Generalizations of the distributed Deutsch-Jozsa promise problem

Jozef Gruska[1], Daowen Qiu[2], Shenggen Zheng[1,*]

[1] *Faculty of Informatics, Masaryk University, Brno 60200, Czech Republic*

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

Abstract

In the *distributed Deutsch-Jozsa promise problem*, two parties are to determine whether their respective strings $x, y \in \{0,1\}^n$ are at the *Hamming distance* $H(x,y) = 0$ or $H(x,y) = \frac{n}{2}$. Buhrman et al. (STOC' 98) proved that the exact *quantum communication complexity* of this problem is $\mathbf{O}(\log n)$ while the *deterministic communication complexity* is $\mathbf{\Omega}(n)$. This was the first impressive (exponential) gap between quantum and classical communication complexity. In this paper, we generalize the above distributed Deutsch-Jozsa promise problem to determine, for any fixed $\frac{n}{2} \leq k \leq n$, whether $H(x,y) = 0$ or $H(x,y) = k$, and show that an exponential gap between exact quantum and deterministic communication complexity still holds if k is an even such that $\frac{1}{2}n \leq k < (1-\lambda)n$, where $0 < \lambda < \frac{1}{2}$ is given. We also deal with a promise version of the well-known *disjointness* problem and show also that for this promise problem there exists an exponential gap between quantum (and also probabilistic) communication complexity and deterministic communication complexity of the promise version of such a disjointness problem. Finally, some applications to quantum, probabilistic and deterministic finite automata of the results obtained are demonstrated.

Keywords: Quantum communication complexity, Deutsch-Jozsa promise problem, Query complexity, Finite automata

1. Introduction

Since the topic of communication complexity was introduced by Yao [34], it has been extensively studied [9, 13, 22, 26]. In the setting of two parties, Alice is given an $x \in \{0,1\}^n$, Bob is given a $y \in \{0,1\}^n$ and their task is to communicate in order to determine the value of some given Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, while exchanging as small number of bits as possible. In this setting, local computations of the parties are considered to be free, but communication is considered to be expensive and has to be minimized. Moreover, for computation, Alice and Bob have access to arbitrary computational power.

There are usually three types of communication complexities considered for the above communication task: deterministic, probabilistic or quantum.

Two of the most often studied communication problems are that of equality and disjointness [26], defined as follows:

- **Equality**: $\mathrm{EQ}(x,y) = 1$ if $x = y$ and 0 otherwise.

*Corresponding author.
E-mail addresses: zhengshenggen@gmail.com (S. Zheng), gruska@fi.muni.cz (J. Gruska), issqdw@mail.sysu.edu.cn (D. Qiu).

- **Disjointness**: $\mathrm{DISJ}(x,y) = 1$ if there is no index i such that $x_i = y_i = 1$ and 0 if such an index exists. Equivalently, this function can be defined also as $\mathrm{DISJ}(x,y) = 1$ if $\sum_{i=1}^{n} x_i \wedge y_i = 0$ and 0 if $\sum_{i=1}^{n} x_i \wedge y_i > 0$. (We can view x and y as being subsets of $\{1, \cdots, n\}$ represented by characteristic vectors and to have $\mathrm{DISJ}(x,y) = 1$ iff these two subsets are disjoint.)

Deterministic communication complexities of the above problems EQ and DISJ are both n [26].

Buhrman et al. [11, 13] proved that the exact quantum communication complexity of the distributed Deutsch-Jozsa promise problem, for $x, y \in \{0,1\}^n$ and n is even, that is for

$$\mathrm{EQ}'(x,y) = \begin{cases} 1 & \text{if } H(x,y) = 0 \\ 0 & \text{if } H(x,y) = \frac{n}{2}, \end{cases} \tag{1}$$

is $\mathbf{O}(\log n)$. This was the first impressively large (exponential) gap between quantum and classical communication complexity[1].

It has been so far a folklore belief that the promise $H(x,y) = \frac{n}{2}$ is essential for the above result. However, we prove that the result holds also for the following generalizations of this promise problem

$$\mathrm{EQ}_k(x,y) = \begin{cases} 1 & \text{if } H(x,y) = 0 \\ 0 & \text{if } H(x,y) = k, \end{cases} \tag{2}$$

for any fixed $k \geq \frac{n}{2}$. That is the exact quantum communication complexity of EQ_k is $\mathbf{O}(\log n)$ while the classical deterministic communication complexity is $\mathbf{\Omega}(n)$ if k is an even such that $\frac{1}{2}n \leq k < (1-\lambda)n$, where $0 < \lambda < \frac{1}{2}$ is given. Our proof has been inspired by methods used in [5].

Let us consider also the following problem. Namely, an analogue of the Deutsch-Jozsa promise problem:

$$\mathrm{DJ}_k(x) = \begin{cases} 1 & \text{if } W(x) = 0 \\ 0 & \text{if } W(x) > k, \end{cases} \tag{3}$$

where $k \geq \frac{n}{2}$ is fixed and $W(x)$ is the Hamming weight of x. We prove that the exact quantum query complexity of DJ_k is 1 while the deterministic query complexity is $n - k + 1$.

If errors can be tolerated, both quantum and probabilistic communication complexities of the equality problem are $\mathbf{O}(\log n)$.

Concerning disjointness problem, the probabilistic communication complexity is $\mathbf{\Omega}(n)$ [8, 24, 32] even if errors are tolerated. In the quantum cases, Buhrman et al. [11] proved that quantum communication complexity of DISJ is $\mathbf{O}(\sqrt{n}\log n)$. This bound has been improved to $\mathbf{O}(\sqrt{n})$ by Aaronson and Ambainis [3]. Finally, Razborov showed that any bounded-error quantum protocol for DISJ needs to communicate about $\sqrt{n}$ qubits [33]. Situation is different from the EQ problem, for which there is an exponential gap between quantum (and also probabilistic) communication complexity and deterministic communication complexity as shown in [11, 13, 26]. All known gaps for DISJ are not larger than quadratic. It is therefore of interest to find out whether there are some promise versions of the disjointness problem for which bigger communication complexity gaps can be obtained. We give a positive answer to such a question. In order to do that, we consider the following set of promise problems where $0 < \lambda \leq \frac{1}{4}$

$$\mathrm{DISJ}_\lambda(x,y) = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} x_i \wedge y_i = 0 \\ 0 & \text{if } \lambda n \leq \sum_{i=1}^{n} x_i \wedge y_i \leq (1-\lambda)n. \end{cases} \tag{4}$$

[1] In fact, both n and $\frac{n}{2}$ must be even in order to obtain an exponential quantum speed-up. We will justify this claim in the Remark 1 in Section 3.

We prove that quantum communication complexity of DISJ_λ is not more than $\frac{\log 3}{3\lambda}(3+2\log n)$ while the deterministic communication complexity is $\mathbf{\Omega}(n)$. For example, if $\lambda = \frac{1}{4}$, then the quantum communication complexity of DISJ_λ is not more than $3 + 2\log n$ while the deterministic communication complexity is more than 0.007n. We prove also that probabilistic communication complexity of DISJ_λ is not more than $\frac{\log 3}{\lambda}\log n$. Therefore, there is an exponential gap between quantum (and also probabilistic) communication complexity and deterministic communication complexity of the above promise problem.

Number of states is a natural complexity measure for all models of finite automata and state complexity of finite automata is one of the research fields with many applications [36]. There is a variety of methods how to prove lower bounds on the state complexity and methods as well as the results of communication complexity are among the main ones [23, 25, 26]. In this paper we also show how to make use of our new communication complexity results to get new state complexity bounds.

The paper is structured as follows. In Section 2 basic needed concepts and notations are introduced and models involved are described in details. Communication complexities and query complexities of the promise problems EQ_k and DJ_k are investigated in Section 3. Communication complexity of the promise problem DISJ_λ is dealt with in Section 4. Applications to finite automata are explored in Section 5. Some open problems are discussed in Section 6.

2. Preliminaries

In this section, we recall some basic definitions about communication complexity, query complexity and quantum finite automata. Concerning basic concepts and notations of quantum information processing, we refer the reader to [18, 29].

2.1. Communication complexity

We recall here only very basic concepts and notations of *communication complexity*, and we refer the reader to [13, 26] for more details. We will deal with the situation that there are two communicating parties and with very simple tasks of computing two-argument Boolean functions for the case 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 parties and focus solely on the amount of communication that is need to be exchanged between both parties in order to compute the value of a given Boolean function.

More technically, let $X = Y = \{0,1\}^n$. We will consider two-argument functions $f : X \times Y \to \{0,1\}$ and two communicating parties. Alice will be given an $x \in X$ and Bob a $y \in Y$. They want to compute $f(x, y)$. If f is defined only on a proper subset of $X \times Y$, f is said to be a partial function or a promise problem.

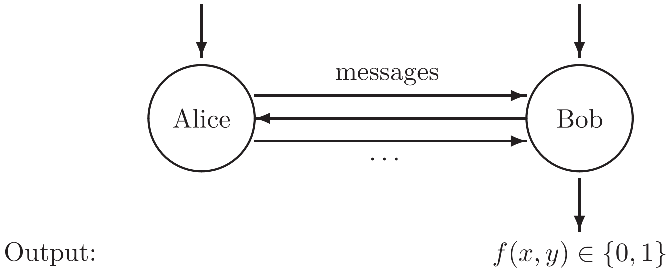

Figure 1: Communication protocol

The computation of $f(x, y)$ will be done using a communication protocol, presented in Figure 1. During the execution of the protocol, parties alternate roles in sending messages. Each of these messages will be a bit-string. The protocol, whose steps are based on the communication so far, also specifies for each step whether the communication terminates (in which case it also specifies what is the output). If the communication does not terminate, the protocol also specifies what kind of message the sender (Alice or Bob) should send next as a function of its input and communication so far.

A deterministic communication protocol $\mathcal{P}$ computes a (partial) function f, if for every (promise) input pair $(x, y) \in X \times Y$ the protocol terminates with the value $f(x, y)$ as its output. In a probabilistic protocol, Alice and Bob may also flip coins during the protocol execution and proceed according to their outputs and the protocol can also have an erroneous output with a small probability. In a quantum protocol, Alice and Bob may use also quantum resources for communication.

Let $\mathcal{P}(x, y)$ denote the output of the protocol $\mathcal{P}$. We will consider two kinds of protocols for computing a function f:

- An exact protocol, that always outputs the correct answer (that is $Pr(\mathcal{P}(x, y) = f(x, y)) = 1$).
- A two-sided error (bounded error) protocol $\mathcal{P}$ such that $Pr(\mathcal{P}(x, y) = f(x, y)) \geq \frac{2}{3}$.

The communication complexity of a protocol $\mathcal{P}$ is the worst case number of (qu)bits exchanged. The communication complexity of f is, with which respect to the communication mode used, the complexity of an optimal protocol for f.

We will use $D(f)$ and $R(f)$ to denote the deterministic communication complexity and the two-sided error probabilistic communication complexity of a function f, respectively. Similarly we use notations $Q_E(f)$ and $Q(f)$ for the exact and two-sided error quantum communication complexity of a function f.

Let us also summarize already known communication complexity results concerning communication problems EQ, DISJ and EQ$'$:

1. $D(\mathrm{EQ}) = n$, $D(\mathrm{DISJ}) = n$ [26], $D(\mathrm{EQ}') \in \mathbf{\Omega}(n)$ [11].
2. $Q_E(\mathrm{EQ}') \in \mathbf{O}(\log n)$ [11].
3. $R(\mathrm{EQ}) \in \mathbf{O}(\log n)$ [26], $R(\mathrm{DISJ}) \in \mathbf{\Omega}(n)$ [8, 24, 32].
4. $Q(\mathrm{DISJ}) \in \mathbf{\Theta}(\sqrt{n})$ [3, 33].

2.2. Exact query complexity

The exact quantum query complexity for partial functions was dealt with also in [10, 15] and for total functions in [5–7, 28].

In the next we recall definitions of two exact query complexity models. For more concerning basic concepts and notations related to query complexity, we refer the reader to [14].

Exact classical (deterministic) query algorithms to compute a Boolean function $f : \{0, 1\}^n \to \{0, 1\}$ can be described using decision trees, in the following way:

Let the input string be $x = x_1 x_2 \ldots x_n$. A decision tree T_f for x is a rooted binary tree in which each internal vertex has exactly two children. Moreover, each internal vertex is labeled with a variable x_i $(1 \leq i \leq n)$ and each leaf is labeled with a value 0 or 1. T_f should be designed in such a way that it can be used to compute function f in the following way: Let us start at the root. If this is a leaf then stop and the value of f is that assigned to that leaf. Otherwise, query the value of the variable x_i that labels the root. If $x_i = 0$, then evaluate recursively the left subtree, if $x_i = 1$ then the right subtree. The output of the tree is then the value of the leaf that is reached eventually. The depth of T_f is the maximal length of any path from the root to any leaf (i.e. the worst-case number of queries used for all inputs). The minimal depth over all decision trees computing f is the exact classical query complexity (deterministic query complexity, decision tree complexity) $DT(f)$ of f.

Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function and $x = x_1x_2\cdots x_n$ be an input bit string. Each exact quantum query algorithm for f works in a Hilbert space with some fixed basis, called standard. Each of the basis states corresponds to either one or none of the input bits. It starts in a fixed starting state, then performs on it a sequence of transformations U_1, Q, U_2, Q, $\ldots$, U_t, Q, U_{t+1}. Unitary transformations U_i do not depend on the input bits, while Q, called the *query transformation*, does, in the following way. If a basis state $|\psi\rangle$ corresponds to the i-th input bit, then $Q|\psi\rangle = (-1)^{x_i}|\psi\rangle$. If it does not correspond to any input bit, then Q leaves it unchanged: $Q|\psi\rangle = |\psi\rangle$. Finally, the algorithm performs a measurement in the standard basis. Depending on the result of the measurement, the algorithm outputs either 0 or 1 which must be equal to $f(x)$. The *exact quantum query complexity* $QT_E(f)$ is the minimum number of queries used by any quantum algorithm which computes $f(x)$ exactly for all x.

2.3. Lower bound methods for deterministic communication complexity

There are quite a few of lower bound methods to determine deterministic communication complexity. We just recall so called "rectangles" method in this subsection. Concerning more on lower bound methods, see [13, 22, 26].

A *rectangle* in $X \times Y$ is a subset $R \subseteq X \times Y$ such that $R = A \times B$ for some $A \subseteq X$ and $B \subseteq Y$. A rectangle $R = A \times B$ is called 1(0)-rectangle of a function $f : X \times Y \to \{0,1\}$ if for every $(x,y) \in A \times B$ the value of $f(x,y)$ is 1(0). For a partial function $f : X \times Y \to \{0,1\}$ with domain $\mathcal{D}$, a rectangle $R = A \times B$ is called 1(0)-rectangle if the value of $f(x,y)$ is 1(0) for every $(x,y) \in \mathcal{D} \cap (A \times B)$ – we do not care about values for $(x,y) \notin \mathcal{D}$. Moreover, $C^i(f)$ is defined as the minimum number of i-rectangles that partition the space of i-inputs (such inputs x and y that $f(x,y) = i$) of f.

We now recall a lemma on "rectangles" method from [26]:

Lemma 1. *For every (partial) function f,* $D(f) \geq \max\{\log C^1(f), \log C^0(f)\}$.

2.4. Measure-once one-way finite automata with quantum and classical states

In this subsection we recall the definition of 1QCFA. Concerning more on classical and quantum automata see [18, 19, 21, 30].

Two-way finite automata with quantum and classical states (2QCFA) were introduced by Ambainis and Watrous [2] and explored also by Yakaryılmaz, Zheng and others [27, 35, 38–40]. Informally, a 2QCFA can be seen as a *two-way deterministic finite automaton* (2DFA) with an access to a quantum memory for states of a fixed Hilbert space upon which at each step either a unitary operation is performed or a projective measurement and the outcomes of which then probabilistically determine the next move of the underlying 2DFA. 1QCFA are one-way versions of 2QCFA [37]. In this paper, we only use 1QCFA in which a unitary transformation is applied in every step after scanning a symbol and a measurement is performed at the end of the computation. Such model is called a measure-once 1QCFA (MO-1QCFA) and corresponds to a variant of *measure-once quantum finite automata*, which can also be seen as a special case of *one-way quantum finite automata together with classical states* defined in [31].

Definition 1. An MO-1QCFA $\mathcal{A}$ is specified by a 8-tuple

$$\mathcal{A} = (Q, S, \Sigma, \Theta, \delta, |q_0\rangle, s_0, Q_a), \tag{5}$$

where

1. Q is a finite set of orthonormal quantum (basis) states;
2. S is a finite set of classical states;
3. Σ is a finite alphabet of input symbols and let $\Sigma' = \Sigma \cup \{¢, \$\}$, where symbol ¢ will be used as the left end-marker and symbol \$ as the right end-marker;
4. $|q_0\rangle \in Q$ is the initial quantum state;
5. s_0 is the initial classical state;
6. $Q_a \subseteq Q$ denotes the set of accepting quantum basis states;
7. Θ is a quantum transition function
$$\Theta : S \times \Sigma' \to U(\mathcal{H}(Q)), \tag{6}$$
where $U(\mathcal{H}(Q))$ is the set of unitary operations on the Hilbert space generated by quantum states from Q;
8. δ is a classical transition function
$$\delta : S \times \Sigma' \to S, \tag{7}$$
such that $\delta(s, \sigma) = s'$, then the new classical state of the automaton is s'.

The computation of an MO-1QCFA $\mathcal{A} = (Q, S, \Sigma, \Theta, \delta, |q_0\rangle, s_0, Q_a)$ on an input $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ starts with the string $¢w\$$ on the input tape. At the start, the tape head of the automaton is positioned on the left end-marker and the automaton begins the computation in the initial classical state and in the initial quantum state. After that, in each step, if the classical state of the automaton is s, its tape head reads a symbol σ and its quantum state is $|\psi\rangle$, then the automaton changes its quantum state to $\Theta(s, \sigma)|\psi\rangle$ and its classical state to $\delta(s, \sigma)$. At the end of the computation, the projective measurement $\{P_a, P_r\}$ is applied on the current quantum state, where $P_a = \sum_{|i\rangle \in Q_a} |i\rangle\langle i|$ and $P_r = I - P_a$. If the classical outcome is a (r), then the input is accepted (rejected).

For any state s, any string $w \in (\Sigma')^*$ and any $\sigma \in \Sigma$, let $\widehat{\delta}(s, \sigma w) = \widehat{\delta}(\delta(s, \sigma), w)$; if $|w| = 0$, $\widehat{\delta}(s, w) = s$. Let $\sigma_0 = ¢$ and $\sigma_{n+1} = \$$. The probability that the automaton $\mathcal{A}$ accepts the input w is

$$Pr[\mathcal{A} \text{ accepts } w] = \|P_a \Theta(s_{n+1}, \sigma_{n+1}) \cdots \Theta(s_1, \sigma_1)\Theta(s_0, \sigma_0)|q_0\rangle\|^2, \tag{8}$$

where $s_{i+1} = \widehat{\delta}(s_0, \sigma_0 \cdots \sigma_i)$. The probability that $\mathcal{A}$ rejects the input w is $Pr[\mathcal{A} \text{ rejects } w] = 1 - Pr[\mathcal{A} \text{ accepts } w]$.

The language acceptance is a special case of so called promise problem solving. A *promise problem* [17] over an alphabet Σ is a pair $A = (A_{yes}, A_{no})$, where A_{yes}, $A_{no} \subset \Sigma^*$ are disjoint sets. Languages over an alphabet Σ may be viewed as promise problems that obey the additional constraint $A_{yes} \cup A_{no} = \Sigma^*$.

A promise problem $A = (A_{yes}, A_{no})$ is solved exactly by a finite automaton $\mathcal{A}$ if

- $\forall w \in A_{yes}$, $Pr[\mathcal{A} \text{ accepts } w] = 1$, and
- $\forall w \in A_{no}$, $Pr[\mathcal{A} \text{ rejects } w] = 1$.

On the other side, a finite automaton $\mathcal{A}$ is said to solve a promise problem $A = (A_{yes}, A_{no})$ with a one-sided error ε ($0 < \varepsilon \leq \frac{1}{2}$) if

- $\forall w \in A_{yes}$, $Pr[\mathcal{A}$ accepts $w] = 1$, and
- $\forall w \in A_{no}$, $Pr[\mathcal{A}$ rejects $w] \geq 1 - \varepsilon$.

3. Generalizations of the distributed Deutsch-Jozsa promise problem

We will explore communication complexity of several generalizations of the distributed Deutsch-Jozsa promise problem.

Theorem 1. *$Q_E(EQ_k) \in \mathbf{O}(\log n)$ for any fixed $k \geq \frac{n}{2}$.*

Proof. Assume that Alice is given an input $x = x_1 \cdots x_n$ and Bob an input $y = y_1 \cdots y_n$. The following quantum communication protocol $\mathcal{P}$ computes $\mathrm{EQ}_k(x, y)$ using $n+1$ quantum basis states $|0\rangle, |1\rangle, \ldots, |n\rangle$ as follows:

1. Alice begins with the initial quantum state $|0\rangle$ and performs on it the unitary map U_k such that $U_k|0\rangle = \sqrt{\frac{2k-n}{2k}}|0\rangle + \sqrt{\frac{n}{2k}}|1\rangle$, where
$$U_k = \begin{pmatrix} \sqrt{\frac{2k-n}{2k}} & -\sqrt{\frac{n}{2k}} & \mathbf{0} \\ \sqrt{\frac{n}{2k}} & \sqrt{\frac{2k-n}{2k}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{n-1,n-1} \end{pmatrix}. \tag{9}$$
2. Alice then performs the unitary map U_h on her quantum state such that $U_h|0\rangle = |0\rangle$ and $U_h|1\rangle = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}|i\rangle$, i.e. the first column of U_h is $(1, 0, \ldots, 0)^T$, the second column of U_h is $(0, \frac{1}{\sqrt{n}}, \ldots, \frac{1}{\sqrt{n}})^T$, and the other entries are arbitrary, but such that the resulting matrix is unitary what is clearly always possible.
3. Alice then applies to the current state the unitary map U_x such that $U_x|0\rangle = |0\rangle$ and $U_x|i\rangle = (-1)^{x_i}|i\rangle$ for $i > 0$.
4. Afterwards Alice sends her current quantum state $|\psi_4\rangle = U_x U_h U_k|0\rangle = \sqrt{\frac{2k-n}{2k}}|0\rangle + \sqrt{\frac{n}{2k}}\sqrt{\frac{1}{n}}\sum_{i=1}^{n}(-1)^{x_i}|i\rangle$ to Bob.
5. Bob then applies to the state that he has received the unitary map U_y such that $U_y|0\rangle = |0\rangle$ and $U_y|i\rangle = (-1)^{y_i}|i\rangle$ for $i > 0$.
6. Bob applies the unitary map $U_k^{-1}U_h^{-1}$ to his quantum state.
7. Afterwards Bob measures the resulting state in the standard basis and outputs 1 if the measurement outcome is $|0\rangle$ and outputs 0 otherwise.

The state after the step 5 will be

$$|\psi_5\rangle = U_y U_x U_h U_k|0\rangle = \sqrt{\frac{2k-n}{2k}}|0\rangle + \sqrt{\frac{n}{2k}}\sqrt{\frac{1}{n}}\sum_{i=1}^{n}(-1)^{x_i+y_i}|i\rangle. \tag{10}$$

Therefore, if $x = y$, then the state after the step 6 will be

$$|\psi_6\rangle = U_k^{-1}U_h^{-1}U_y U_x U_h U_k|0\rangle = U_k^{-1}U_h^{-1}U_h U_k|0\rangle = |0\rangle. \tag{11}$$

If $x \neq y$, then $H(x,y) = k$ and the state after the step 6 is

$$|\psi_6\rangle = U_k^{-1}U_h^{-1}U_yU_xU_hU_k|0\rangle = U_k^{-1}U_h^{-1}\left(\sqrt{\frac{2k-n}{2k}}|0\rangle + \sqrt{\frac{n}{2k}}\sqrt{\frac{1}{n}}\sum_{i=1}^{n}(-1)^{x_i+y_i}|i\rangle\right) \tag{12}$$

$$= U_k^{-1}\left(\sqrt{\frac{2k-n}{2k}}|0\rangle + \sqrt{\frac{n}{2k}}\frac{1}{n}\sum_{i=1}^{n}(-1)^{x_i+y_i}|1\rangle + \sum_{i=2}^{n}\alpha_i|i\rangle\right) \tag{13}$$

$$= U_k^{-1}\left(\sqrt{\frac{2k-n}{2k}}|0\rangle + \sqrt{\frac{n}{2k}}\frac{n-2k}{n}|1\rangle + \sum_{i=2}^{n}\alpha_i|i\rangle\right) \tag{14}$$

$$= \left(\sqrt{\frac{2k-n}{2k}}\sqrt{\frac{2k-n}{2k}} + \sqrt{\frac{n}{2k}}\sqrt{\frac{n}{2k}}\frac{n-2k}{n}\right)|0\rangle + \sum_{i=1}^{n}\beta_i|i\rangle \tag{15}$$

$$= \sum_{i=1}^{n}\beta_i|i\rangle, \tag{16}$$

where α_i, β_i are amplitudes that we do not need to be specified more exactly.

Because the amplitude of $|0\rangle$ is 0, we can get the exact result after the measurement in the step 7.

It is clear that this protocol communicates only $\lceil\log(n+1)\rceil$ qubits. $\square$

Obviously, $D(\mathrm{EQ}_k) \leq n-k+1$. For the case that $k = \frac{n}{2}$ and k is even, $\mathrm{EQ}_k = \mathrm{EQ}'$ and $D(\mathrm{EQ}_k) \in \mathbf{\Omega}(n)$ [11, 13]. For the cases that $\frac{1}{2}n \leq k < (1-\lambda)n$, where $0 < \lambda < \frac{1}{2}$ is given, we can prove, using a similar proof method as in [11, 13], the following theorem:

Theorem 2. *Suppose $0 < \lambda < \frac{1}{2}$ is given and k is an even. Then $D(EQ_k) \in \mathbf{\Omega}(n)$ for all k such that $\frac{1}{2}n \leq k < (1-\lambda)n$.*

Proof. In order to prove the theorem, we introduce a lemma (Theorem 1 in [16]) first.

For $x, y \in \{0,1\}^n$, let us denote $|x \wedge y| = \sum_{i=1}^{n} x_i \wedge y_i$. Let also $M(n,l)$ denote the maximum of the sets cardinality $|F|$, where $F \subset \{0,1\}^n$ subject to the constraint: $|x \wedge y| \neq l$ holds for all distinct $x, y \in F$.

Lemma 2. *[16] If $0 < \eta < \frac{1}{4}$ is given, then there exists a positive constant $\varepsilon_0 = \varepsilon_0(\eta)$ such that $M(n,l) \leq (2-\varepsilon_0)^n$ for all l such that $\eta n < l < (\frac{1}{2}-\eta)n$.*

Let $\mathcal{P}$ be a deterministic protocol for EQ_k. Let us consider the set $E = \{(x,x) \,|\, W(x) = \lfloor\frac{n}{2}\rfloor\}$. For every $(x,x) \in E$, we have $\mathcal{P}(x,x) = 1$. Suppose now that there is a 1-monochromatic rectangle $R = A \times B \subseteq \{0,1\}^n \times \{0,1\}^n$ such that $\mathcal{P}(x,y) = 1$ for every promise pair $(x,y) \in R$. Let $S = R \cap E$. We now prove that for any distinct $(x,x),(y,y) \in S$, $|x \wedge y| \neq \lfloor\frac{n-k}{2}\rfloor$.

If $|x \wedge y| = \lfloor\frac{n-k}{2}\rfloor$, then $H(x,y) = 2(\lfloor\frac{n}{2}\rfloor - \lfloor\frac{n-k}{2}\rfloor) = k$ and $\mathcal{P}(x,y) = 0$. Since $(x,x) \in R$ and $(y,y) \in R$, we have $(x,y) \in R$ and $\mathcal{P}(x,y) = 0$, which is a contradiction.

Because of the assumption, we have $\frac{\lambda}{2}n < \lfloor\frac{n-k}{2}\rfloor \leq \frac{1}{4}n < (\frac{1}{2}-\frac{\lambda}{2})n$. Let $\eta = \frac{\lambda}{2}$. According to Lemma 2, there exists a constant ε_0 such that $|S| \leq (2-\varepsilon_0)^n$.

Let us now continue the proof of Theorem 2 . The minimum number of 1-monochromatic rectangles that partition the space of inputs is

$$C^1(\mathrm{EQ}_k) \geq \frac{|E|}{|S|} \geq \frac{\binom{n}{\lfloor n/2\rfloor}}{(2-\varepsilon_0)^n} > \frac{2^n/n}{(2-\varepsilon_0)^n}. \tag{17}$$

According to Lemma 1, the deterministic communication complexity of the problem EQ_k then holds:

$$D(\mathrm{EQ}_k) \geq \log C^1(\mathrm{EQ}_k) > \log \frac{2^n/n}{(2-\varepsilon_0)^n} = n - \log n - n\log(2-\varepsilon_0). \tag{18}$$

Since $1-u \leq e^{-u} \leq 2^{-u}$, for any real number $u > 0$, we have $\log(2-\varepsilon_0) = 1 + \log(1-\varepsilon_0/2) < 1 - \varepsilon_0/2$. Therefore

$$D(\mathrm{EQ}_k) \geq n - \log n - n(1-\frac{\varepsilon_0}{2}) = \frac{\varepsilon_0}{2}n - \log n. \tag{19}$$

Thus, $D(\mathrm{EQ}_k) \in \mathbf{\Omega}(n)$. □

Remark 1. If k is odd, we can prove that $D(\mathrm{EQ}_k) \in \mathbf{O}(1)$ as follows:

1. Alice calculates $W(x)$ and then sends one bit information of $W(x)$'s parity to Bob (for example, Alice sends "1" if $W(x)$ is even and "0" otherwise).
2. After receiving Alice's information, Bob calculates $W(y)$. If the parities of $W(y)$ and $W(x)$ are the same, then $\mathrm{EQ}_k(x,y) = 1$; otherwise, $\mathrm{EQ}_k(x,y) = 0$.

The above protocol computes EQ_k since if $H(x,y) = 0$, $W(x)+W(y)$ must be even; if $H(x,y) = k$, then the parity of $W(x)+W(y)$ must be the same as the parity of k.

We can now explore also the exact quantum query complexity of DJ_k.

Theorem 3. *The exact quantum query complexity $QT_E(DJ_k) = 1$ for any fixed $k \geq \frac{n}{2}$.*

Proof. Let us consider a query algorithm $\mathcal{A}$ that will solve the promise problem DJ_k using $n+1$ quantum basis states $|0\rangle, |1\rangle, \ldots, |n\rangle$ and works as follows: (where the unitary transformations U_k and U_h are the same ones as in the proof of the Theorem 1.)

1. $\mathcal{A}$ begins in the state $|0\rangle$ and performs on it the unitary transformation $U_1 = U_h U_k$.
2. $\mathcal{A}$ performs a query Q.
3. $\mathcal{A}$ performs the unitary transformation $U_2 = U_k^{-1}U_h^{-1}$.
4. $\mathcal{A}$ measures the resulting state in the standard basis and outputs 1 if the measurement outcome is $|0\rangle$ and outputs 0 otherwise.

The rest of the proof is similar to that of Theorem 1. □

Obviously, the exact classical query complexity of DJ_k is $n-k+1$.

4. Communication complexity of a promise version of the disjointness problem

It may seem that if we consider DISJ'_k as a similar promise version to the problem DISJ as we did with EQ_k, we get a similar result.

However, the reality is a bit different. Indeed, let us denote

$$\mathrm{DISJ}'_k(x,y) = \begin{cases} 1 & \text{if } \sum_{i=1}^n x_i \wedge y_i = 0 \\ 0 & \text{if } \sum_{i=1}^n x_i \wedge y_i = k, \end{cases} \tag{20}$$

where $k \geq \frac{n}{2}$ is fixed. Using an analogous proof method as in Section 3, we can prove that $Q_E(\mathrm{DISJ}'_k) \in \mathbf{O}(\log n)$. But, when comparing to the deterministic communication complexity, this is no improvement at all. Actually, we can prove that for $k > \frac{n}{2}$, $D(\mathrm{DISJ}'_k) \in \mathbf{O}(1)$. Indeed, let us consider the following protocol:

1. Alice calculates $W(x)$. If $W(x) < k$, Alice sends 1 as the outcome of $\mathrm{DISJ}'_k(x,y)$ to Bob; otherwise, she sends 0 to Bob.
2. After receiving Alice's information, if Bob did not get 1 as the result of $\mathrm{DISJ}'_k(x,y)$ from Alice, he then calculates $W(y)$. If $W(y) < k$, then Bob outputs 1 as the result of $\mathrm{DISJ}'_k(x,y)$; otherwise, $\mathrm{DISJ}'_k(x,y) = 0$.

For the case $k = \frac{n}{2}$, we can prove that $D(\mathrm{DISJ}'_k) \in \mathbf{O}(1)$ using the following protocol:

1. Alice calculates $W(x)$. If $W(x) < \frac{n}{2}$, then Alice sends 1 as the outcome of $\mathrm{DISJ}'_k(x,y)$ to Bob; if $W(x) = \frac{n}{2}$, Alice sends 0 and x_1 to Bob; otherwise, she sends 0 to Bob.
2. After receiving Alice's information, if Bob did not get 1 as the result of $\mathrm{DISJ}'_k(x,y)$ from Alice, he then calculates $W(y)$. If $W(y) < \frac{n}{2}$, then Bob outputs the result 1 as the of $\mathrm{DISJ}'_k(x,y)$. If $W(y) = \frac{n}{2} = W(x)$, Bob compares y_1 with x_1 and then outputs the result $\mathrm{DISJ}'_k(x,y) = 0$ if $y_1 = x_1$ and $\mathrm{DISJ}'_k(x,y) = 1$ if $y_1 \neq x_1$. Otherwise, $\mathrm{DISJ}'_k(x,y) = 0$.

Obviously, the above protocol computes $\mathrm{DISJ}'_k(x,y)$ and uses for communication only $\mathbf{O}(1)$ bits.

4.1. Quantum protocol

Let us now explore how much of advantages can be obtained when quantum resources can be used for dealing with such communication problems as DISJ_λ. We give at first a quantum communication protocol for $\mathrm{DISJ}_{\frac{1}{4}}(x,y)$. From this protocol we can get the following result.

Theorem 4. *$Q(DISJ_{\frac{1}{4}}) \leq 3 + 2\log n$.*

Proof. Assume that Alice is given an input $x = x_1 \cdots x_n$ and Bob an input $y = y_1 \cdots y_n$. The quantum communication protocol $\mathcal{P}$ which computes $\mathrm{DISJ}_{\frac{1}{4}}$ using $2n$ quantum basis states $\{|i,j\rangle : 1 \leq i \leq n, 0 \leq j \leq 1\}$ (the basis state $|i,j\rangle$ is a $2n$-dimensional column vector with the $(nj+i)$-th entry being 1 and others being 0's.) will work as follows:

1. Alice starts with the quantum state $|\psi_0\rangle = |1,0\rangle = (1, \overbrace{0, \cdots, 0}^{2n-1})^T$ and applies to it the following unitary transformation U_s:
$$U_s|\psi_0\rangle = \sum_{i=1}^{n} \frac{1}{\sqrt{n}}|i,0\rangle = \frac{1}{\sqrt{n}}(\overbrace{1, \cdots, 1}^{n}, \overbrace{0, \cdots, 0}^{n})^T. \tag{21}$$
Alice then applies the following unitary transformation U_x when $x = x_1 \cdots x_n$ is the input word:
$$U_x = U_{x_n} \cdots U_{x_1} \tag{22}$$
where
$$U_{x_i} = \begin{cases} I, & \text{if } x_i = 0 \\ |i,1\rangle\langle i,0| + |i,0\rangle\langle i,1| + \sum_{j\neq i} |j,0\rangle\langle j,0| + \sum_{j \neq i} |j,1\rangle\langle j,1|, & \text{if } x_i = 1. \end{cases} \tag{23}$$
U_x is therefore a unitary transformation that exchanges the amplitudes of $|i,0\rangle$ and $|i,1\rangle$ if $x_i = 1$. The resulting quantum state, after performing U_x, will be
$$|\psi_1\rangle = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} ((1-x_i)|i,0\rangle + x_i|i,1\rangle) = \frac{1}{\sqrt{n}}(\overline{x}_1, \cdots, \overline{x}_n, x_1, \cdots, x_n)^T, \tag{24}$$
where $\overline{x}_i = 1 - x_i$.
Alice then sends the resulting quantum state $|\psi_1\rangle$ to Bob.

2. Bob applies to the state received the unitary mapping V_y, defined for each y as follows

$$V_y|i,0\rangle = |i,0\rangle \tag{25}$$

and

$$V_y|i,1\rangle = (-1)^{y_i}|i,1\rangle. \tag{26}$$

The quantum state after applying V_y will therefore be

$$|\psi_2\rangle = \frac{1}{\sqrt{n}}(\overline{x}_1, \cdots, \overline{x}_n, (-1)^{y_1}x_1, \cdots, (-1)^{y_n}x_n)^T. \tag{27}$$

If $x_i = y_i = 1$, then $(-1)^{y_i}x_i = -1 = (-1)^{x_i \wedge y_i}$; if $x_i = 1$ and $y_i = 0$, then $(-1)^{y_i}x_i = 1 = (-1)^{x_i \wedge y_i}$; otherwise $(-1)^{y_i}x_i = 0$.

Bob then sends his quantum state $|\psi_2\rangle$ to Alice.

3. Alice applies the unitary transformation U_x to the state $|\psi_2\rangle$ received from Bob and gets a new quantum state:

$$|\psi_3\rangle = \frac{1}{\sqrt{n}}(z_1, \cdots, z_n, \overbrace{0, \cdots, 0}^{n})^T. \tag{28}$$

If $x_i = 0$, then $z_i = \overline{x}_i = 1 = (-1)^{x_i \wedge y_i}$. If $x_i = 1$, then $z_i = (-1)^{y_i}x_i = (-1)^{x_i \wedge y_i}$. Therefore, $z_i = (-1)^{x_i \wedge y_i}$ for $1 \le i \le n$.

Alice then applies the unitary transformation U_f (to be specified later) to get the following state:

$$U_f|\psi_3\rangle = \left(\frac{1}{n}\sum_{i=1}^{n}(-1)^{x_i \wedge y_i}, \overbrace{*, \cdots, *}^{2n-1}\right)^T. \tag{29}$$

and then she measures the resulting quantum state with the observable $\{|i,0\rangle\langle i,0|, |i,1\rangle\langle i,1|\}_{i=1}^{n}$. If the measurement outcome is $|1,0\rangle$, Alice sends 1 otherwise 0 to Bob.

It is clear that this protocol uses for communication $1 + 2(\log 2n) = 3 + 2\log n$ qubits. Unitary transformations U_s and U_f do exist. The first column of U_s is $\frac{1}{\sqrt{n}}(\overbrace{1, \cdots, 1}^{n}, \overbrace{0, \cdots, 0}^{n})^T$ and the first row of U_f is $\frac{1}{\sqrt{n}}(\overbrace{1, \cdots, 1}^{n}, \overbrace{0, \cdots, 0}^{n})$. It is easy to verify that V_y's are unitary transformations.

If $\sum_{i=1}^{n} x_i \wedge y_i = 0$, then $\frac{1}{n}\sum_{i=1}^{n}(-1)^{x_i \wedge y_i} = 1$. After the measurement, Alice gets the quantum outcome $|1,0\rangle$ and sends 1 to Bob. Thus,

$$Pr(\mathcal{P}(x,y) = \mathrm{DISJ}_{\frac{1}{4}}(x,y)) = 1. \tag{30}$$

If $n/4 \le \sum_{i=1}^{n} x_i \wedge y_i \le 3n/4$, then $|\frac{1}{n}\sum_{i=1}^{n}(-1)^{x_i \wedge y_i}| \le 1/2$ and Alice gets as the quantum outcome $|1,0\rangle$ with the probability not more than $|\frac{1}{n}\sum_{i=1}^{n}(-1)^{x_i \wedge y_i}|^2 = 1/4$. Thus,

$$Pr(\mathcal{P}(x,y) = \mathrm{DISJ}_{\frac{1}{4}}(x,y)) = 1 - \left|\frac{1}{n}\sum_{i=1}^{n}(-1)^{x_i \wedge y_i}\right|^2 \ge \frac{3}{4}. \tag{31}$$

Therefore $\mathcal{P}$ is a bounded error protocol for $\mathrm{DISJ}_{\frac{1}{4}}$ and $Q(\mathrm{DISJ}_{\frac{1}{4}}) \le 3 + 2\log n$. □

Now we are in position to deal with the general case.

Theorem 5. *$Q(DISJ_\lambda) \le \frac{\log 3}{3\lambda}(3 + 2\log n)$, where $0 < \lambda \le \frac{1}{4}$.*

Proof. For the general case, the new quantum protocol $\mathcal{P}'$ works as follows: Repeat the protocol $\mathcal{P}$ from the proof of previous theorem k times (k will be specified later). If all measurement outcomes in Step 3 are $|1,0\rangle$, then $\mathcal{P}'(x,y)=1$; otherwise, $\mathcal{P}'(x,y)=0$.

If $\sum_{i=1}^{n} x_i \wedge y_i = 0$, then

$$Pr(\mathcal{P}(x,y)=1)=1 \tag{32}$$

and

$$Pr(\mathcal{P}(x,y)=0)=0. \tag{33}$$

Therefore,

$$Pr(\mathcal{P}'(x,y)=\mathrm{DISJ}_\lambda(x,y)=1)=1. \tag{34}$$

If $\lambda n \leq \sum_{i=1}^{n} x_i \wedge y_i \leq (1-\lambda)n$, then

$$p_0 = Pr(\mathcal{P}(x,y)=\mathrm{DISJ}_\lambda(x,y)=0) = 1-|\frac{1}{n}\sum_{i=1}^{n}(-1)^{x_i\wedge y_i}|^2 \geq 1-|1-2\lambda|^2 \tag{35}$$

$$= 4\lambda-\lambda^2 = 4\lambda(1-\lambda) \geq 4\lambda(1-\frac{1}{4}) = 3\lambda. \tag{36}$$

If $k=\frac{\log 1/3}{\log(1-3\lambda)}$, and the protocol $\mathcal{P}$ is repeated k times, then

$$Pr(\mathcal{P}'(x,y)=\mathrm{DISJ}_\lambda(x,y)=0) = 1-(1-p_0)^k \geq 1-(1-3\lambda)^k \geq 1-(1-3\lambda)^{\frac{\log 1/3}{\log(1-3\lambda)}} \tag{37}$$

$$= 1-2^{\log((1-3\lambda)^{\frac{\log 1/3}{\log(1-3\lambda)}})} = 1-2^{\frac{\log 1/3}{\log(1-3\lambda)}\times\log((1-3\lambda)} = 1-2^{\log 1/3} = \frac{2}{3}. \tag{38}$$

Since $1-u \leq e^{-u} \leq 2^{-u}$, for any real number $u>0$, we have

$$k = \frac{\log 1/3}{\log(1-3\lambda)} \leq \frac{\log 1/3}{\log 2^{(-3\lambda)}} = \frac{\log 3}{3\lambda}. \tag{39}$$

Thus, $Q(\mathrm{DISJ}_\lambda) \leq \frac{\log 3}{3\lambda}(3+2\log n)$. □

4.2. Deterministic lower bound

To prove the main result, we will use a modification of the lower bound proof method from [11, 13].

Theorem 6. *$D(DISJ_\lambda) \in \mathbf{\Omega}(n)$, where $0<\lambda\leq\frac{1}{4}$.*

Proof. Let $\mathcal{P}$ be a deterministic protocol for DISJ_λ. Let us consider the set $F_\lambda = \{x\in\{0,1\}^n \,|\, \lambda n \leq W(x) \leq (1-\lambda)n\}$. If $x\in F_\lambda$, then also $\overline{x}\in F_\lambda$, where $\overline{x}=\overline{x}_1\cdots\overline{x}_n$. Let $E=\{(x,\overline{x})\,|\,x\in F_\lambda\}$. For every $(x,\overline{x})\in E$, we then have $\mathcal{P}(x,\overline{x})=1$. Suppose now that there is a 1-monochromatic rectangle $R=A\times B\subseteq\{0,1\}^n\times\{0,1\}^n$ such that $\mathcal{P}(x,y)=1$ for every pair of promise input $(x,y)\in R$. For $S=R\cap E$, we now prove that $|S|<1.99^n$.

Suppose $|S|\geq 1.99^n$. According to Corollary 1.2 from [16], there exist $(x,\overline{x})\in S$ and $(z,\overline{z})\in S$ such that $|x\wedge z|=\frac{n}{4}$. Since $S\subseteq E$, we have $x,\overline{x},z,\overline{z}\in F_\lambda$. Without a lost of generality, let

$$x = \overbrace{1\cdots1}^{n/4}\ \overbrace{0\cdots0}^{\lambda n}\ \overbrace{1\cdots1}^{\lambda n}\ \overbrace{*\cdots*}^{3n/4-2\lambda n} \quad \text{and} \tag{40}$$

$$z = \overbrace{1\cdots1}^{n/4}\ \overbrace{1\cdots1}^{\lambda n}\ \overbrace{0\cdots0}^{\lambda n}\ \overbrace{*\cdots*}^{3n/4-2\lambda n} \tag{41}$$

such that $|x \wedge z| = \frac{n}{4}$. In such a case

$$\overline{x} = \overbrace{0\cdots0}^{n/4}\ \overbrace{1\cdots1}^{\lambda n}\ \overbrace{0\cdots0}^{\lambda n}\ \overbrace{*\cdots*}^{3n/4-2\lambda n} \tag{42}$$

and therefore $\lambda n \leq |z \wedge \overline{x}| \leq 3n/4 - \lambda n < (1-\lambda)n$. Thus, $\mathcal{P}(z, \overline{x}) = 0$. Since $S \subset R$ and R is a 1-rectangle, we get $(x, \overline{x}) \in R, (z, \overline{z}) \in R$ and also $(z, \overline{x}) \in R$. Since $(z, \overline{x})$ is a pair of the promise input, it holds $\mathcal{P}(z, \overline{x}) = 1$, which is a contradiction.

Therefore, the minimum number of 1-monochromatic rectangles that partition the space of inputs is

$$C^1(\mathrm{DISJ}_\lambda) \geq \frac{|E|}{|S|} = \frac{|F_\lambda|}{|S|} \geq \frac{|F_{1/4}|}{|S|} > \frac{2^n/2}{1.99^n}. \tag{43}$$

According to Lemma 1, the deterministic communication complexity then holds:

$$D(\mathrm{DISJ}_\lambda) \geq \log C^1(\mathrm{DISJ}_\lambda) > \log\left(\frac{2^n/2}{1.99^n}\right) = n - 1 - n\log 1.99 \tag{44}$$

$$> n - 1 - 0.9927n = 0.0073n - 1. \tag{45}$$

Thus, $D(\mathrm{DISJ}_\lambda) \in \boldsymbol{\Omega}(n)$. □

Remark 2. The lower bound proved in the previous theorem is quite a weak bound. We expect that a better lower bound will be relative to λ. When λ is close to 0, then the lower bound is expected to be close to n instead of 0.007n.

4.3. Probabilistic protocol

As already mentioned, the two-sided error probabilistic communication complexity $R(\mathrm{DISJ}) \in \boldsymbol{\Omega}(n)$. However, for DISJ_λ, the communication complexity can be dramatically improved as will now be shown.

Let us first deal with the case $\lambda = \frac{1}{4}$.

Theorem 7. $R(DISJ_{\frac{1}{4}}) \leq 5\log n$.

Proof. Let us consider the probabilistic protocol $\mathcal{P}$ which works as follows (where integer k will be speified later).

1. If $W(x) < k$, then Alice sends 1 as the result of $\mathrm{DISJ}_{\frac{1}{4}}(x, y)$ to Bob. Otherwise, Alice chooses randomly k 1's of her input, says $x_{i_1}, \cdots, x_{i_k}$, and sends their positions $i_1, \cdots, i_k$ to Bob.
2. If Bob does not receive 1 as the result from Alice, then he checks the positions $i_1, \cdots, i_k$ of his input. If there exists a $1 \leq j \leq k$ such that $y_{i_j} = 1$, then $\mathcal{P}(x, y) = 0$; otherwise $\mathcal{P}(x, y) = 1$.

If $\sum_{i=1}^n x_i \wedge y_i = 0$, then

$$Pr(\mathcal{P}(x, y) = \mathrm{DISJ}_{\frac{1}{4}}(x, y) = 1) = 1. \tag{46}$$

If $n/4 \leq \sum_{i=1}^n x_i \wedge y_i \leq 3n/4$, then for any $i \in \{i_1, \cdots, i_k\}$

$$Pr(y_i = x_i) \geq \frac{1}{4}. \tag{47}$$

Therefore,

$$Pr(\mathcal{P}(x, y) = 0) \geq 1 - (1 - \frac{1}{4})^k = 1 - (\frac{3}{4})^k. \tag{48}$$

If $k = 5$, then $Pr(\mathcal{P}(x, y) = 0) > 0.76 > \frac{2}{3}$. Since Alice needs $\log n$ bits to specifies every position, we have $R(\mathrm{DISJ}_{\frac{1}{4}}) \leq 5\log n$. □

A more general result we get for all problems $R(\mathrm{DISJ}_\lambda)$ where $0 < \lambda \le \frac{1}{4}$.

Theorem 8. *$R(DISJ_\lambda) \le \frac{\log 3}{\lambda}\log n$, where $0 < \lambda \le \frac{1}{4}$*

Proof. For this general cases, we will use almost the same protocol as in the proof of the previous theorem, only Alice will send to Bob more positions of 1's in her input. It holds:

If $\sum_{i=1}^{n} x_i \wedge y_i = 0$, then

$$Pr(\mathcal{P}(x,y) = \mathrm{DISJ}_\lambda(x,y) = 1) = 1. \tag{49}$$

If $\lambda n \le \sum_{i=1}^{n} x_i \wedge y_i \le (1-\lambda)n$, then for any $i \in \{i_1, \cdots, i_k\}$

$$Pr(y_i = x_i) \ge \lambda. \tag{50}$$

Therefore,

$$Pr(\mathcal{P}(x,y) = 0) \ge 1 - (1-\lambda)^k. \tag{51}$$

If $k = \frac{\log 1/3}{\log(1-\lambda)}$, then $(1-\lambda)^{\frac{\log 1/3}{\log(1-\lambda)}} = \frac{1}{3}$ and $Pr(\mathcal{P}(x,y) = 0) \ge \frac{2}{3}$. Thus, $R(\mathrm{DISJ}_\lambda) \le \frac{\log 1/3}{\log(1-\lambda)}\log n \le \frac{\log 3}{\lambda}\log n$. $\square$

Remark 3. We can also define two-sided error mode as tolerating an error probability ε instead of $\frac{1}{3}$. Modifying our proof in Theorem 5 and Theorem 8, we can get $Q(\mathrm{DISJ}_\lambda) \le \frac{\log\varepsilon}{3\lambda}(3+2\log n)$ and $R(\mathrm{DISJ}_\lambda) \le \frac{\log\varepsilon}{\lambda}\log n$ for any error probability ε.

5. Applications to quantum, probabilistic and deterministic finite automata

It has been known, since the paper [1], that for some regular languages 1QFA can be more succinct than their classical counterparts. However, Klauck [25] proved, for any regular language L, that the state complexity of the exact one-way quantum finite automata for L is not less than the state complexity of an equivalent one-way deterministic finite automata (DFA). Surprisingly, situation is again different for some promise problems [4, 20, 39].

For any $n \in \mathbb{Z}^+$, let us consider the promise problem $A_{EQ_k}(n)$ over an alphabet $\Sigma = \{0, 1, \#\}$, corresponding to the EQ_k problem, that is defined as follow:

$$A_{EQ_k}(n) : \begin{cases} A_{yes}(n) = \{x\#y \,|\, H(x,y) = 0, x, y \in \{0,1\}^n\} \\ A_{no}(n) = \{x\#y \,|\, H(x,y) = k, x, y \in \{0,1\}^n\}, \end{cases} \tag{52}$$

where k is a fixed even such that $k \ge n/2$.

The quantum protocol for EQ_k which is described in Theorem 1 can be implemented on an MO-1QCFA as shown bellow. Therefore, we get the following result:

Theorem 9. *The promise problem $A_{EQ_k}(n)$ can be solved exactly by an MO-1QCFA $\mathcal{A}(n)$ with $n+1$ quantum basis states and $\mathbf{O}(n)$ classical states, whereas the sizes of the corresponding DFA are $2^{\mathbf{\Omega}(n)}$ if k is an even such that $\frac{1}{2}n \le k < (1-\lambda)n$, where $0 < \lambda < \frac{1}{2}$ is given.*

Proof. Let $x = x_1 \cdots x_n$ and $y = y_1 \cdots y_n$. Let us consider an MO-1QCFA $\mathcal{A}(n) = (Q, S, \Sigma, \Theta, \delta, |0\rangle, s_0, Q_a)$, where $Q = \{|i\rangle\}_{i=0}^{n}$, $S = \{s_i\}_{i=0}^{n+1}$ and $Q_a = \{|0\rangle\}$. $\mathcal{A}(n)$ will start in the initial quantum state $|0\rangle$ and then perform the unitary transformation $\Theta(s_0, \cent) = U_{\cent} = U_h U_k$ to the state $|0\rangle$, where U_h, U_k are the ones defined in the proof of Theorem 1. We use classical states $s_i \in S$ $(1 \le i \le n+1)$ to point out the positions of the

1. Read the left end-marker ¢, perform $\Theta(s_0, ¢) = U_{¢} = U_h U_k$ on the initial quantum state $|0\rangle$, change its classical state to $\delta(s_0, ¢) = s_1$, and move the tape head one cell to the right.
2. While the currently scanned symbol σ is not #, do the following:
   - 2.1 Apply $\Theta(s_i, \sigma) = U_{i,\sigma}$ to the current quantum state.
   - 2.2 Change the classical state s_i to s_{i+1} and move the tape head one cell to the right.
3. Change the classical state s_{n+1} to s_1 and move the tape head one cell to the right.
4. While the currently scanned symbol σ is not the right end-marker \$, do the following:
   - 4.1 Apply $\Theta(s_i, \sigma) = U_{i,\sigma}$ to the current quantum state.
   - 4.2 Change the classical state s_i to s_{i+1} and move the tape head one cell to the right.
5. When the right end-marker is reached, perform $\Theta(s_{n+1}, \$) = U_\$ = U_k^{-1} U_h^{-1}$ on the current quantum state and measure the current quantum state with the projective measurement $\{P_a = |0\rangle\langle 0|, P_r = I - |0\rangle\langle 0|\}$. If the outcome is $|0\rangle$, accept the input; otherwise reject the input.

Figure 2: Description of the behavior of $\mathcal{A}(n)$ when solving the promise problem $A_{EQ_k}(n)$.

tape head that will provide some information for quantum transformations. If the classical state of $\mathcal{A}(n)$ is s_i $(1 \leq i \leq n)$, then the next scanned symbol of the tape head is the i-th symbol of $x(y)$ and s_{n+1} means that the next scanned symbol of the tape head is #(\$). The automaton proceeds as shown in Figure 2, where

$$U_{i,\sigma}|i\rangle = (-1)^\sigma |i\rangle \text{ and } U_{i,\sigma}|j\rangle = |j\rangle \text{ for } j \neq i \tag{53}$$

The rest of the proof is analogues to the proof in Theorem 1.

The deterministic communication complexity of EQ_k is $\mathbf{\Omega}(n)$. Therefore, the sizes of the corresponding DFA are $2^{\mathbf{\Omega}(n)}$ [26]. □

We now apply also to finite automata the communication complexity results for DISJ_λ. Let us consider the following promise problem

$$A_D(n): \begin{cases} A_{yes}(n) = \{x\#y\#x \mid \sum_{i=1}^n x_i \wedge y_i = 0, x, y \in \{0,1\}^n\} \\ A_{no}(n) = \{x\#y\#x \mid \frac{1}{4}n \leq \sum_{i=1}^n x_i \wedge y_i \leq \frac{3}{4}n, x, y \in \{0,1\}^n\}. \end{cases} \tag{54}$$

We implement the protocols used in Section 4 for an MO-1QCFA and for a one-way probabilistic finite automaton (1PFA) and get the following result:

Theorem 10. *The promise problem $A_D(n)$ can be solved with one-sided error $\frac{1}{4}$ by an MO-1QCFA $\mathcal{A}(n)$ with $2n$ quantum basis states and $\mathbf{O}(n)$ classical states and also by a 1PFA $\mathcal{P}(n)$ with $\mathbf{O}(n^5)$ states, whereas the sizes of the corresponding DFA are $2^{\mathbf{\Omega}(n)}$.*

Proof. Let $x = x_1 \cdots x_n$ and $y = y_1 \cdots y_n$. Let us consider an MO-1QCFA $\mathcal{A}(n) = (Q, S, \Sigma, \Theta, \delta, |q_0\rangle, s_0, Q_a)$, where $Q = \{|i,0\rangle, |i,1\rangle\}_{i=1}^n$, $|q_0\rangle = |1,0\rangle$ and $Q_a = \{|1,0\rangle\}$. The automaton proceeds as shown in Figure 3, where U_s, U_f are the ones defined in the proof of Theorem 4 and

$$U_{i,\sigma}|j,0\rangle = |j,1\rangle \text{ and } U_{i,\sigma}|j,1\rangle = |j,0\rangle \text{ if } \sigma = 1 \text{ and } j = i, \text{ otherwise } U_{i,\sigma}|j,k\rangle = |j,k\rangle; \tag{55}$$

$$V_{i,\sigma}|j,1\rangle = (-1)^\sigma |j,1\rangle \text{ if } j = i, \text{ otherwise } V_{i,\sigma}|j,k\rangle = |j,k\rangle; \tag{56}$$

1. Read the left end-marker ¢, perform U_s on the initial quantum state $|1,0\rangle$, change its classical state to $\delta(s_0, ¢) = s_1$, and move the tape head one cell to the right.
2. While the currently scanned symbol σ is not #, do the following:
   2.1 Apply $\Theta(s_i, \sigma) = U_{i,\sigma}$ to the current quantum state.
   2.2 Change the classical state s_i to s_{i+1} and move the tape head one cell to the right.
3. Move the tape head one cell to the right.
4. While the currently scanned symbol σ is not #, do the following:
   4.1 Apply $\Theta(s_{n+i}, \sigma) = V_{i,\sigma}$ to the current quantum state.
   4.2 Change the classical state s_{n+i} to s_{n+i+1} and move the tape head one cell to the right.
5. Change the classical state s_{2n+1} to s_1 and move the tape head one cell to the right.
6. While the currently scanned symbol σ is not the right end-marker \$, do the following:
   6.1 Apply $\Theta(s_i, \sigma) = U_{i,\sigma}$ to the current quantum state.
   6.2 Change the classical state s_i to s_{i+1} and move the tape head one cell to the right.
7. When the right end-marker \$ is reached, perform U_f on the current quantum state, measure the current quantum state with the projective measurement $\{P_a = |1,0\rangle\langle 1,0|, P_r = I - P_a\}$. If the outcome is $|1,0\rangle$, accept the input; otherwise reject the input.

Figure 3: Description of the behavior of $\mathcal{A}(n)$ when solving the promise problem $A_D(n)$.

It is easy to verify that for $1 \leq i \leq n$, $U_{i,\sigma}$ and $V_{i,\sigma}$ are unitary transformations. According to the analysis in the proof of Theorem 4, if the input string $w \in A_{yes}(n)$, then the automaton will get the outcome $|1,0\rangle$ in Step 7 with certainty and therefore

$$Pr[\mathcal{A} \text{ accepts } w] = 1. \tag{57}$$

If the input string $w \in A_{no}(n)$, the automaton gets the outcome $|1,0\rangle$ with probability not more than 1/4. Thus,

$$Pr[\mathcal{A} \text{ rejects } w] \geq \frac{3}{4}. \tag{58}$$

Using the protocol from the proof of Theorem 7 and the proof that its probabilistic communication complexity is not more than $5 \log n$, it is easy to design a 1PFA with $O(n^5)$ states to solve the promise problem.

The deterministic state complexity lower bound can now be proved as follows.

Let an N-states DFA $\mathcal{A}'(n) = (S, \Sigma, \delta, s_0, S_{acc})$ solves the promise problem $A_D(n)$, then we can get a deterministic protocol for $\text{DISJ}_{\frac{1}{4}}(x, y)$ as follows:

1. Alice simulates the computation of $\mathcal{A}'(n)$ on the input "$x\#$" and then sends her state $\widehat{\delta}(s_0, x\#)$ to Bob.
2. Bob simulates the computation of $\mathcal{A}'(n)$ on the input "$y\#$" starting at the state $\widehat{\delta}(s_0, x\#)$, and then sends his state $\widehat{\delta}(s_0, x\#y\#)$ to Alice.
3. Alice simulates the computation of $\mathcal{A}'(n)$ on the input "x" starting at the state $\widehat{\delta}(s_0, x\#y\#)$. If $\widehat{\delta}(s_0, x\#y\#x) \in S_{acc}$, then Alice sends the result 1 to Bob, otherwise Alice sends the result 0 to Bob.

The deterministic complexity of the above protocol is $1+2\log N$ and therefore $D(\mathrm{DISJ}_{\frac{1}{4}}) \leq 1+2\log N$. According to the analysis in Theorem 6, we have

$$1+2\log N \geq D(\mathrm{DISJ}_{\frac{1}{4}}) > 0.0073n-1 \tag{59}$$

$$\Rightarrow N \in 2^{\mathbf{\Omega}(n)}. \tag{60}$$

$\square$

6. Conclusion

We have explored generalizations of the Deutsch-Jozsa promise problem and its communication and also query complexities. We have proved that the exact quantum communication complexity $Q_E(\mathrm{EQ}_k) \in \mathbf{O}(\log n)$ for any fixed $k \geq \frac{n}{2}$, whereas the exact classical communication complexity $D(\mathrm{EQ}_k) \in \mathbf{\Omega}(n)$ if k is an even such that $\frac{1}{2}n \leq k < (1-\lambda)n$, where $0 < \lambda < \frac{1}{2}$ is given. We have also shown that the exact quantum query complexity $QT_E(\mathrm{DJ}_k) = 1$ for any fixed $k \geq \frac{n}{2}$, whereas the exact classical query complexity $DT(\mathrm{DJ}_k) = n-k+1$. Promise versions of the disjointness problem also have been discussed. We have proved that for some promise versions of the disjointness problem that there exist exponential gaps between quantum (and also probabilistic) communication complexity and deterministic communication complexity.

Using results of the communication complexity to prove lower bounds of the state complexity of finite automata is one of the important methods [23, 25, 26]. In this paper we have used them not only to prove lower bounds but also upper bounds. Two communicating parties Alice and Bob are supposed to have access to arbitrary computational power in communication complexity models. However, we have also designed communication protocols in Section 3 and Section 4 in which both Alice and Bob are using very limited computational power. The computations of both Alice and Bob can even be simulated by finite automata.

Some problems for future work are:

1. We have generalized the distributed Deutsch-Jozsa promise problem to determine whether $H(x,y)=0$ or $H(x,y)=k$, where k is a fixed integer such that $k \geq \frac{n}{2}$. Does there exist similar results for some cases where $k < \frac{n}{2}$?
2. Does there exist a promise version of the disjointness problem such that its exact quantum communication complexity can be exponential better than its deterministic communication complexity?

Acknowledgements

The authors are thankful to anonymous referees for their comments and suggestions that greatly help to improve the quality of the manuscript. We also thank Abuzer Yakaryılmaz for useful comments on earlier drafts of this paper.

References

[1] A. Ambainis and R. Freivalds, One-way quantum finite automata: strengths, weaknesses and generalizations, in Proceedings of the 39th FOCS (1998) 332–341.

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

[3] S. Aaronson, A. Ambainis, Quantum search of spatial regions, In Proceedings of 44th IEEE FOCS (2003) 200–209.

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

[5] A. Ambainis, Superlinear advantage for exact quantum algorithms, In Proceedings of 45th ACM STOC (2013) 891–900.

[6] A. Ambainis, A. Iraids, J. Smotrovs, Exact quantum query complexity of EXACT and THRESHOLD, In Proceedings of 8th TQC (2013) 263–269. Also arXiv:1302.1235.

[7] A. Ambainis, J. Gruska and S.G. Zheng, Exact quantum algorithms have advantage for almost all Boolean functions, Quantum Information and Computation **15** (2015) 0435–0452. Also arXiv:1404.1684.

[8] Z. Bar-Yossef, T. S. Jayram, R. Kumar, D. Sivakumar, An information statistics approach to data stream and communication complexity, Journal of Computer and System Sciences **68** (2004) 702–732.

[9] G. Brassard, Quantum communication complexity, Foundations of Physics **70** (2003) 1593–1616,

[10] G. Brassard, P. Høyer, An exact quantum polynomial-time algorithm for Simon's problem, In Proceedings of the Israeli Symposium on Theory of Computing and Systems (1997) 12–23.

[11] H. Buhrman, R. Cleve and A. Wigderson, Quantum vs. classical communication and computation, In Proceedings of 30th Annual ACM Symposium on Theory of Computing (1998) 63–68.

[12] H. Buhrman, R. de Wolf. Communication complexity lower bounds by polynomials. In Proceedings of 16th IEEE Conference on Computational Complexity (2001) 120–130.

[13] H. Buhrman, R. Cleve, S. Massar, R. de Wolf, Nonlocality and Communication Complexity, Rev. Mod. Phys. **82** (2010) 665–698. Also arXiv:0907.3584.

[14] H. Buhrman, R.de Wolf, Complexity measures and decision tree complexity: a survey, Theoretical Computer Science **288** (2002) 21–43.

[15] D. Deutsch, R.Jozsa, Rapid solution of problems by quantum computation, In Proceedings of the Royal Society of London (1992), volume A439, 553–558.

[16] P. Frankl and V. Rodl, Forbidden intersections, Trans. Amer. Math. Soc. 300, 1, (1987) 259–286.

[17] O. Goldreich, On promise problems: A survey, Shimon Even Festschrift, LNCS **3895** (2006) 254–290.

[18] J. Gruska, Quantum Computing, McGraw-Hill, London (1999).

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

[20] J. Gruska, D.W. Qiu and S.G. Zheng, Potential of quantum finite automata with exact acceptance, International Journal of Foundation of Computer Science, accepted. Also arXiv:1404.1689 (2014).

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

[22] J. Hromkovič, Communication Complexity and Parallel Computing, Springer, Berlin (1997).

[23] J. Hromkovič, G. Schintger, On the Power of Las Vegas for One-Way Communication Complexity, OBDDs, and Finite Automata, Information and Computation **169** (2001) 284–296.

[24] B. Kalyanasundaram, G. Schintger, The probabilistic communication complexity of set intersection, SIAM Journal on Discrete Mathematics **5** (1992) 545–557.

[25] H. Klauck, On quantum and probabilistic communication: Las Vegas and one-way protocols, Proceedings of the 32th annual ACM symposium on Theory of computing (2000) 644-651.

[26] E. Kushilevitz, N. Nisan, Communication Complexity, Cambridge University Press (1997).

[27] L.Z. Li, Y. Feng, On hybrid models of quantum finite automata, Journal of Computer and System Sciences, to appear, doi:10.1016/j.jcss.2015.01.001 (2015). Also arXiv:1206.2131.

[28] A. Montanaro, R. Jozsa, G. Mitchison, On exact quantum query complexity, Algorithmica, doi:10.1007/s00453-013-9826-8 (2013). Also arXiv:1111.0475.

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

[30] D. W. Qiu, L. Z. Li, P. Mateus, J. Gruska, Quantum finite automata, CRC Handbook of Finite State Based Models and Applications, CRC Press (2012) 113–144.

[31] D.W. Qiu, L.Z. Li, P. Mateus, A. Sernadas, Exponentially more concise quantum recognition of non-RMM regular languages, Journal of Computer and System Sciences **81** (2) 359–375. Also arXiv:0909.1428.

[32] A. Razborov, On the distributional complexity of disjointness, Theor. Comput. Sci. **106** (1992) 385–390.

[33] A. Razborov, Quantum communication complexity of symmetric predicates, Izvestiya of the Russian Academy of Sciences, mathematics **67** (2003) 159–176.

[34] A. C. Yao, Some Complexity Questions Related to Distributed Computing, in: Proc. of 11th STOC (1979) 209–213.

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

[36] S. Yu, State Complexity: Recent Results and Open Problems, Fundamenta Informaticae **64** (2005) 471–480.

[37] S.G. Zheng, D.W. Qiu, L.Z. Li, J. Gruska, One-way finite automata with quantum and classical states, Dassow Festschrift 2012, LNCS **7300** (2012) 273–290.

[38] S. G. Zheng, D. W. Qiu, J. Gruska, L. Z. Li, P. Mateus. State succinctness of two-way finite automata with quantum and classical states, Theoretical Computer Science, **499** (2013) 98–112.

[39] S. G. Zheng, J. Gruska, D. W. Qiu, On the state complexity of semi-quantum finite automata. RAIRO-Theoretical Informatics and Applications **48** 187–207. Earlier version in LATA'14. Also arXiv:1307.2499.

[40] S.G. Zheng, D.W.Qiu, J. Gruska Power of the interactive proof systems with verifiers modeled by semi-quantum two-way finite automata, Information and Computation, to appear, doi:10.1016/j.ic.2015.02.003 (2015). Also arXiv:1304.3876.