Improved Lower Bounds for Locally Decodable Codes and Private Information Retrieval

Stephanie Wehner*  
CWI, Amsterdam  
wehner@cwi.nl

Ronald de Wolf*  
CWI, Amsterdam  
rewolf@cwi.nl

Abstract

We prove new lower bounds for locally decodable codes and private information retrieval. We show that a 2-query LDC encoding $n$-bit strings over an $\ell$-bit alphabet, where the decoder only uses $b$ bits of each queried position of the codeword, needs code length

$$m = \exp \left( \Omega \left( \frac{n}{2^b \sum_{i=0}^{\ell-1} \binom{i}{b}} \right) \right).$$

Similarly, a 2-server PIR scheme with an $n$-bit database and $t$-bit queries, where the user only needs $b$ bits from each of the two $\ell$-bit answers, unknown to the servers, satisfies

$$t = \Omega \left( \frac{n}{2^b \sum_{i=0}^{\ell-1} \binom{i}{b}} \right).$$

This implies that several known PIR schemes are close to optimal. Our results generalize those of Goldreich et al. [6], who proved roughly the same bounds for linear LDCs and PIRs. Like earlier work by Kerenidis and de Wolf [9], our classical lower bounds are proved using quantum computational techniques. In particular, we give a tight analysis of how well a 2-input function can be computed from a quantum superposition of both inputs.

1 Introduction

1.1 Locally decodable codes

Error correcting codes allow reliable transmission and storage of information in noisy environments. Such codes often have the disadvantage that one has to read almost the entire codeword, even if one is only interested in a small part of the encoded information. A locally decodable code $C : \{0,1\}^n \to \Sigma^m$ over alphabet $\Sigma$ is an error-correcting code that allows efficient decoding of individual bits of the encoded information: given any string $y$ that is sufficiently close to the real codeword $C(x)$, we can probabilistically recover any bit $x_i$ of the original input $x$, while only looking at $k$ positions of $y$. The code length $m$ measures the cost of the encoding, while $k$ measures the efficiency of decoding individual bits. Such codes have had a number of applications in recent computer science research, including PCPs and worst-case to average-case reductions. One can also think of applications encoding a large chunk of data in order to protect it from noise, where

*Partially supported by the EU fifth framework project RESQ, IST-2001-37559.
we’re only interested in extracting small pieces at a time. Imagine for example an encoding of all books in a library, where we would like to retrieve only the first paragraph of this paper.

The main complexity question of interest is the tradeoff between $m$ and $k$. With $k = \text{polylog}(n)$ queries, the code length can be made polynomially small, even over the binary alphabet $\Sigma = \{0, 1\}$ [2]. However, for fixed $k$, the best upper bounds are superpolynomial. Except for the $k = 2$ case with fairly small alphabet $\Sigma$, no good lower bounds are known. Katz and Trevisan [8] showed superlinear but at most quadratic lower bounds for constant $k$. Goldreich et al. [6] showed an exponential lower bound for linear codes with $k = 2$ queries and constant alphabet, and Kerenidis and de Wolf [9] extended the latter result to all codes, using techniques from quantum computing. For alphabet $\Sigma = \{0, 1\}^\ell$ their lower bound is
\[
m = 2^{\Omega(n/2^\ell)}.
\]
They also slightly improved the polynomial lower bounds of [8] for $k > 2$.

Clearly the above lower bound becomes trivial if each position of the codeword has $\ell \geq \log(n)/5$ bits. In this paper we analyze the case where $\ell$ can be much larger, but the decoder uses only $b$ bits out of the $\ell$ bits that a query gives. The $b$ positions that he uses may depend on the index $i$ he’s interested in, as well as his randomness. This setting is interesting because many existing constructions are of this form, for quite small $b$. Goldreich et al. [6] also analyzed this situation, and showed the following lower bound for linear codes:
\[
m = 2^{\Omega(n/\sum_i b_i)}.
\]
Here we prove a slightly weaker lower bound for all codes:
\[
m = 2^{\Omega(n/2^b \sum_i b_i)}.
\]
In particular, if $b = \ell$ we improve the bound from [9] to
\[
m = 2^{\Omega(n/2^\ell)}.
\]
We lose a factor of $2^b$ compared to Goldreich et al. This factor can be dispensed with if the decoder outputs the parity of a subset of the bits he receives. All known LDCs are of this type.

Our proofs are completely different from the combinatorial approach of Goldreich et al. Following [9], we proceed in two steps: (1) we reduce the two classical queries to one quantum query and (2) show a lower bound for the induced one-quantum-query-decodable code by deriving a random access code from it. The main novelty is a tight analysis of the following problem, which may be of independent interest. Suppose we want to compute a Boolean function $f(a_0, a_1)$ on $2b$ bits, given a quantum superposition $\frac{1}{\sqrt{2}} (0, a_0 + |1, a_1\rangle)$ of both halves of the input. We show that any Boolean function $f$ can be computed with advantage $1/2^{b+1}$ from this superposition, and that this advantage is best-achievable for the parity function.

1.2 Private information retrieval

There is a very close connection between LDCs and the setting of private information retrieval. In PIR, the user wants to retrieve some item from a database without letting the database learn anything at all about what item he asked for. In the general model, the user retrieves the $i$th bit from an $n$-bit database $x = x_1 \ldots x_n$ that is replicated over $k \geq 1$ non-communicating servers. He communicates with each server without revealing any information about $i$ to individual servers,
and at the end of the day learns $x_i$. This is a natural cryptographic problem that could have applications in systems where privacy of the user is important, for example databases providing medical information. Much research has gone into optimizing the communication complexity of one-round PIR schemes. Here the user sends a $t$-bit message (“query”) to each server, who responds with an $\ell$-bit message (“answer”), from which the user infers $x_i$. A number of non-trivial upper bounds have been found [5, 1, 3, 4], but, as in the LDC case, the optimality of such schemes is open. In fact, the best known constructions of LDCs with constant $k$ come from PIR schemes with $k$ servers. Roughly speaking, concatenating the servers’ answers to all possible queries gives a codeword $C(x)$ of length $m = k2^t$ over the alphabet $\Sigma = \{0, 1\}^\ell$ that is decodable with $k$ queries. The privacy of the PIR scheme translates into the error-correcting property of the LDC: since many different sets of $k$ queries have to work for recovering $x_i$, we can afford some corruption. Conversely, we can turn a $k$-query LDC into a $k$-server PIR scheme by asking one query to each server (so $t = \log m$). The privacy of the resulting PIR scheme follows from the fact that an LDC can be shown to have a “smoothness” property, meaning that most positions are about equally likely to be queried independent of the index $i$.

Here we will restrict attention to the 2-server PIR case. The paper by Chor et al. [5] that introduced PIR, gave a PIR scheme where both the queries to the servers and the answers from the servers have length $\Theta(n^{1/3})$ bits. Later constructions gave alternative ways of achieving the same complexity, but have not given asymptotic improvements for the 2-server case (in contrast to the case of 3 or more servers [4]). Though general lower bounds for 2-server PIRs still elude us, reasonably good lower bounds can be proven for schemes that either have very short answers, or only use a small number of bits from longer answers. Goldreich et al. [5] give a lemma translating 2-server PIRs to 2-query LDCs, where the property of only using $b$ bits from each $\ell$-bit string carries over. Combining this lemma with our LDC lower bounds gives the following bound for 2-server PIRs with $t$-bit queries and $\ell$-bit answers that use only $b$ bits from each answer:

$$t = \Omega(n/2^b \sum_{i=0}^{b} \binom{\ell}{i}).$$

In particular, for fixed $b$ the overall communication is $C = 2(t + \ell) = \Omega(n^{1/(b+1)})$. This is tight for $b = 1$, we describe an $O(\sqrt{n})$ scheme in Section 2.5.1. It is close to optimal for $b = 3$, since a small variation of the Chor et al. scheme achieves $C = O(n^{1/3})$ using only 3 bits from each answer (Section 2.5.2).\footnote{An alternative polynomial-based $O(n^{1/3})$-scheme from [3] does not have this “small $b$”-property.} For general schemes, where $b = \ell$, we obtain

$$t = \Omega(n/2^{2\ell}).$$

This improves the $n/2^{5\ell}$ bound from [9]. It implies a lower bound of $5 \log n$ on the total communication $C = 2(t + \ell)$, which is incredibly weak, but still an improvement over what was known [10, 9]. Similar results were already established for linear PIR schemes by Goldreich et al., but our results now apply to all PIR schemes. In particular, they imply that in improved 2-server PIR schemes, the user will need to look at more bits from the servers’ answers.

2 Preliminaries

2.1 Notation

Throughout this paper we will use $a|_S$ to denote the string $a$ restricted to a set of bits $S \subseteq [n] = \{1, \ldots, n\}$. For example, $11001|_{\{1,4,5\}} = 101$. We identify a set $S \subseteq [n]$ with $n$-bit string
$S = S_1 \ldots S_n$, where $i \in S$ iff the $i$th bit $S_i = 1$. We use $e_i$ for the $n$-bit string corresponding to the singleton set $S = \{i\}$. Furthermore, if $y \in \Sigma^m$ where $\Sigma = \{0, 1\}^\ell$, then $y_j \in \Sigma$ denotes its $j$th entry, and $y_{j, i}$ with $i \in [\ell]$ is the $i$th bit of $y_j$.

### 2.2 Quantum

We assume general familiarity with the quantum model [12]. Since our proofs depend heavily on the notion of a quantum query, we briefly review the definition. We consider queries with $\ell$-bit answers, where $\ell \geq 1$. For $\Sigma = \{0, 1\}^\ell$, a quantum query to a string $y \in \Sigma^m$ is the unitary transformation specified by

$$|j\rangle|z\rangle \mapsto |j\rangle|z \oplus y_j\rangle$$

where $j \in [m]$, $z \in \{0, 1\}^\ell$ is called the target register, and $z \oplus y_j$ is the string resulting from the xor of the individual bits of $z$ and $y_j$, i.e. $z \oplus y_j = (z_1 \oplus y_{j, 1}) \ldots (z_\ell \oplus y_{j, \ell})$. It is sometimes convenient to get the query result in the phase. To achieve this, define

$$|z_T\rangle = \frac{1}{\sqrt{2^\ell}} \bigotimes_{i=1}^{\ell} (|0\rangle + (-1)^{T_i}|1\rangle)$$

where $T_i$ is the $i$th bit of the $\ell$-bit string $T$. Since $|0 \oplus y_{j, i}\rangle + (-1)^{T_i}|1 \oplus y_{j, i}\rangle = (-1)^{T_i y_{j, i}} (|0\rangle + (-1)^{T_i}|1\rangle)$, a query maps

$$|j\rangle|z_T\rangle \mapsto |j\rangle(-1)^{T_j y_j}|z_T\rangle.$$

### 2.3 Locally decodable codes

A locally decodable code is an error-correcting code that allows efficient decoding of individual bits.

**Definition 1** $C : \{0, 1\}^n \rightarrow \Sigma^m$ is a $(k, \delta, \varepsilon)$-locally decodable code (LDC), if there exists a classical randomized decoding algorithm $A$ with input $i \in [n]$ and oracle access to a string $y \in \Sigma^m$ such that

1. $A$ makes $k$ distinct queries $j_1, \ldots, j_k$ to $y$, non-adaptively, gets query answers $a_1 = y_{j_1}, \ldots, a_k = y_{j_k}$ and outputs a bit $f(a_1, \ldots, a_k)$, where $f$ depends on $i$ and $A$’s randomness.

2. For every $x \in \{0, 1\}^n$, $i \in [n]$, and $y \in \Sigma^m$ with Hamming distance $d(x, C(x)) \leq \delta m$ we have

$$\Pr[f(a_1, \ldots, a_k) = x_i] \geq 1/2 + \varepsilon.$$

Here probabilities are taken over $A$’s internal randomness. For $\Sigma = \{0, 1\}^\ell$, we say the LDC uses $b$ bits, if $A$ only uses $b$ predetermined bits of each query answer: it outputs $f(a_1|S_1, \ldots, a_k|S_k)$ where the sets $S_1, \ldots, S_k$ are of size $b$ each and are determined by $i$ and $A$’s randomness.

The LDC is called linear, if $C$ is a linear function over $GF(2)$ (i.e. $C(x + y) = C(x) + C(y)$).

### 2.4 Smooth codes

In our arguments we will use smooth codes. These are codes where the decoding algorithm spreads its queries "smoothly" across the codeword, meaning it queries no code location too frequently.

**Definition 2** $C : \{0, 1\}^n \rightarrow \Sigma^m$ is a $(k, c, \varepsilon)$-smooth code (SC), if there exists a classical randomized decoding algorithm $A$ with input $i \in [n]$ and oracle access to $C(x)$ such that
1. A makes \( k \) distinct queries \( j_1, \ldots, j_k \) to \( C(x) \), non-adaptively, gets query answers \( a_1 = C(x)_{j_1}, \ldots, a_k = C(x)_{j_k} \) and outputs a bit \( f(a_1, \ldots, a_k) \), where \( f \) depends on \( i \) and \( A \)'s randomness.

2. For every \( x \in \{0,1\}^n \) and \( i \in [n] \) we have \( \Pr[f(a_1, \ldots, a_k) = x_i] \geq 1/2 + \varepsilon \).

3. For every \( x \in \{0,1\}^n \), \( i \in [n] \) and \( j \in [m] \), \( \Pr[A \text{ queries } j] \leq c/m \).

The smooth code uses \( b \) bits, if \( A \) only uses \( b \) predetermined bits of each query answer.

Note that the decoder of smooth codes deals only with valid codewords \( C(x) \). The decoding algorithm of an LDC on the other hand can deal with corrupted codewords \( y \) that are still sufficiently close to the original. Katz and Trevisan [8, Theorem 1] showed that LDCs and smooth codes are closely related:

**Theorem 1 (Katz & Trevisan)** If \( C : \{0,1\}^n \to \Sigma^m \) is a \((k, \delta, \varepsilon)\)-locally decodable code, then \( C \) is also a \((k, k/\delta, \varepsilon)\)-smooth code (the property of using \( b \) bits carries over).

The following definition of a one-query quantum smooth code is rather ad hoc and not the most general possible, but sufficient for our purposes.

**Definition 3** \( C : \{0,1\}^n \to \Sigma^m \) is a \((1, c, \varepsilon)\)-quantum smooth code (QSC), if there exists a quantum decoding algorithm \( A \) with input \( i \in [n] \) and oracle access to \( C(x) \) such that

1. A probabilistically picks a string \( r \), makes a query of the form

   \[
   |Q_{ir} = \frac{1}{\sqrt{2}} \left( |j_{ir} \rangle \frac{1}{\sqrt{2}^b} \sum_{T \subseteq S_{1r}} |z_T \rangle + |j_{2r} \rangle \frac{1}{\sqrt{2}^b} \sum_{T \subseteq S_{2r}} |z_T \rangle \right)
   \]

   and returns the outcome of some quantum measurement on the resulting state.

2. For every \( x \in \{0,1\}^n \) and \( i \in [n] \) we have \( \Pr[A \text{ outputs } x_i] \geq 1/2 + \varepsilon \).

3. For every \( x \in \{0,1\}^n \), \( i \in [n] \) and \( j \in [m] \), \( \Pr[A \text{ queries } j] \leq c/m \).

The QSC uses \( b \) bits, if the sets \( S_{1r}, S_{2r} \) have size \( b \).

2.5 Private information retrieval

A PIR scheme allows a user to retrieve the \( i \)th bit from an \( n \)-bit database \( x \), replicated over \( k \geq 1 \) servers, without revealing any information about \( i \) to individual database servers.

**Definition 4** A one-round, \((1 - \eta)\)-secure, \( k \)-server private information retrieval (PIR) scheme for a database \( x \in \{0,1\}^n \) with recovery probability \( 1/2 + \varepsilon \), query size \( t \), and answer size \( \ell \), consists of a randomized algorithm (user) and \( k \) deterministic algorithms \( S_1, \ldots, S_k \) (servers), such that

1. On input \( i \in [n] \), the user produces \( k \) \( t \)-bit queries \( q_1, \ldots, q_k \) and sends these to the respective servers. The \( j \)th server sends back an \( \ell \)-bit string \( a_j = S_j(x, q_j) \). The user outputs a bit \( f(a_1, \ldots, a_k) \) where \( f \) depends on \( i \) and his randomness.

2. For every \( x \in \{0,1\}^n \) and \( i \in [n] \) we have \( \Pr[f(a_1, \ldots, a_k) = x_i] \geq 1/2 + \varepsilon \).
3. For all \( x \in \{0, 1\}^n \), \( j \in [k] \), and any two indices \( i_1, i_2 \in [n] \), the two distributions on \( q_j \) (over the user’s randomness) induced by \( i_1 \) and \( i_2 \) are \( \eta \)-close in total variation distance.

The scheme uses \( b \) bits, if the user only looks at \( b \) predetermined bits from each of \( a_1, \ldots, a_k \).

The scheme is called linear, if for every \( j \) and \( q_j \) the \( j \)th server’s answer \( S_j(x, q_j) \) is a linear combination (over \( GF(2) \)) of the bits of \( x \).

The setting \( \eta = 0 \) corresponds to the case where the server gets no information at all about \( i \).

All known non-trivial PIR schemes have \( \eta = 0 \), perfect recovery \( (\varepsilon = 1/2) \), and only one round of communication.

2.5.1 Square scheme

To illustrate the concept of PIR, we take a quick look at two well-known 2-server PIR schemes from [5], each having \( \eta = 0 \) and \( \varepsilon = 1/2 \). The first arranges the database \( x = x_1 \ldots x_n \) in a square:

\[
\begin{pmatrix}
  x_1 & x_2 & \cdots & x_{\sqrt{n}} \\
  x_{\sqrt{n}+1} & \cdots & \cdots & x_{2\sqrt{n}} \\
  \vdots & \ddots & \ddots & \vdots \\
  \vdots & \cdots & \cdots & x_n
\end{pmatrix}
\]

The index \( i \) can now be described by two coordinates \((i_1, i_2)\). The user picks a random string \( A \in \{0, 1\}^{\sqrt{n}} \), and sends \( \sqrt{n} \)-bit queries \( q_1 = A \) and \( q_2 = A \oplus e_1 \) to the two servers, respectively. The first server returns the \( \sqrt{n} \)-bit answer \( a_1 = q_1 \cdot C_1, \ldots, q_1 \cdot C_{\sqrt{n}} \), where \( q_1 \cdot C_c \) denotes the inner product mod 2 of \( q_1 \) with the \( c \)th column of \( x \). The second server sends \( a_2 \) analogously. The user selects the bit \( q_1 \cdot C_{i_1} \) from \( a_1 \) and \( q_2 \cdot C_{i_2} \) from \( a_2 \) and xors these two bits to get \((A \cdot C_{i_2}) \oplus ((A \oplus e_1) \cdot C_{i_2}) = e_{i_1} \cdot C_{i_2} = x_i \). This scheme has \( t = \ell = \sqrt{n} \) and uses \( b = 1 \) bits from each answer.

2.5.2 Cube scheme

A more efficient scheme arranges \( x \) in a cube instead of a square, so each \( i \) can be described by 3 coordinates \((i_1, i_2, i_3)\). The user picks 3 random strings \( T_1, T_2, \) and \( T_3 \) of \( n^{1/3} \) bits each, and sends two queries \( q_1 = T_1, T_2, T_3 \) and \( q_2 = (T_1 \oplus e_1), (T_2 \oplus e_2), (T_3 \oplus e_3) \). The first server computes

\[
b_{T_1T_2T_3} = \bigoplus_{j_1 \in T_1, j_2 \in T_2, j_3 \in T_3} x_{j_1j_2j_3},
\]

Its answer \( a_1 \) consists of the bits \( b_{T'_1T'_2T'_3} \oplus b_{T_1T_2T_3} \oplus b_{T_1T_2T_3} \oplus b \) for all \( T'_j \) with \( j \in \{1, 2, 3\} \) differing from \( T_j \) in exactly one place. The second server does the same with its query \( q_2 \). Note that the answer length is \( \ell = 3n^{1/3} \). The user now selects those 3 bits of each answer that correspond to \( T'_1 = T_1 \oplus e_1, T'_2 = T_2 \oplus e_2, T'_3 = T_3 \oplus e_3 \) respectively, and xors those 6 bits. Since every other \( x_{j_1j_2j_3} \) occurs exactly twice in the sum, what is left is just \( x_{i_1, i_2, i_3} = x_i \).

3 Computing \( f(a_0, a_1) \) from Superposed Input

In order to prove the lower bound on LDCs and PIRs, we first construct the following quantum tool. Consider a state \( |\Psi_{a_0a_1}\rangle = \frac{1}{\sqrt{2}}(|0, a_0\rangle + |1, a_1\rangle) \) with \( a_0, a_1 \) both \( b \)-bit strings. We show that
we can compute any Boolean function $f(a_0, a_1)$ with bias $1/2^{b+1}$ given one copy of this state. After that we show that bias optimal if $f$ is the $2b$-bit parity function.

3.1 Upper bound

The key to constructing the algorithm is the following observation:

**Lemma 1** For every function $f : \{0, 1\}^{2b} \to \{0, 1\}$ there exist non-normalized states $|\varphi_a\rangle$ such that

$$U : |a\rangle |0\rangle \to \gamma \sum_{w \in \{0, 1\}^b} (-1)^{f(w, a)} |w\rangle |0\rangle + |\varphi_a\rangle |1\rangle,$$

with $\gamma = 1/2^b$, is unitary.

**Proof.** Let $|\psi_a\rangle = \gamma \sum_{w \in \{0, 1\}^b} (-1)^{f(w, a)} |w\rangle |0\rangle + |\varphi_a\rangle |1\rangle$. $U$ is unitary if and only if $\langle \psi_a | \psi_{a'} \rangle = \delta_{aa'}$ for all $a, a'$. We show that we can choose $|\varphi_a\rangle$ to achieve this.

First, since $\langle w | w' \rangle = \delta_{ww}$ and $\langle w, 0 | \varphi_a, 1 \rangle = 0$, we have

$$\langle \psi_a | \psi_{a'} \rangle = \gamma^2 \sum_{w \in \{0, 1\}^b} (-1)^{f(w, a) + f(w, a')} + \langle \varphi_a | \varphi_{a'} \rangle.$$

Let $C$ denote the $2^b \times 2^b$ matrix with entries $C_{aa'} = \gamma^2 \sum_{w \in \{0, 1\}^b} (-1)^{f(w, a) + f(w, a')}$ where the indices $a$ and $a'$ are $b$-bit strings. From the definition of $C_{aa'}$ we have $|C_{aa'}| \leq 1/2^b$ for $\gamma = 1/2^b$. Then by [7, Corollary 6.1.5], the largest eigenvalue of $C$ is

$$\lambda_{\max}(C) \leq \min \left\{ \max_{a} \sum_{a' \in \{0, 1\}^b} |C_{aa'}|, \max_{a' \in \{0, 1\}^b} \sum_{a \in \{0, 1\}^b} |C_{aa'}| \right\} \leq \sum_{a \in \{0, 1\}^b} \frac{1}{2^b} = 1.$$  

However, $\lambda_{\max}(C) \leq 1$ implies that the Hermitian matrix $I - C$ is positive semidefinite and hence, by [7, Corollary 7.2.11], $I - C = A^\dagger A$ for some matrix $A$. Now define $|\varphi_a\rangle$ to be the $a$th column of $A$. Since the matrix $C + A^\dagger A = I$ is composed of all inner products $\langle \psi_a | \psi_{a'} \rangle$, we have $\langle \psi_a | \psi_{a'} \rangle = \delta_{aa'}$ and it follows that $U$ is unitary. 

Using these observations, we can now prove the following theorem.

**Theorem 2** Suppose $f : \{0, 1\}^{2b} \to \{0, 1\}$ is a Boolean function. There exists a quantum algorithm to compute $f(a_0, a_1)$ with success probability $1/2 + 1/2^{b+1}$ using one copy of $|\Psi_{a_0, a_1}\rangle = \frac{1}{\sqrt{2}}(|0, a_0\rangle + |1, a_1\rangle)$, with $a_0, a_1 \in \{0, 1\}^b$.

**Proof.** First we extend the state $|\Psi_{a_0, a_1}\rangle$ by a $|0\rangle$-qubit. Let $U$ be as in Lemma 1. Applying the unitary transform $|0\rangle |0\rangle \otimes I^{\otimes b+1} + |1\rangle |1\rangle \otimes U$ to $|\Psi_{a_0, a_1}\rangle |0\rangle$ gives

$$\frac{1}{\sqrt{2}} \left( |0\rangle |0\rangle |0\rangle + |1\rangle \left( \frac{1}{2^b} \sum_{w \in \{0, 1\}^b} (-1)^{f(w, a_1)} |w\rangle |0\rangle + |\varphi_{a_1}\rangle |1\rangle \right) \right).$$

Define $|\Gamma\rangle = |a_0\rangle |0\rangle$ and $|A\rangle = \frac{1}{2^b} \sum_{w} (-1)^{f(w, a_1)} |w\rangle |0\rangle + |\varphi_{a_1}\rangle |1\rangle$. Then $\langle \Gamma | A \rangle = \frac{1}{2^b} (-1)^{f(a_0, a_1)}$ and the above state is

$$\frac{1}{\sqrt{2}} (|0\rangle |\Gamma\rangle + |1\rangle |A\rangle).$$
We apply a Hadamard transform to the first qubit to get
\[ \frac{1}{2} \left( |0\rangle (|\Gamma\rangle + |\Lambda\rangle) + |1\rangle (|\Gamma\rangle - |\Lambda\rangle) \right). \]

The probability that a measurement of the first qubit yields a 0 is
\[ \frac{1}{4} \left( |\Gamma\rangle + \Lambda |\Gamma\rangle + \Lambda \right) = \frac{1}{2} + \frac{1}{2} (\Gamma |\Lambda\rangle) = \frac{1}{2} + \frac{(-1)^{f(a_0,a_1)}}{2^{b+1}}. \]

Thus by measuring the first qubit, we obtain the value of \( f(a_0,a_1) \) with bias \( 1/2^{b+1} \). \( \square \)

### 3.2 Lower bound

To prove that the above algorithm is optimal for the parity function, we need to consider how well we can distinguish two density matrices \( \rho_0 \) and \( \rho_1 \). By distinguishing we mean that given an unknown state, we can determine whether it is \( \rho_0 \) or \( \rho_1 \). Let \( \| A \|_{tr} \) denote the trace norm of matrix \( A \), which equals the sum of its singular values.

**Lemma 2** Two density matrices \( \rho_0 \) and \( \rho_1 \) cannot be distinguished with probability better than
\[ \frac{1}{2} + \frac{1}{4} \frac{\| \rho_0 - \rho_1 \|_{tr}}{\frac{1}{4}}. \]

**Proof.** The most general way of distinguishing \( \rho_0 \) and \( \rho_1 \) is a POVM with two operators \( E_0 \) and \( E_1 \), such that \( \rho_0 = tr(\rho_0 E_0) \geq 1/2 + \varepsilon \) and \( q_0 = tr(\rho_1 E_0) \leq 1/2 - \varepsilon \). Then \( |\rho_0 - q_0| \geq 2\varepsilon \) and likewise, \( |p_1 - q_1| \geq 2\varepsilon \). By \( [12, \text{Theorem 9.1}] \), \( \| \rho_0 - \rho_1 \|_{tr} = \max_{E_0,E_1} (|p_0 - q_0| + |p_1 - q_1|) \) and thus \( \| \rho_0 - \rho_1 \|_{tr} \geq 4\varepsilon \). Hence \( \varepsilon \leq \| \rho_0 - \rho_1 \|_{tr}/4 \). \( \square \)

**Theorem 3** Suppose that \( f \) is the parity of \( a_0 a_1 \). Then any quantum algorithm for computing \( f \) from one copy of \( |\Psi_{a_0 a_1}\rangle \) has success probability \( \leq 1/2 + 1/2^{b+1} \).

**Proof.** Define \( \rho_0 \) and \( \rho_1 \) by
\[ \rho_c = \frac{1}{2^{2b-1}} \sum_{a_0 a_1 \in f^{-1}(c)} |\Psi_{a_0 a_1}\rangle \langle \Psi_{a_0 a_1}|, \]
with \( c \in \{0,1\} \). A quantum algorithm that computes the parity of \( a_0 a_1 \) with probability \( 1/2 + \varepsilon \) can be used to distinguish \( \rho_0 \) and \( \rho_1 \). Hence from Lemma 2 we have \( \varepsilon \leq \| \rho_0 - \rho_1 \|_{tr}/4 \).

Let \( A = \rho_0 - \rho_1 \). It is easy to see that the \( |0, a_0\rangle \langle 0, a_0| \) entries are the same in \( \rho_0 \) and in \( \rho_1 \), so these entries are 0 in \( A \). Similarly, the \( |1, a_1\rangle \langle 1, a_1| \) entries in \( A \) are 0. In the off-diagonal blocks, the \( |0, a_0\rangle \langle 1, a_1| \) entry of \( A \) is \( (-1)^{|a_0|+|a_1|}/2^b \). For \( |\phi\rangle = \frac{1}{\sqrt{2^b}} \sum_{w \in \{0,1\}^b} (-1)^w |w\rangle \) we have
\[ |\phi\rangle \langle \phi| = \frac{1}{2^b} \sum_{a_0 a_1} (-1)^{|a_0|+|a_1|} |a_0\rangle \langle a_1| \]
and hence
\[ A = \frac{1}{2^b} (|0, \phi\rangle \langle 1, \phi| + |1, \phi\rangle \langle 0, \phi|). \]
Let $U$ and $V$ be unitary transforms such that $U|0,\phi\rangle = |0,0^b\rangle$, $U|1,\phi\rangle = |1,0^b\rangle$ and $V|0,\phi\rangle = |1,0^b\rangle$, $V|1,\phi\rangle = |0,0^b\rangle$. Then

$$UAV^\dagger = \frac{1}{2^b} (U|0,\phi\rangle\langle 1,\phi|V^\dagger + U|1,\phi\rangle\langle 0,\phi|V^\dagger) = \frac{1}{2^b} (|0,0^b\rangle\langle 0,0^b| + |1,0^b\rangle\langle 1,0^b|).$$

Since $UAV^\dagger$ is diagonal, its only non-zero singular values are $\sigma_1 = \sigma_2 = 1/2^b$. Hence

$$\|\rho_0 - \rho_1\|_\text{tr} = \|A\|_\text{tr} = \|UAV^\dagger\|_\text{tr} = \sum_i \sigma_i = \frac{2}{2^b},$$

so $\varepsilon \leq \|\rho_0 - \rho_1\|_\text{tr}/4 = 1/2^{b+1}$.

\[\square\]

4 Lower Bounds for Locally Decodable Codes that Use Few Bits

We now make use of the technique developed above to prove new lower bounds for 2-query LDCs over non-binary alphabets. First we construct a 1-query quantum smooth code (QSC) from a 2-query smooth code (SC), and then prove lower bounds for QSCs. In the sequel, we will index the two queries by 0 and 1 instead of 1 and 2, to conform to the two basis states $|0\rangle$ and $|1\rangle$ of a qubit.

4.1 Constructing a 1-query QSC from a 2-query SC

Theorem 4 If $C : \{0,1\}^n \rightarrow (\{0,1\}^\ell)^m$ is a $(2,c,\varepsilon)$-smooth code that uses $b$ bits, then $C$ is a $(1,c,\varepsilon/2^b)$-quantum smooth code that uses $b$ bits.

Proof. Fix index $i \in [n]$ and encoding $y = C(x)$. The 1-query quantum decoder will pick a random string $r$ with the same probability as the 2-query classical decoder. This $r$ determines two indices $j_0, j_1 \in [m]$, two $b$-element sets $S_0, S_1 \subseteq [\ell]$, and a function $f : \{0,1\}^{2^b} \rightarrow \{0,1\}$ such that

$$\Pr[f(y_{j_0}|S_0, y_{j_1}|S_1) = x_i] = p \geq \frac{1}{2} + \varepsilon,$$

where the probability is taken over the decoder's randomness. Assume for simplicity that $j_0 = 0$ and $j_1 = 1$, and define $a_0 = y_{j_0}|S_0$ and $a_1 = y_{j_1}|S_1$. We now construct a 1-query quantum decoder that outputs $f(a_0, a_1)$ with probability $1/2 + 1/2^{b+1}$, as follows. The quantum query is

$$|Q_{a_0} \rangle = \frac{1}{\sqrt{2}} \left( |0\rangle \frac{1}{\sqrt{2^b}} \sum_{T \subseteq S_0} |z_T\rangle + |1\rangle \frac{1}{\sqrt{2^b}} \sum_{T \subseteq S_1} |z_T\rangle \right).$$

The result of this query is

$$\frac{1}{\sqrt{2}} \left( |0\rangle \frac{1}{\sqrt{2^b}} \sum_{T \subseteq S_0} (-1)^{a_0 \cdot T} |z_T\rangle + |1\rangle \frac{1}{\sqrt{2^b}} \sum_{T \subseteq S_1} (-1)^{a_1 \cdot T} |z_T\rangle \right).$$

We can unitarily transform this to

$$\frac{1}{\sqrt{2}} \left( |0\rangle |a_0\rangle + |1\rangle |a_1\rangle \right).$$

9
By Theorem 2, we can compute a bit \( o \) from this such that \( \Pr[o = f(a_0, a_1)] = 1/2 + 1/2^{b+1} \). The probability of success is then given by

\[
\Pr[o = x_i] = \Pr[o = f(a_0, a_1)] \cdot \Pr[x_i = f(a_0, a_1)] + \Pr[o \neq f(a_0, a_1)] \cdot \Pr[x_i \neq f(a_0, a_1)]
\]

\[
= \left( \frac{1}{2} + \frac{1}{2^{b+1}} \right) p + \left( \frac{1}{2} - \frac{1}{2^{b+1}} \right) (1 - p)
\]

\[
= \frac{1}{2} - \frac{1}{2^{b+1}} + \frac{1}{2^b} p
\]

\[
\geq \frac{1}{2} + \varepsilon.
\]

Since no \( j \) is queried with probability more than \( c/m \) by the classical decoder, the same is true for the quantum decoder. Hence we have constructed a QSC with the appropriate properties. \( \Box \)

### 4.2 Improved lower bounds for 2-query LDCs over an \( \ell \)-bit alphabet

Our proof of a lower bound for 2-query LDCs uses the notion of a quantum random access code. That is an encoding \( x \mapsto \rho_x \) of \( n \)-bit string \( x \) into \( m \)-qubit states \( \rho_x \), such that any bit \( x_i \) can be recovered with some probability \( p \geq 1/2 + \varepsilon \) from \( \rho_x \). For the length of such quantum codes there is a known lower bound [11]:

**Theorem 5 (Nayak)** An encoding \( x \mapsto \rho_x \) of \( n \)-bit strings into \( m \)-qubit states with recovery probability at least \( p \) has \( m \geq (1 - H(p))n \).

The main ingredient of our proof is the following lemma, which shows how the query of a QSC gives rise to a quantum random access code. Let \( u = \sum_{i=0}^b \binom{\ell}{i} \) and define the \( \log(u) \)-qubit pure states

\[
|U(x)_j\rangle = \frac{1}{\sqrt{u}} \sum_{|T| \leq b} (-1)^{|T\cdot C(x)|} |z_T\rangle
\]

and the \( (\log(m) + \log(u)) \)-qubit states

\[
|U(x)\rangle = \frac{1}{\sqrt{m}} \sum_{j=1}^m |j\rangle |U(x)_j\rangle.
\]

**Lemma 3** Suppose \( C : \{0,1\}^n \rightarrow \{(0,1)^\ell\}^m \) is a \((1,c,\varepsilon)\)-quantum smooth code that uses \( b \) bits. Then given one copy of \( |U(x)\rangle \), there is a quantum algorithm that outputs ‘fail’ with probability \( 1 - 2^{b+1}/cu \) with \( u = \sum_{i=0}^b \binom{\ell}{i} \), but if it succeeds it outputs \( x_i \) with probability at least \( 1/2 + \varepsilon \).

**Proof.** Let us fix \( i \in [n] \). Suppose the quantum decoder of \( C \) makes query \( |Q_{j_0r}\rangle \) to indices \( j_0 \) and \( j_{1r} \) with probability \( p_r \). Consider the following state

\[
|V_i(x)\rangle = \sum_r \sqrt{p_r} |j_0\rangle \frac{1}{\sqrt{2}} (|j_{0r}\rangle |U(x)_{j_{0r}}\rangle + |j_{1r}\rangle |U(x)_{j_{1r}}\rangle).
\]

We will first show how to obtain \( |V_i(x)\rangle \) from \( |U(x)\rangle \) with some probability. Rewrite \( |V_i(x)\rangle \) to

\[
|V_i(x)\rangle = \sum_{j=1}^m \alpha_j |j\rangle |U(x)_j\rangle,
\]

with probability \( p_r \). Consider the following state

\[
|V_i(x)\rangle = \sum_{j=1}^m \alpha_j |j\rangle |U(x)_j\rangle,
\]

10
where the \( \alpha_j \) are nonnegative reals, and \( \alpha_j^2 \leq c/(2m) \) because \( C \) is a QSC (the \( 1/2 \) comes from the amplitude \( 1/\sqrt{2} \)). Using the unitary map \( |0\rangle|j\rangle \mapsto |\phi_j\rangle|j\rangle \), we can obtain \( |V_i(x)\rangle \) from the state

\[
|V_i(x)\rangle = \sum_{j=1}^{m} \alpha_j |j\rangle |U(x)_j\rangle.
\]

We thus have to show that we can obtain \( |V_i(x)\rangle \) from \( |U(x)\rangle \). To this end, define operator

\[
M = \frac{2m}{c} \sum_{j=1}^{m} \alpha_j |j\rangle \langle j| \otimes I
\]

and consider a POVM with measurement operators \( M^\dagger M \) and \( I - M^\dagger M \). These operators are both positive because \( \alpha_j^2 \leq c/2m \). Note that, up to normalization, \( M|U(x)\rangle = |V_i(x)\rangle \). The probability that the measurement succeeds (i.e., takes us from \( |U(x)\rangle \) to \( |V_i(x)\rangle \)) is

\[
\langle U(x)| M^\dagger M |U(x)\rangle = \frac{2m}{c} \langle U(x)| \left( \sum_j \alpha_j^2 |j\rangle \langle j| \otimes I \right) |U(x)\rangle = \frac{2}{c} \sum_j \alpha_j^2 = \frac{2}{c}
\]

Now given \( |V_i(x)\rangle \) we can measure \( r \), and then project the last register onto the sets \( S_{0r} \) and \( S_{1r} \) that we need for \( |Q_{ir}\rangle \), by means of the measurement operator

\[
|j_{ir}\rangle \langle j_{ir}| \otimes \sum_{T \subseteq S_{0r}} |T\rangle \langle T| + |j_{ir}\rangle \langle j_{ir}| \otimes \sum_{T \subseteq S_{1r}} |T\rangle \langle T|.
\]

This measurement succeeds with probability \( 2^h/u \), but if it succeeds we have the state corresponding to the answer to query \( |Q_{ir}\rangle \), from which we can predict \( x_i \). Putting everything together, we succeed with probability \( (2^h/u) \cdot (2/c) \), and if we succeed, we output \( x_i \) with probability \( 1/2 + \varepsilon \).

We can avoid failures by taking many copies of \( |U(x)\rangle \):

**Lemma 4** If \( C: \{0,1\}^n \rightarrow (\{0,1\}^\ell)^m \) is a \((1, c, \varepsilon)\)-quantum smooth code, then \( |W(x)\rangle = |U(x)\rangle^{\otimes cu}/2^{b+1} \) is a \( cu(\log(m) + \log(u))/2^{b+1} \)-qubit random access code for \( x \) with recovery probability \( 1/2 + \varepsilon/2 \) where \( u = \sum_{i=0}^b \binom{\ell}{i} \).

**Proof.** We do the experiment of the previous lemma on each copy of \( |U(x)\rangle \) independently. The probability that each experiment fails is \( (1 - 2^{b+1}/(cu))^{cu/2^{b+1}} \leq 1/2 \). In that case we output a fair coin flip. If at least one experiment succeeds, we can predict \( x_i \) with probability \( 1/2 + \varepsilon \). This gives overall success probability at least \( 1/2 \left( \frac{1}{2} + \varepsilon \right) \left( \frac{1}{2} \right)^2 = \frac{1}{2} + \frac{\varepsilon}{2} \).

Armed with these tricks, we can finally prove the lower bound for 2-query smooth codes and LDCs over non-binary alphabets.

**Theorem 6** If \( C: \{0,1\}^n \rightarrow \Sigma^m = (\{0,1\}^\ell)^m \) is a \((2, c, \varepsilon)\)-smooth code where the decoder uses only \( b \) bits of each answer, then

\[
m \geq 2^{d_n - \log(u)}
\]

for \( d = (1 - H(1/2 + \varepsilon/2^{b+1}))2^{b+1}/(cu) = \Theta(\varepsilon^2/(2^{b}cu)) \) and \( u = \sum_{i=0}^b \binom{\ell}{i} \). In particular, \( m = 2^{\Omega(\varepsilon^2n/(2^{b}c))} \) if \( b = \ell \).
Proof. Theorem 4 implies that $C$ is a $(1,c,\varepsilon/2^b)$-quantum smooth code. Lemma 4 gives us a random access code of $\log(m)+\log(u))/2^{b+1}$ qubits with recovery probability $p = 1/2 + \varepsilon/2^{b+1}$. Finally, the random access code lower bound, Theorem 5, implies $\log(m)+\log(u))/2^{b+1} \geq (1 - H(p))n$. Rearranging and using that $1 - H(1/2 + \eta) = \Theta(\eta^2)$ gives the result. \hfill \Box

Since a $(2,\delta,\varepsilon)$-LDC is a $(2,2/\delta,\varepsilon)$-smooth code (Theorem 1), we obtain the main result:

**Corollary 1** If $C : \{0,1\}^n \to \Sigma^m = (\{0,1\}^\ell)^m$ is a $(2,\delta,\varepsilon)$-locally decodable code, then

$$m \geq 2^{d n - \log(u)}$$

for $d = (1 - H(1/2 + \varepsilon/2^{b+1}))\delta^b / u = \Theta(\delta^2 / (2^b u))$ and $u = \sum_{i=0}^b \binom{\ell}{i}$. In particular, $m = 2^{\Omega(\delta^2 n/2^{2l})}$ if $b = \ell$.

In all known non-trivial constructions of LDCs and smooth codes, the decoder outputs the parity of the bits that it’s interested in. In this case we can prove a slightly stronger bound.

**Theorem 7** If $C : \{0,1\}^n \to \Sigma^m = (\{0,1\}^\ell)^m$ is a $(2,c,\varepsilon)$-smooth code where the decoder outputs $f(g(a_0, s_0), g(a_1, s_1))$, with $f : \{0,1\}^2 \to \{0,1\}$ and $g : \{0,1\}^b \to \{0,1\}$ fixed functions, then

$$m \geq 2^{d n - \log(\ell')}$$

for $d = \Omega(\varepsilon^2/(c\ell'))$ and $\ell' = \binom{\ell}{b}$.\hfill \Box

**Proof.** We can transform $C$ into a smooth code $C' : \{0,1\}^n \to (\{0,1\}^\ell)^m$ with $\ell' = \binom{\ell}{b}$ by defining $C'(x)_j$ to be the value of $g$ on all $\binom{\ell}{b}$ possible $b$-subsets of the original $\ell$ bits of $C(x)$. Now we’re interested in $\ell' = 1$ bit of each $C'(x)_j$. The result then follows from Theorem 6.\hfill \Box

## 5 Lower Bounds for Private Information Retrieval

### 5.1 Lower bounds for 2-server PIRs that use few bits

Here we derive improved lower bounds for 2-server PIRs from our LDC bounds. We use the following lemma from Goldreich et al. [6, Lemma 7.1] to translate PIR schemes to smooth codes.

**Lemma 5 (GKST)** Suppose there is a one-round, $(1-\eta)$-secure PIR scheme with two servers, database size $n$, query size $t$, answer size $\ell$, and recovery probability at least $1/2+\varepsilon$. Then there is a $(2,3,\varepsilon-\eta)$-smooth code $C : \{0,1\}^n \to (\{0,1\}^\ell)^m$, where $m \leq 6 \cdot 2^t$. If the PIR scheme uses only $b$ bits of each server answer, then the resulting smooth code uses only $b$ bits of each query answer.

We now combine this lemma with Theorem 6 to obtain the following theorem. This slightly improves the lower bound given in [9] and extends it to the case where we only use $b$ bits out of each server reply.

**Theorem 8** A classical 2-server $(1-\eta)$-secure PIR scheme with $t$-bit queries, $\ell$-bit answers that uses $b$ bits and has recovery probability $1/2+\varepsilon$ satisfies

$$t = \Omega \left( \frac{n(\varepsilon-\eta)^2}{2^tu} \right)$$

with $u = \sum_{i=0}^b \binom{\ell}{i}$. In particular, if all bits of the answer are used, then $t = \Omega(n(\varepsilon-\eta)^2/2^{2t})$.\hfill \Box
Proof. Using Lemma 5 we turn the PIR scheme into a $(2, 3, \varepsilon - \eta)$-smooth code $C : \{0, 1\}^n \rightarrow (\{0, 1\}^\ell)^m$ that uses $b$ bits of $\ell$ where $m \leq 6 \cdot 2^t$. From Theorem 6 we have $m \geq 2^{d - a}$ with $d = \Theta((\varepsilon - \eta)^2 / (2^k u))$. \hfill \square

If $b$ is fixed, $\varepsilon = 1/2$ and $\eta = 0$, the above bound simplifies to $t = \Omega(n/\ell)$, hence

**Corollary 2** A 2-server PIR scheme with $t$-bit queries and $\ell$-bit answers has total communication

$$C = 2(t + \ell) = \Omega\left(n^{1/\ell}\right).$$

For $b = 1$ this gives $C = \Omega(\sqrt{n})$, which is achieved by the square scheme of Section 2.5.1. For $b = 3$ we get $C = \Omega(n^{1/3})$, which is close to the $C = O(n^{1/3})$ of the cube scheme of Section 2.5.2.

As in Theorem 7, we can get slightly better bounds for PIR schemes where the user just outputs the parity of $b$ bits from each answer. All known non-trivial PIR schemes have this property.

**Corollary 3** If the PIR’s user outputs $f(g(a_0|S_0), g(a_1|S_1))$, for fixed $f$ and $g$, then

$$t = \Omega\left(n(\varepsilon - \eta)^2\right).$$

### 5.2 Weak lower bounds for general 2-server PIR

The previous lower bounds on the query length of 2-server PIR schemes were significant only for protocols that use few bits from each answer. Here we slightly improve the best known bound of $4.4 \log n$ [9] on the overall communication complexity of 2-server PIR schemes, by combining our Theorem 8 and Theorem 6 of Katz and Trevisan [8]. We restate their theorem for the PIR setting, assuming for simplicity that $\varepsilon = 1/2$ and $\eta = 0$.

**Theorem 9** (Katz & Trevisan) Every 2-server PIR scheme with $t$-bit queries and $\ell$-bit answers has

$$t \geq 2 \log \frac{n}{\ell} - O(1).$$

We now prove the following lower bound on the total communication $C = 2(t + \ell)$ of any 2-server PIR scheme with $t$-bit queries and $\ell$-bit answers:

**Theorem 10** Every 2-server PIR scheme has total communication

$$C \geq (5 - o(1)) \log n.$$

Proof. We distinguish three cases, depending on the answer length. Let $\delta = \log \log n / \log n$.

**case 1:** $\ell \leq (0.5 - \delta) \log n$. Then from Theorem 8 we get that $C \geq t = \Omega(n^{2\delta}) = \Omega((\log n)^2)$.

**case 2:** $(0.5 - \delta) \log n < \ell < 2.5 \log n$. Then from Theorem 9 we have

$$C = 2(t + \ell) > 2(2 \log(n/(2.5 \log n)) - O(1) + (0.5 - \delta) \log n) = (5 - o(1)) \log n.$$

**case 3:** $\ell \geq 2.5 \log n$. Then $C = 2(t + \ell) \geq 5 \log n.$ \hfill \square
5.3 Quantum PIRs from classical PIR with non-binary answers

Using the tricks employed for LDCs above, we can construct a 2-server quantum PIR scheme from a 4-server classical PIR scheme that uses $b$ bits, as follows. The user flips his randomness as in the classical scheme. This fixes queries $q_0, q_1, q_2, q_3$ as well as sets $S_0, S_1, S_2, S_3$ of $b$-bit indices to use from answers $a_0, a_1, a_2, a_3$ respectively. He now picks random permutations $\pi_1, \pi_2, \pi_3$ on the set of all $\binom{\ell}{b}$ $b$-element subsets from an $\ell$-element set, such that $\pi_1(S_0) = S_1$, $\pi_2(S_0) = S_2$ and $\pi_3(S_0) = S_3$. The user then constructs the following quantum state

$$\frac{1}{\sqrt{\binom{\ell}{b}}} \sum_{|T|=b} |T\rangle \left( \frac{1}{\sqrt{2}}(|0\rangle|q_0\rangle|T\rangle + |1\rangle|q_1\rangle|\pi_1(T)\rangle + |2\rangle|q_2\rangle|\pi_2(T)\rangle + |3\rangle|q_3\rangle|\pi_3(T)\rangle \right).$$

He keeps the first two registers to himself and sends the rest to the two quantum servers as indicated. Each server now only sees a random mixture over the classical queries and a random $T$. Thus the privacy of the 4-server scheme carries over to the quantum scheme. Each server tags on $b$ $0\rangle$-qubits, maps $|j\rangle|q_j\rangle|T\rangle 0^b \mapsto |j\rangle|q_j\rangle|T\rangle|a_j\rangle\langle a_j|\langle T|$ and sends everything back. The user now measures the first register, hoping to obtain $T = S_0$. He succeeds with probability $1/\binom{\ell}{b}$. He can then unitarily remove the $S_j$ and $q_j$ to get

$$\frac{1}{\sqrt{2}}(|0\rangle|a_0|S_0\rangle a_1|S_1\rangle + |1\rangle|a_2|S_2\rangle a_3|S_3\rangle).$$

From this he can compute $f(a_0|S_0, a_1|S_1, a_2|S_2, a_3|S_3)$ with probability $1/2 + 1/2^{2b+1}$ using Theorem 2. The expected number of repetitions (in parallel) before success is $\binom{\ell}{b}$. This means that if we can construct 4-server classical PIR schemes where $\binom{\ell}{b}$ is quite small, then we obtain an efficient 2-server quantum PIR scheme. For example, if there exists a 4-server classical PIR scheme with $t, \ell = O(n^{1/8})$, using only $b = 1$ bits from each of the 4 answers, then we obtain a 2-server quantum PIR scheme with an expected number of $O(n^{1/4})$ qubits of communication and recovery probability close to 1. Currently, the best known 2-server quantum PIR communicates $O(n^{3/10})$ qubits [9].

6 Conclusion and Future Work

In this paper we improved the best known lower bounds on the length of 2-query locally decodable codes and the communication complexity of 2-server private information retrieval schemes. Our bounds are significant whenever the decoder uses only few bits from the two query answers, even if the alphabet (LDC case) or answer length (PIR case) is large. This contrasts with the earlier results of Kerenidis and de Wolf [9], which become trivial even for logarithmic alphabet or answer length, and those of Goldreich et al. [6], which only apply to linear schemes.

Still, general lower bounds without constraints on alphabet or answer size completely elude us. Clearly, this is one of the main open questions in this area. Barring that, we could at least improve the dependence on $b$ of our current bounds. For example, a PIR lower bound like $t = \Omega(n/\ell^{b/2})$ might be feasible using some additional quantum tricks. Such a bound for instance implies that the total communication is $\Omega(n^{1/3})$ for $b = 3$, which would show that the Cube scheme of Section 2.5.2 is optimal. Another question is to obtain strong lower bounds for the case of $k \geq 3$ queries or servers. For this case, no superpolynomial lower bounds are known even if the alphabet or answer size is only one bit. Finally, our constructions motivate the search for 4-server classical PIR schemes with
fairly large answer length \( \ell \), but using very few bits from each answer. As explained in Section 5.3, such schemes would give better 2-server quantum PIR schemes.

References

[1] A. Ambainis. Upper bound on communication complexity of private information retrieval. In Proceedings of the 24th ICALP, volume 1256 of Lecture Notes in Computer Science, pages 401–407, 1997.

[2] L. Babai, L. Fortnow, L. Levin, and M. Szegedy. Checking computations in polylogarithmic time. In Proceedings of 23rd ACM STOC, pages 21–31, 1991.

[3] A. Beimel and Y. Ishai. Information-theoretic private information retrieval: A unified construction. In Proceedings of 28th ICALP, pages 912–926, 2001. Longer version on ECCC.

[4] A. Beimel, Y. Ishai, E. Kushilevitz, and J. Raymond. Breaking the \( O(n^{1/(2k-1)}) \) barrier for information-theoretic Private Information Retrieval. In Proceedings of 43rd IEEE FOCS, pages 261–270, 2002.

[5] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. Journal of the ACM, 45(6):965–981, 1998. Earlier version in FOCS’95.

[6] O. Goldreich, H. Karloff, L. Schulman, and L. Trevisan. Lower bounds for linear locally decodable codes and private information retrieval. In Proceedings of 17th IEEE Conference on Computational Complexity, pages 175–183, 2002. Also on ECCC.

[7] R. A. Horn and C. R. Johnson. Matrix Analysis. Cambridge University Press, 1985.

[8] J. Katz and L. Trevisan. On the efficiency of local decoding procedures for error-correcting codes. In Proceedings of 32nd ACM STOC, pages 80–86, 2000.

[9] I. Kerenidis and R. de Wolf. Exponential lower bound for 2-query locally decodable codes via a quantum argument. Journal of Computer and Systems Sciences, 2004. Earlier version in STOC’03. quant-ph/0208062.

[10] E. Mann. Private access to distributed information. Master’s thesis, Technion - Israel Institute of Technology, Haifa, 1998.

[11] A. Nayak. Optimal lower bounds for quantum automata and random access codes. In Proceedings of 40th IEEE FOCS, pages 369–376, 1999. quant-ph/9904093.

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