A Nearly Tight Lower Bound for Restricted Private Information Retrieval Protocols

Richard Beigel*
Temple University

Lance Fortnow†
NEC Laboratories America

William Gasarch‡
University of Maryland at College Park

Abstract

We show that any 1-round 2-server Private Information Retrieval Protocol where the answers are 1-bit long must ask questions that are at least $n-2$ bits long, which is nearly equal to the known $n-1$ upper bound. This improves upon the approximately $0.25n$ lower bound of Kerenidis and de Wolf while avoiding their use of quantum techniques.

1 Introduction

Following prior papers on Private Information Retrieval Protocols ([1, 3, 4, 6, 9]) we model a database as an n-bit string $x = x_1 \dots x_n$. Suppose that the user wants to know x_i but does not want the database to obtain any information about i. We do not impose any computational limits on the database, though some researchers have considered such limits [3, 9]. If there is only one copy of the database then the only way to ensure privacy is to request the entire string x, which is n bits long. If there are $k \geq 2$ copies of the database that do not communicate with each other then the number of bits can be reduced. We refer to a copy of the database as a *server*.

Many upper bounds have been obtained. These include

1. If there are two servers then $O(n^{1/3})$ bits of communication suffice [4].
2. If there are k servers then $O(n^{1/(2k-1)})$ bits of communication suffice [1, 2].
3. If there are k servers then $n^{O(\log \log k / k \log k)}$ bits of communication suffice [2].

Lower bounds on Private Information Retrieval Protocols have been hard to obtain. The lower bounds that are known either limit the type of query [7, 10] or are weak [8, 11].

We assume throughout the paper that the queries sent to each server are the same length. Consider the case that the answers from the database are linear, i.e., they are an XOR of some subset of the bits of the database. Goldreich, Karloff, Schulman, and Trevisan [7] show that $\Omega(\frac{n}{2^a})$ bits must be sent to each server where a is the number of bits each server could send back to the user. The lower bound also holds for randomized protocols with a small probability of error. The multiplicative constant depends on the probability of error. In the special case of $a = 1$ where

*Temple University, Dept. of Computer and Information Sciences, 1805 N. Broad St. Philadelphia, PA 19122. beigel@cis.temple.edu

†Currently at Department of Computer Science, University of Chicago, 1100 E. 58th St., Chicago, IL 60637. fortnow@cs.uchicago.edu

‡University of Maryland, Dept. of Computer Science and Institute for Advanced Computer Studies, College Park, MD 20742. gasarch@cs.umd.edu

the user simply XORs the bits he gets, Chor, Kushilevitz, Goldreich and Sudan [4] show that any protocol would require $n-1$ bits sent to each server. They also give a matching upper bound in this model.

In the case that answers are not restricted to be linear, nontrivial lower bounds have only recently been discovered. Kerenidis and de Wolf [8] show that at least $\Omega(n/2^{5a})$ bits must be sent to each server. This has been improved to $\Omega(n/2^{2a})$ by Wehner and de Wolf [11]. In the case $a=1$ Kerendis and de Wolf show that at least $(1-H(11/14))n-4 \sim 0.25n$ bits are required. Their proof first converts a 2-server randomized protocol to a 1-server quantum protocol and then they show lower bounds on the quantum protocol. Hence their lower bounds hold for randomized protocols that allow a small probability of error.

In this paper we obtain a lower bound of $n-2$ for 2-server deterministic error-free PIR schemes with the assumption that the answers are 1-bit long. This nearly matching the $n-1$ upper bound of Chor, Kushilevitz, Goldreich and Sudan [4].

We avoid the quantum techniques used by Kerenidis and de Wolf. Rather our proof builds on classical tools developed by Yao [12] and Fortnow and Szegedy [5] for studying locally-random reductions, a complexity-theoretic tool for information hiding that predates private information retrieval.

2 The Lower Bound

In this section we formally define the model and state and prove our main result.

Definition 2.1 A *2-server 1-round r-random bit PIR for databases of size n with m-bit queries and a-bit answers* is a tuple $(q_1, q_2, a_1, a_2, \phi)$ such that the following hold.

1. $q_j : [n] \times \{0,1\}^r \to \{0,1\}^m$. This is the query sent to server j. The distribution of $q_j(i,\rho)$ is independent of i (this ensures privacy).

2. $a_j : \{0,1\}^n \times \{0,1\}^m \to \{0,1\}^a$. This is the response server j gives if the database is $x \in \{0,1\}^n$ and he sees query $\mu \in \{0,1\}^m$.

3. $\phi : [n] \times \{0,1\}^r \times \{0,1\}^m \times \{0,1\}^m \times \{0,1\}^a \times \{0,1\}^a \to \{0,1\}$. This is how the user puts together the information he has received. Say he wants to know x_i. If the random string is $\rho \in \{0,1\}^r$ and the queries are q_1, q_2, and he gets back a-bit strings b_1 and b_2 then the user computes $x_i = \phi(i, \rho, q_1, q_2, b_1, b_2)$. (Note that since q_1 and q_2 are functions of i, ρ we could have defined ϕ to be a function of just (i, ρ, b_1, b_2); however, making q_1, q_2 explicit inputs has notational advantages.)

Assume that $(q_1, q_2, a_1, a_2, \phi)$ is a 2-server 1-round r-random bit PIR for databases of size n with m-bit queries and 1-bit answers. Imagine that the user wants to find x_i, has random string ρ, and has found out $a_1(x, q_1(i,\rho))$. It is possible that $a_2(x, q_2(i,\rho))$ is not needed. This would happen if $a_2(x, q_2(i,\rho)) = 0$ and $a_2(x, q_2(i,\rho)) = 1$ yield the same value for x_i. If this happens then we say that $i, \rho, a_1(x, q_1(i,\rho))$ *set* x_i. It is also possible that $a_2(x, q_2(i,\rho))$ is crucial. In this case, if the user happened to know x_i he could determine $a_2(x, q_2(i,\rho))$. In this case we say that $i, \rho, a_1(x, q_1(i,\rho))$ and x_i *force* $a_2(x, q_2(i,\rho))$. Either way is a win. The next definition and lemma formalize this notion.

For the next definition and the two lemmas following it let $(q_1, q_2, a_1, a_2, \phi)$ be a 2-server 1-round r-random bit PIR for databases of size n with m-bit queries and 1-bit answers.

Definition 2.2 Let $i \in [n]$, $\rho \in \{0,1\}^r$, and $x \in \{0,1\}^n$.

1. The values of $i, \rho, a_1(x, q_1(i,\rho))$ *set* x_i if

$$\phi(i, \rho, q_1(i,\rho), q_2(i,\rho), a_1(x, q_1(i,\rho)), 0) = \phi(i, \rho, q_1(i,\rho), q_2(i,\rho), a_1(x, q_1(i,\rho)), 1).$$

   Note that if the user knows i, ρ, and $a_1(x, q_1(i,\rho))$ then he knows x_i. This is a win. The values of $i, \rho, a_2(x, q_2(i,\rho))$ *set* x_i can be defined similarly.

2. We say the values of $i, \rho, a_1(x, q_1(i,\rho))$, and x_i *force* $a_2(x, q_2(i,\rho))$ if

$$\phi(i, \rho, q_1(i,\rho), q_2(i,\rho), a_1(x, q_1(i,\rho)), 0) \neq \phi(i, \rho, q_1(i,\rho), q_2(i,\rho), a_1(x, q_1(i,\rho)), 1).$$

   Note that if the user knows $i, \rho, a_1(x, q_1(i,\rho))$ and x_i then he knows $a_2(x, q_2(i,\rho))$. This is also a win. The values of $i, \rho, a_2(x, q_2(i,\rho))$, and x_i *force* $a_1(x, q_1(i,\rho))$ is defined similarly.

We need the restriction of 1-bit back answers in order to define set and force. The next Lemma uses these notions. It is the only place we use that the answers are 1 bit long. Any attempt to extend our proof to 2 or more bits will have to get around this obstacle.

The following lemma follows from the Definition 2.2

Lemma 2.3 *Let $i \in [n]$, $\rho \in \{0,1\}^r$, and $x \in \{0,1\}^n$. Then both of the following hold:*

1. *Either $i, \rho, a_1(x, q_1(i,\rho))$ set x_i or $i, \rho, a_1(x, q_1(i,\rho))$, and x_i force $a_2(x, q_2(i,\rho))$.*
2. *Either $i, \rho, a_2(x, q_2(i,\rho))$ set x_i or $i, \rho, a_2(x, q_2(i,\rho))$, and x_i force $a_1(x, q_1(i,\rho))$.*

In Lemma 2.5 and Theorem 2.6 we will use the mythical character Alice. Alice is computationally unbounded and knows the protocol but she does not know x. We will be concerned with what she can and cannot deduce from other information she is given.

Notation 2.4

1. Let ORD_1 (ORD_2) be a set of ordered pairs of queries to the first (second) server and the answers to these queries. The phrase *'Alice can deduce x_i from ORD_1, ORD_2, and i'* means that Alice, who has unlimited power and access to the protocol, can determine a value $b \in \{0,1\}$ such that $x_i = b$ is consistent with her data while $x_i \neq b$ is not.
2. We can define a similar notion of deduce for other information Alice may have. For example, it is possible that if Alice knows some x_i and some query answers she can deduce other query answers (see Definition 2.2).

Lemma 2.5 *Let $x \in \{0,1\}^n$. Let S^1, S^2 be multisets of $\{0,1\}^m$. Assume that, for every $q_1 \in S^1$ Alice knows $a_1(x, q_1)$; and, for every $q_2 \in S^2$ Alice knows $a_2(x, q_2)$. Let*

$INFO = \{(q_1, a_1(x, q_1)) \mid q_1 \in S^1\} \cup \{(q_2, a_2(x, q_2)) \mid q_2 \in S^2\}$

Note that Alice knows the set INFO. Assume that i_0 is such that Alice cannot deduce x_{i_0} from i_o and INFO. Let T^1 and T^2 be the following multisets.

$$T^1 = \{q_1(i_0, \rho)) \mid q_2(i_0, \rho) \in S^2\}$$
$$T^2 = \{q_2(i_0, \rho)) \mid q_1(i_0, \rho) \in S^1\}.$$

Then

1. *Assume x_{i_0} and INFO are known to Alice. For every $q_1 \in T_1$ Alice can deduce $a_1(x, q_1)$; and, for every $q_2 \in T_2$ Alice can deduce $a_2(x, q_2)$.*

2. *$|T^1| = |S^2|$ and $|T^2| = |S^1|$.*

3. *$|(S^1 \cup T^1) \cup (S^2 \cup T^2)| = 2|S^1 \cup S^2|$. (These are multisets.)*

Proof:
1) Let $q_1(i_0, \rho) \in T^1$. By Lemma 2.3 either $i_0, \rho, a_2(x, q_2(i_0, \rho))$ *set* x_{i_0} or $i_0, \rho, a_2(x, q_2(i_0, \rho))$, and x_{i_0} *force* $a_1(x, q_1(i_0, \rho))$. Since $q_2(i_0, \rho) \in S^2$ and Alice cannot deduce x_{i_0} from the information, the former cannot happen. Hence the later happens. Hence, knowing x_{i_0} and the information Alice can deduce $a_1(x, q_1(i_0, \rho))$. A similar proof holds for $q_2(i_0, \rho) \in T^2$.

2) There is a bijection between the multiset S^2 and the multiset T^1: map $q_2(i_0, \rho)$ to $q_1(i_0, \rho)$. Hence $|T^1| = |S^2|$. Similar for $|T^2| = |S^1|$.

3) This follows from part 2. ∎

Theorem 2.6 *Any 2-server 1-round r-random bit PIR for databases of size n with m-bit queries and 1-bit answers must have $m \geq n - 2$.*

Proof:

The following theorem was originally proven using Kolmogorov Complexity; however, we have rephrased the proof in terms of simple combinatorics.

Let $(q_1, q_2, a_1, a_2, \phi)$ be a 2-server 1-round r-random bit PIR for databases of length n with m-bit queries and 1-bit answers.

Let M_1 and M_2 be the following multisets of $\{0,1\}^m$.

$$M_1 = \{q_1(1, \rho) \mid \rho \in \{0,1\}^r\}$$
$$M_2 = \{q_2(1, \rho) \mid \rho \in \{0,1\}^r\}.$$

By privacy, for all i,

$$M_1 = \{q_1(i, \rho) \mid \rho \in \{0,1\}^r\}$$
$$M_2 = \{q_2(i, \rho) \mid \rho \in \{0,1\}^r\}.$$

Fix ρ. For every $i \in [n]$ there exists ρ', ρ'' such that $q_1(1, \rho) = q_1(i, \rho')$ and $q_2(1, \rho) = q_2(i, \rho'')$.

We exhibit an injection $f : \{0,1\}^n \to \{0,1\}^{m+2}$, hence we obtain $n \leq m+2$, so $m \geq n-2$. The proof that f is an injection will follow easily from the fact that from $f(x)$ and the protocol Alice can reconstruct x.

Since $|M_1| = 2^r$ and the total number of distinct strings is at most 2^m there must be a string that occurs with multiplicity 2^{r-m}. Let μ_0 be such a string. For notational convenience we assume

$$\mu_0 = q_1(1, \rho_1) = q_1(1, \rho_2) = \cdots = q_1(1, \rho_{2^{r-m}}).$$

We describe a process for generating a (short) string we call $ADVICE$ that will begin with $a_1(x, \mu_0)$ but then have several bits of x. From $ADVICE$ we will be able to reconstruct the entire string x. We will end up taking $f(x)$ to be $ADVICE$ padded with 0's to make it the right length. *Intuition:* At the end of stage ℓ we will have the following.

1. A set $I_\ell \subseteq [n]$.

2. A string $ADVICE_\ell$ which is the concatentaion of x_i for $i \in I_\ell$. The ideas is that sometimes we will use the bits in $ADVICE_\ell$ to deduce answers to queries, and sometimes we will use answers to queries to deduce values x_j where $j \notin I_\ell$.

3. A multiset $S^1_\ell \subseteq M_1$, and a multiset $S^2_\ell \subseteq M_2$.

4. Given $ADVICE_\ell$, the protocol, and the construction so far, Alice will be able to deduce the following.

   (a) For every $q_1 \in S^1_\ell$, $a_1(x, q_1)$.

   (b) For every $q_2 \in S^2_\ell$, $a_2(x, q_2)$.

   (c) For every $i \in I$, x_i.

These answers will enable Alice to deduce some values of x_i. If x_{i_0} cannot be deduced then adding x_{i_0} to the advice will double the number of strings in $M_1 \cup M_2$ for which Alice can deduce the answers and thus get $|S^1_{\ell+1} \cup S^2_{\ell+1}| = 2|S^1_\ell \cup S^2_\ell|$.

We now give the formal construction.

1. Let $ADVICE_0 = a_1(x, \mu_0)$. Throughout the construction $ADVICE_\ell \in \{0,1\}^*$ will be $a_1(x, \mu_0)$ followed by a string of bits that represent particular x_i values. We do not need to put i's into the advice as they will be deduced from the construction.

2. Let S^1_0 be the multiset $\{\mu_0, \ldots, \mu_0\}$ (2^{r-m} μ_0's). Formally we define the multiset S^1_0 as

$$S^1_0 = \{q_1(1, \rho_1), q_1(1, \rho_2), \ldots, q_1(1, \rho_{2^{r-m}})\}.$$

   Let $S^2_0 = \emptyset$.

3. Let $I_0 = \emptyset$. Throughout the construction $I_\ell \subseteq [n]$ will be the set of indices i such that Alice can deduce x_i from knowing the answers to the queries in $S^1_\ell \cup S^2_\ell$.

4. Assume S^1_ℓ, S^2_ℓ have been constructed and $I_\ell \neq [n]$. Let i_0 be the least element of $[n] - I_\ell$.

   (a)
$$ADVICE_{\ell+1} = ADVICE_\ell \cdot x_{i_0}.$$

   (b) We will now add strings to S^1_ℓ (S^2_ℓ) to obtain $S^1_{\ell+1}$ ($S^2_{\ell+1}$). When adding strings to a multiset you need to be careful. We will use the notation $q(i, \rho) \notin S^1_\ell$. This may be true even if the string $q(i, \rho)$ is in S^1_ℓ. What we mean is that it was never put into S^1_ℓ explicitly *as* $q(i, \rho)$. For example, if $q(2, 0010) = q(7, 1101)$ then we may have earlier put $q(7, 1101)$ into S^1_ℓ; however we would still say $q(2, 0010) \notin S^1_\ell$.

$$S^1_{\ell+1} = S^1_\ell \cup \{q_1(i_0, \rho) \mid q_1(i_0, \rho) \notin S^1_\ell \wedge q_2(i_0, \rho) \in S^2_\ell\}$$
$$S^2_{\ell+1} = S^2_\ell \cup \{q_2(i_0, \rho) \mid q_2(i_0, \rho) \notin S^2_\ell \wedge q_1(i_0, \rho) \in S^1_\ell\}$$

   By Lemma 2.5 $|S^1_{\ell+1} \cup S^2_{\ell+1}| = 2|S^1_\ell \cup S^2_\ell|$.

   (c)

$$\begin{aligned} I_{\ell+1} = \; & I_\ell \cup \\ & \{j \mid (\exists \rho)[q_1(j, \rho) \in S^1_{\ell+1} \wedge j, \rho, a_1(x, q_1(j, \rho)), x_j \text{ force } a_2(x, q_2(j, \rho))\} \cup \\ & \{j \mid (\exists \rho)]q_2(j, \rho) \in S^2_{\ell+1} \wedge j, \rho, a_2(x, q_2(j, \rho)), x_j \text{ force } a_1(x, q_2(j, \rho))\}. \end{aligned}$$

5. If $I_\ell = [n]$ then terminate. If $I_\ell \neq [n]$ then set $\ell = \ell + 1$ and goto step 4. Note that if $S^1_\ell \cup S^2_\ell = M_1 \cup M_2$ then $I_\ell = [n]$ and the construction will terminate.

Since $|S^1_0 \cup S^2_0| = 2^{r-m}$ and this union doubles with every stage, $|S^1_\ell \cup S^2_\ell| = 2^{r-m+\ell}$. Let ℓ' be the final value of ℓ. Since $|M_1 \cup M_2| = 2^{r+1}$ and $S^1_\ell \cup S^2_\ell \subseteq M_1 \cup M_2$, $r - m + \ell' \leq r + 1$ so $\ell' \leq m+1$. Since $ADVICE$ began with one additional bit, $|ADVICE| \leq \ell' + 1 \leq m + 2$. Let $f(x)$ be $ADVICE$ followed by enough 0's to pad it out to length $m + 2$. This padding does not affect the reconstruction of x from $f(x)$ since the advice produced for different x's is prefix free. ∎

3 Open Problems

Chor, Kushilevitz, Goldreich and Sudan [4] showed that, there is a 2-server 1-round n-random bit PIR for databases of size n with $n - 1$ bit queries and 1-bit answers. By combining this with a general communication balancing technique (also from [4]) one can obtain the following:

Theorem 3.1 *Fix $n \in \mathsf{N}$. Let a be such that $a < n$. There exists a 2-server 1-round $(\lceil n/a \rceil - 1)$-random bit PIR for databases of size n with $(\lceil n/a \rceil - 1)$-bit queries and a-bit answers.*

Our lower bound showed that this upper bound is tight in the $a = 1$ case up to an additive constant. It is an open question to show this for all constant a or even for $a = 2$.

4 Acknowledgments

We would like to thank Jonathan Katz for pointing out that our original proof could be rephrased in terms of simple combinatorics rather than Kolmogorov Theory. We would also like to thank Ronald de Wolf for helpful commentary and updates on his paper with Kerenidis. Thanks to Umesh Vazirani and Stephanie Wehner for helpful discussions and Nan Wang and the anonymous referee for proofreading.

References

[1] A. Ambainis. Upper bound on the communication complexity of private information retrieval. In *Proc. of the 24th ICALP*, 1997.

[2] A. Beimel, Y. Ishai, E. Kushilevitz, and J.-F. Rayomnd. Breaking the $o(n^{1/(2k-1)})$ barrier for information-theoretic private information retrieval. In *Proc. of the 43st IEEE Sym. on Found. of Comp. Sci.*, 2002.

[3] C. Cachin, S. Micali, and M. Stadler. Computationally private information retrieval with polylog communication. In *EUROCRYPT99*, 1999.

[4] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan. Private information retrieval. *Journal of the ACM*, 45, 1998. Earlier version in FOCS 95.

[5] L. Fortnow and M. Szegedy. On the power of two-local random reductions. *Information Processing Letters*, 44:303–306, 1992.

[6] W. Gasarch. A survey on private information retrieval. *Bulletin of the European association of theoretical computer science(BEATCS)*, 82:84–102, 2004. Also see website on this topic: www.cs.umd.edu/~gasarch/pir/pir.html.

[7] O. Goldreich, H. Karloff, L. Schulman, and L. Trevisan. Lower bounds for linear local decodable codes and private information retrieval systems. In *Proc. of the 17th IEEE Conf on Complexity Theory*. IEEE Computer Society Press, 2002.

[8] I. Kerenidis and R. de Wolf. Exponential lower bound for 2-query locally decodable codes. *Journal of Computer and Systems Sciences*, pages 395–420, 2004. Earlier version in STOC03. Electronic version at arxiv.org/abs/quant-ph/0403140.

[9] E. Kushilevitz and R. Ostrovsky. Replication is not needed: Single database, computationally-private information retrieval (extended abstract). In *Proc. of the 38st IEEE Sym. on Found. of Comp. Sci.*, pages 364–373, 1997.

[10] E. Mann. *Private access to distributed information*. PhD thesis, Technion – Israel Institute of Technology, Haifa, 1998. Masters Thesis.

[11] S. Wehner and R. de Wolf. Improved lower bounds for locally decodable codes and private information retrieval, 2004. arxiv.org/abs/quant-ph/0403140.

[12] A. Yao. An application of communication complexity to cryptography, 1990. Lecture DIMACS Workshop on Structural Complexity and Cryptography.