtime of the $\mathcal{A}(m,\epsilon)$ on w is $\mathbf{O}(2^{|w|}|w|)$.*

Lemma 27 ([38])**.** *For any 1QFA$^{\circlearrowleft}$ $\mathcal{A}_1$ with n quantum states and the expected running time $t(|w|)$, there exists a 2QCFA $\mathcal{A}_2$ with n quantum states, $\mathbf{O}(n)$ classical states, and expected running time $\mathbf{O}(t(|w|))$, such that $\mathcal{A}_2$ accepts every input string w with the same probability that $\mathcal{A}_1$ accepts w.*

Theorem 28. *For any $\epsilon < 1/2$, there exists a 2QFA $\mathcal{A}(m,\epsilon)$ which accepts any $w \in \Sigma^* \setminus L(m)$ with certainty, and rejects any $w \notin L(m)$ with the probability at least $1-\epsilon$. Moreover, $QS(\mathcal{A}(m,\epsilon)) = 7$, $CS(\mathcal{A}(m,\epsilon))$ is a constant, and the expected runtime of $\mathcal{A}(m,\epsilon)$ on w is $\mathbf{O}(2^{|w|}|w|)$.*

Proof. It follows from Lemma 26 and Lemma 27. □

Theorem 29. *For any $\epsilon < 1/2$, there exists a 2QFA $\mathcal{A}(m,\epsilon)$ which for $w \in \Sigma^*$ accepts any $w \in L^{twin}(m)$ with certainty, and rejects any $w \notin L^{twin}(m)$ with the probability at least $1-\epsilon$. Moreover, $QS(\mathcal{A}(m,\epsilon))$ is a constant, $CS(\mathcal{A}(m,\epsilon)) \in \mathbf{O}(\log \frac{1}{\epsilon})$, and the expected running time of $\mathcal{A}(m,\epsilon)$ on w is $\mathbf{O}(|w|2^{k|w|})$.*

Proof. Obviously, $L^{twin}(m) = L^{twin} \cap L(2m+1)$. According to Theorem 16, for any $\epsilon_1 < 1/2$, there is a 2QCFA $\mathcal{A}_1(\epsilon_1)$ that recognizes L^{twin} with the one-sided error ϵ_1, and $QS(\mathcal{A}_1(\epsilon_1)) = 3$, $CS(\mathcal{A}_1(\epsilon_1)) \in \mathbf{O}(\log \frac{1}{\epsilon_1})$ and $T(\mathcal{A}_1(\epsilon_1) \in \mathbf{O}(|w|2^{k|w|})$ where k is a constant. According to Theorem 28, for any $\epsilon_2 < 1/2$, there is a 2QCFA $\mathcal{A}_2(m, \epsilon)$ that recognizes $L(2m+1)$ with the one-sided error ϵ_2, and $QS(\mathcal{A}_2(m, \epsilon)) = 7$, $CS(\mathcal{A}_2(m, \epsilon))$ is a constant and $T(\mathcal{A}_2(m, \epsilon)) \in \mathbf{O}(2^{|w|}|w|)$. For any $\epsilon < 1/2$, let $\epsilon_1 = \epsilon/2$ and $\epsilon_2 = \epsilon/2$. According to Lemma 2, there is a 2QCFA $\mathcal{A}(m, \epsilon)$ recognizing $L^{twin} \cap L(2m+1)$ with the one-sided error ϵ, where $QS(\mathcal{A}(m, \epsilon)) = QS(\mathcal{A}_1(\epsilon_1)) + QS(\mathcal{A}_2(m, \epsilon)) = 10$, $CS(\mathcal{A}(m, \epsilon)) = CS(\mathcal{A}_1(\epsilon_1)) + CS(\mathcal{A}_2(m, \epsilon)) + QS(\mathcal{A}_1(\epsilon_1)) \in \mathbf{O}(\log \frac{1}{\epsilon})$ and $T(\mathcal{A}(m, \epsilon)) = T(\mathcal{A}_1(\epsilon_1)) + T(\mathcal{A}_2(m, \epsilon)) \in \mathbf{O}(|w|2^{k|w|})$. Hence, the theorem has been proved. □

For a fixed $m \in \mathbb{Z}^+$, $L^{twin}(m)$ is finite, and thus there exists a DFA accepting the language $L^{twin}(m)$. In the following theorem we use methods and results of *communication complexity* to derive a lower bound on the number of states of finite automata accepting the language $L^{twin}(m)$.

Theorem 30. *For any $m \in \mathbb{Z}^+$, any DFA recognizing $L^{twin}(m)$ has at least 2^m states.*

Proof. Assume that a DFA $\mathcal{A}(m)$ recognizes $L^{twin}(m)$. For an input string xcy of $L^{twin}(m)$ let us consider the following communication protocol between Alice and Bob with Alice having x at the beginning and Bob having y at the beginning. A protocol can be derived for $EQ_m(x, y)$ as follows: Alice first simulates the path taken by DFA $\mathcal{A}(m)$ on her input x. She then sends the name of the last state s in this path to Bob, which needs $\log(|S|)$ bits, where S is the set of states in DFA $\mathcal{A}(m)$. Afterwards, Bob simulates DFA $\mathcal{A}(m)$, starting from the state s, on input cy. All together, they get a simulation of the DFA $\mathcal{A}(m)$ on the input $w = xcy$. By the above assumptions, if $w = xcy$ is accepted by DFA $\mathcal{A}(m)$ then $EQ_m(x, y) = 1$ while if w is rejected then $EQ_m(x, y) = 0$. Therefore, we have $D(EQ_m) \leq \log(|S|)$. According to Lemma 7, we have

$$D(EQ_m) = m \leq \log(|S|) \tag{67}$$

$$\Rightarrow m \leq \log(|S|) \Rightarrow |S| \geq 2^m. \tag{68}$$

□

Theorem 31. *For any $m \in \mathbb{Z}^+$, any 2DFA, 2NFA and polynomial expected running time 2PFA recognizing $L^{twin}(m)$ have at least $\sqrt{m}$, $\sqrt{m}$ and $\sqrt[3]{m/b}$ states, where b is a constant.*

Proof. Assume that an n_1-state 2DFA $\mathcal{A}(m)$ recognizes $L^{twin}(m)$. It is easy to prove that $n_1 \geq 3$. According to Lemma 3, there is a DFA that recognizes $L^{twin}(m)$ with $(n_1+1)^{n_1+1}$ states. According to Theorem 30, we have

$$(n_1+1)^{n_1+1} \geq 2^m \Rightarrow (n_1+1)\log(n_1+1) \geq m. \tag{69}$$

Because $n_1 \geq 3$, we get

$$n_1^2 > (n_1+1)\log(n_1+1) > m \Rightarrow n_1 > \sqrt{m}. \tag{70}$$

Assume that an n_2-state 2NFA $\mathcal{A}(m)$ that recognizes $L^{twin}(m)$. According to Lemma 4, there is a DFA that recognizes $L^{twin}(m)$ with $2^{(n_2-1)^2+n_2}$ states. According to Theorem 30, we have

$$2^{(n_2-1)^2+n_2} \geq 2^m \Rightarrow (n_2-1)^2 + n_2 \geq m \tag{71}$$

$$\Rightarrow n_2^2 > m \Rightarrow n_2 > \sqrt{m}. \tag{72}$$

Assume that an n_3-state 2PFA $\mathcal{A}(m)$ recognizes $L^{twin}(m)$ with an error probability $\epsilon < 1/2$ and also within a polynomial expected running time. According to Lemma 5, there is a DFA that recognizes $L^{twin}(m)$ with $n_3^{bn_3^2}$ states, where $b > 0$ is a constant. According to Theorem 30, we have

$$n_3^{bn_3^2} \geq 2^m \Rightarrow bn_3^2 \log n_3 \geq m \tag{73}$$

$$\Rightarrow n_3^3 > m/b \Rightarrow n_3 > \sqrt[3]{m/b}. \tag{74}$$

□

5 Concluding remarks

2QCFA were introduced by Ambainis and Watrous [3]. In this paper, we investigated state succinctness of 2QCFA. We have showed that 2QCFA can be more space-efficient than their classical counterparts DFA, 2DFA, 2NFA and polynomial expected running time 2PFA, where the superiority cannot be bounded. For any $m \in \mathbb{Z}^+$ and any $\epsilon < 1/2$, we have proved that there is a promise problem $A^{eq}(m)$ that can be solved by a 2QCFA with one-sided error ϵ in a polynomial expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states, whereas the sizes of the corresponding DFA, 2DFA, 2NFA and polynomial expected running time 2PFA are at least $2m+2$, $\sqrt{\log m}$, $\sqrt{\log m}$ and $\sqrt[3]{(\log m)/b}$. For any $m \in \mathbb{Z}^+$ and any $\epsilon < 1/2$, we have also showed that there exists a 2QCFA recognizing the language $L^{twin}(m)$ with one-sided error ϵ in an exponential expected running time with a constant number of quantum states and $\mathbf{O}(\log \frac{1}{\epsilon})$ classical states, whereas the sizes of the corresponding DFA, 2DFA, 2NFA and polynomial expected running time 2PFA are at least 2^m, $\sqrt{m}$, $\sqrt{m}$ and $\sqrt[3]{m/b}$.

To conclude, we formulate some open problems:

1. Can the result related to a promise problem $A^{eq}(m)$ be improved to deal with a language?

2. In Theorem 31, we gave a bound on a polynomial expected running time 2PFA. What is the bound when the expected running time is exponential?

Acknowledgements

The authors are thankful to the anonymous referees and editor for their comments and suggestions that greatly help to improve the quality of the manuscript. We thank Xiangfu Zou for discussion of a proof in Section 4. This work is supported in part by the National Natural Science Foundation (Nos. 61272058, 61073054, 60873055, 61100001), the Natural Science Foundation of Guangdong Province of China (No. 10251027501000004), the Specialized Research Fund for the Doctoral Program of Higher Education of China (Nos. 20100171110042, 20100171120051), the Fundamental Research Funds for the Central Universities (Nos. 10lgzd12,11lgpy36), the China Postdoctoral Science Foundation project (Nos. 20090460808, 201003375), and the project of SQIG at IT, funded by FCT and EU FEDER projects QSec PTDC/EIA/67661/2006, AMDSC UTAustin/MAT/0057/2008, NoE Euro-NF, and IT Project QuantTel, FCT project PTDC/EEA-TEL/103402/2008 QuantPrivTel, FCT PEst-OE/EEI/LA0008/2011.

References

[1] A. Ambainis, M. Beaudry, M. Golovkins, A. Kikusts, M. Mercer, D. Thérien, Algebraic Results on Quantum Automata, Theory of Computing Systems **39** (2006) 165–188.

[2] A. Ambainis, R. Freivalds, One-way quantum finite automata: strengths, weaknesses and generalizations, in: Proceedings of the 39th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, Palo Alfo, California, USA, 1998, pp. 332–341.

[3] A. Ambainis, J. Watrous, Two-way finite automata with quantum and classical states, Theoretical Computer Science **287** (2002) 299–311.

[4] A. Ambainis, A. Yakaryilmaz, Superiority of exact quantum automata for promise problems, Information Processing Letters **112** (7) (2012) 289–291.

[5] A. Ambainis, N. Nahimovs, Improved constructions of quantum automata, Theoretical Computer Science **410** (2009) 1916–1922.

[6] A. Ambainis, A. Nayak, A. Ta-Shma, U. Vazirani, Dense quantum coding and quantum automata, Journal of the ACM **49** (4) (2002) 496–511.

[7] A. Bertoni, C. Mereghetti, B. Palano, Quantum Computing: 1-Way Quantum Automata, in: Proceedings of the 9th International Conference on Developments in Language Theory (DLT2003), Lecture Notes in Computer Science, Vol. 2710, Springer, Berlin, 2003, pp. 1–20.

[8] J.C. Birget, State-complexity of finite-state devices, state compressibility and incompressibility. Math. Systems Theory, **26** (1993) 237–269.

[9] A. Brodsky, N. Pippenger, Characterizations of 1-way quantum finite automata, SIAM Journal on Computing **31** (2002) 1456–1478.

[10] C. Dwork, L. Stockmeyer, A time-complexity gap for two-way probabilistic finite state automata, SIAM J. Comput. **19** (1990) 1011–1023.

[11] C. Dwork, L. Stockmeyer, Finite state verifiers I: The power of interaction, J. ACM **39** (4) (1992) 800–828.

[12] W. Feller, An Introduction to Probability Theory and its Applications, Vol. I, Wiley, New York, 1967.

[13] J. Gruska, Quantum Computing, McGraw-Hill, London, 1999.

[14] J. Gruska, Descriptional complexity issues in quantum computing. J. Automata, Languages Combin. **5** (3) (2000) 191–218.

[15] J.E. Hopcroft, J.D. Ullman, Introduction to Automata Theory, Languages, and Computation, Addision-Wesley, New York, 1979.

[16] A. Kondacs, J. Watrous, On the power of quantum finite state automata, in: Proceedings of the 38th IEEE Annual Symposium on Foundations of Computer Science, 1997, pp. 66–75.

[17] E. Kushilevitz, Communication Complexity, Advances in Computers, **44** (1997) 331–360.

[18] E. Kushilevitz, N. Nisan, Communication Complexity, Cambridge University Press, 1997.

[19] F. Le Gall, Exponential separation of quantum and classical online space complexity, in: Proceedings of SPAA'06, 2006, pp. 67–73.

[20] L.Z. Li, D.W. Qiu, Determination of equivalence between quantum sequential machines, Theoretical Computer Science **358** (2006) 65–74.

[21] L.Z. Li, D.W. Qiu, Determining the equivalence for one-way quantum finite automata, Theoretical Computer Science **403** (2008) 42–51.

[22] L.Z. Li, D.W. Qiu, X.F. Zou, L.J. Li, L.H. Wu, P. Mateus, Characterizations of one-way general quantum finite automata, Theoretical Computer Science **419** (2012) 73–91.

[23] C. Moore and J.P. Crutchfield, Quantum automata and quantum grammars, Theoretical Computer Science **237** (2000) 275–306.

[24] C. Mereghetti, B. Palano, Quantum finite automata with control language, RAIRO-Inf. Theor. Appl. **40** (2006) 315–332.

[25] M.A. Nielsen, I.L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, Cambridge, 2000.

[26] K. Paschen, Quantum finite automata using ancilla qubits, Technical Report, University of Karlsruhe, 2000.

[27] A. Paz, Introduction to Probabilistic Automata, Academic Press, New York, 1971.

[28] D.W. Qiu, Some Observations on Two-Way Finite Automata with Quantum and Classical States, ICIC 2008, LNCS 5226 (2008), pp. 1–8.

[29] D.W. Qiu, L.Z. Li, An overview of quantum computation models: quantum automata, Frontiers of Computer Science in China **2** (2) (2008) 193–207.

[30] D.W. Qiu, P. Mateus, and A. Sernadas, One-way quantum finite automata together with classical states, arXiv:0909.1428.

[31] D.W. Qiu, S. Yu, Hierarchy and equivalence of multi-letter quantum finite automata, Theoretical Computer Science **410** (2009) 3006–3017.

[32] D.W. Qiu, L.Z. Li, P. Mateus, J. Gruska, Quantum finite automata, in: Jiacun Wang (Eds.), Finite State Based Models and Applications, CRC Handbook, Boca Raton, FL, 2012, pp. 113–144.

[33] J.C. Shepherdson, The reduction of two-way automata to one-way automata, IBM J. Research and Development **3** (1959) 199–201.

[34] M.Y. Vardi, A note on the reduction of two-way automata to one-way automata, Inform. Process. Lett. **30** (5) (1989) 261–264.

[35] R. de Wolf, Quantum communication and complexity, Theoretical Computer Science **287** (2002) 337-353.

[36] J. Watrous, Quantum computational complexity, R.A. Meyers, Editor, Encyclopedia of Complexity and Systems Science, Springer, 2009, pp. 7174–7201.

[37] A.C. Yao, Some Complexity Questions Related to Distributed Computing, in: Proc. of 11th STOC, 1979, pp. 209-213.

[38] A. Yakaryilmaz, A.C.C. Say, Succinctness of two-way probabilistic and quantum finite automata, Discrete Mathematics and Theoretical Computer Science **12** (4) (2010) 19–40.

[39] A. Yakaryilmaz, A.C.C. Say, Languages recognized by nondeterministic quantum finite automata, Quantum Information and Computation **10** (9-10) (2010) 747–770.

[40] A. Yakaryilmaz, A.C.C. Say, Unbounded-error quantum computation with small space bounds, Information and Computation **209** (2011) 873–892.

[41] S. Yu, Regular Languages, In: G. Rozenberg, A. Salomaa (Eds.), Handbook of Formal Languages, Springer-Verlag, Berlin, 1998, pp. 41–110.

[42] S.G. Zheng, D.W. Qiu, L.Z. Li, Some languages recognized by two-way finite automata with quantum and classical states, International Journal of Foundations of Computer Science **23** (5) (2012) 1117–1129. Also arXiv:1112.2844 (2011).

[43] S.G. Zheng, D.W. Qiu, L.Z. Li, Jozef Gruska, One-way finite automata with quantum and classical states, In: H. Bordihn, M. Kutrib, and B. Truthe (Eds.), Dassow Festschrift 2012, LNCS 7300, 2012, pp. 273–290. Also arXiv:1112.2022 (2011).