A Nearly Tight Lower Bound for 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, 8]) 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, 8]. 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. Lower bounds are only known for 2-server protocols with one round and restrictions on the number of bits returned by the servers. Even then, prior to Kerenidis and de Wolf [7] all lower bounds had restrictions on the type of answers the servers could return.

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 [6] 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

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

[†] NEC Laboratories America, 4 Independence Way, Princeton, NJ 08540. fortnow@nec-labs.com

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

multiplicative constant depends on the probability of error. In the special case of $a = 1$ where 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 [7] show that at least $\Omega(n/2^{6a})$ bits must be sent to each server. In the case $a = 1$ they 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 there lower bounds hold for randomized protocols that allow a small probability of error.

In this paper we obtain a lower bound of $n - 2$ with the assumption that the answers are 1-bit long, 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 [9] 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 $(q_1, q_2, ANS_1, ANS_2, \phi)$ 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. $ANS_j : \{0,1\}^n \times \{0,1\}^m \to \{0,1\}^a$. This is the response server j gives if the database is $x \in \{0,1\}^n$ and he sees answer $\mu \in \{0,1\}^m$.

3. $\phi : [n] \times \{0,1\}^r \times \{0,1\}^a \times \{0,1\}^a \to \{0,1\}$. This is how the user puts together the information he has received. If he wants to know x_i and the random string is $\rho \in \{0,1\}^r$, and he gets back a-bit strings b_1 and b_2 then the user computes $x_i = \phi(i, \rho, b_1, b_2)$.

Imagine that the user wants to find x_i, has random string ρ, and has found out $ANS_1(x, q_1(i, \rho))$. It is possible that $ANS_2(x, q_2(i, \rho))$ is not needed. This would happen if $ANS_2(x, q_2(i, \rho)) = 0$ and $ANS_2(x, q_2(i, \rho)) = 1$ yield the same value for x_i. If this happens then we say that $i, \rho, ANS_1(x, q_1(i, \rho))$ set x_i. It is also possible that $ANS_2(x, q_2(i, \rho))$ is crucial. In this case, if the user happened to know x_i he could determine $ANS_2(x, q_2(i, \rho))$. In this case we say that $i, \rho, ANS_1(x, q_1(i, \rho))$ and x_i set $ANS_2(x, q_2(i, \rho))$. Either way is a win. The next definition and lemma formalize this notion.

For the next definition and the two lemmas following it let $(q_1, q_2, ANS_1, ANS_2, \phi)$ 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 $i, \rho, ANS_1(x, q_1(i, \rho))$ *set* x_i if

$$\phi(i, \rho, q_1(i, \rho), q_2(i, \rho), ANS_1(x, q_1(i, \rho)), 0) = \phi(i, \rho, q_1(i, \rho), q_2(i, \rho), ANS_1(x, q_1(i, \rho)), 1).$$

Note that if the user knows i, ρ, and $ANS_1(x, q_1(i,\rho))$ then he knows x_i. This is a win. The values of $i, \rho, ANS_2(x, q_2(i,\rho))$ *set* x_i can be defined similarly.

2. We say the values of $i, \rho, ANS_1(x, q_1(i,\rho))$, and x_i *force* $ANS_2(x, q_2(i,\rho))$ if

$$\phi(i, \rho, q_1(i,\rho), q_2(i,\rho), ANS_1(x, q_1(i,\rho)), 0) \neq \phi(i, \rho, q_1(i,\rho), q_2(i,\rho), ANS_1(x, q_1(i,\rho)), 1).$$

Note that if the user knows $i, \rho, ANS_1(x, q_1(i,\rho))$ and x_i then he knows $ANS_2(x, q_2(i,\rho))$. This is also a win. The values of $i, \rho, ANS_2(x, q_2(i,\rho))$, and x_i *force* $ANS_1(x, q_1(i,\rho))$ are defined similarly. This definition of force is the main place we use that we get back 1-bit answers.

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 $i, \rho, ANS_1(x, q_1(i,\rho))$ set x_i or $i, \rho, ANS_1(x, q_1(i,\rho))$, and x_i force $ANS_2(x, q_2(i,\rho))$.*
2. *Either $i, \rho, ANS_2(x, q_2(i,\rho))$ set x_i or $i, \rho, ANS_2(x, q_2(i,\rho))$, and x_i force $ANS_1(x, q_1(i,\rho))$.*

Note 2.4 The only place we use that the answers are 1 bit long is in Lemma 2.3. Any attempt to extend our proof to 2 or more bits will have to get around this obstacle.

Lemma 2.5 *Let $x \in \{0,1\}^n$. Let S^1, S^2 be multisets of $\{0,1\}^m$. Assume that, for every $q_1 \in S^1$ we know $ANS_1(x, q_1)$; and, for every $q_2 \in S^2$ we know $ANS_2(x, q_2)$. We call this "the information." Assume that x_{i_0} is such that we cannot deduce x_{i_0} from the information. 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 the information are known. For every $q_1 \in T_1$ we can deduce $ANS_1(x, q_1)$; and, for every $q_2 \in T_2$ we can deduce $ANS_2(x, q_2)$.*
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: Let $q_1(i_0, \rho) \in T^1$. By Lemma 2.3 either $i_0, \rho, ANS_2(x, q_2(i_0,\rho))$ *set* x_{i_0} or $i_0, \rho, ANS_2(x, q_2(i_0,\rho))$, and x_{i_0} *force* $ANS_1(x, q_1(i_0,\rho))$. Since $q_2(i_0, \rho) \in S^2$ and we cannot deduce x_{i_0} from the information, the former cannot happen. Hence the later happens. Hence, knowing i_0 and the information we can deduce $ANS_1(x, q_1(i_0,\rho))$. A similar proof holds for $q_2(i_0, \rho) \in T^2$.

For every element in the multiset S^2 we put an element into T^1. Hence $|T^1| = |S^2|$. Similar for $|T^2| = |S^1|$. ∎

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 $(q_1, q_2, ANS_1, ANS_2, \phi)$ be a 2-server 1-round r-random bit PIR for databases of length n with m-bit queries and a 1-bit answers.

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

$$\begin{aligned} M_1 = &\ \{q_1(1,\rho) \mid \rho \in \{0,1\}^r\} \\ M_2 = &\ \{q_2(1,\rho) \mid \rho \in \{0,1\}^r\} \end{aligned}$$

By privacy, for all i,

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

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 $n \le m+2$, so $m \ge n-2$. The proof that f is an injection will follow easily from the fact that from $f(x)$ and the protocol you 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 that 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 $ANS_1(x,\mu_0)$ but then have several bits of x. From $ADVICE$ we will be able to reconstruct the entire string x.

Intuition: At the end of stage ℓ we will have a string $ADVICE_\ell$, multiset $S^1_\ell \subseteq M_1$, and multiset $S^2_\ell \subseteq M_2$. For every $q_1 \in S^1_\ell$ we will be able to recover $ANS_1(x,q_1)$; and for every $q_2 \in S^2_\ell$ we will be able to recover $ANS_2(x,q_2)$. These answers will enable us to recover some values of x_i. If x_{i_0} cannot be recovered then adding x_{i_0} to the advice will double the number of strings in $M_1 \cup M_2$ for which we know 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 $ADVICE_0 = ANS_1(x,\mu_0)$. Throughout the construction $ADVICE_\ell \in \{0,1\}^*$ will be $ANS_1(x,\mu_0)$ 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 recovered from the construction.

2. Let S^1_0 be the multiset $\{q_1(1,\rho_1), q_1(1,\rho_2), \ldots, q_1(1,\rho_{2^{r-m}})\}$. 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 we 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 $I_\ell \ne [n]$. Let i_0 be the least element of $[n] - I_\ell$.

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

   (b)
   $$\begin{aligned} S^1_{\ell+1} = &\ S^1_\ell \cup \{q_1(i_0,\rho)) \mid q_2(i_0,\rho) \in S^2_\ell\} \\ S^2_{\ell+1} = &\ S^2_\ell \cup \{q_2(i_0,\rho)) \mid q_1(i_0,\rho) \in S^1_\ell\} \end{aligned}$$

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

(c)

$$\begin{array}{rl} I_{\ell+1} = & I_\ell \cup \\ & \{j \mid (\exists (q_1(j,\rho)) \in S_{\ell+1})[j,\rho,ANS_1(x,q_1(j,\rho)),x_j \text{ force } ANS_2(x,q_2(j,\rho))\} \cup \\ & \{j \mid (\exists (q_2(j,\rho)) \in S_{\ell+1})[j,\rho,ANS_2(x,q_2(j,\rho)),x_j \text{ force } ANS_1(x,q_2(j,\rho))\}. \end{array}$$

5. If $I_\ell = [n]$ then terminate. If $I_\ell \neq [n]$ then set $\ell = \ell + 1$ and goto step 4.

Since $|S_0^1 \cup S_0^2| = 2^{r-m}$ and this union doubles with every stage, we have $|S_\ell^1 \cup S_\ell^2| = 2^{r-m+\ell}$. Let ℓ' be the final value of ℓ. Since $|M_1 \cup M_2| = 2^{r+1}$ and $S_\ell^1 \cup S_\ell^2 \subseteq M_1 \cup M_2$ we have $r - m + \ell' \leq r + 1$ so $\ell' \leq m+1$. Since $ADVICE$ began with one additional bit we have $|ADVICE| \leq \ell' + 1 \leq m + 2$. 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 Upper Bounds

Chor, Kushilevitz, Goldreich and Sudan [4] give an upper bound if one bit queries are returned.

Theorem 3.1 *For all n, 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 Theorem 3.1 with a general communication balancing technique (also from [4]) we obtain the following theorem. We include the proof for completeness.

Theorem 3.2 *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.*

Proof: Assume that a divides n. Otherwise we can add dummy bits to the database to make n the next largest multiple of a. Let $\ell = n/a$.

By Theorem 3.1 we have a 2-server 1-round ℓ–random bit PIR for databases of size ℓ with $\ell - 1$ bit queries and 1-bit answers. We denote this PIR $(q_1, q_2, ANS_1, ANS_2, \phi)$.

Let $x \in \{0,1\}^n$. View $x = x_0 x_1 \cdots x_{n-1}$ as $y_0 y_1 \cdots y_{a-1}$ where each y_k is of length ℓ. Generate a random string ρ of size ℓ.

Suppose the user wants to know x_i where x_i is the jth bit of y_k.

The user sends to server one $q_1(j,\rho)$ and to server two $q_2(j,\rho)$.

Server one sends back

$$ANS_1(y_0, q_1(j,\rho))ANS_1(y_1, q_1(j,\rho)) \cdots ANS_1(y_{a-1}, q_1(j,\rho)).$$

Server two sends back

$$ANS_2(y_0, q_2(j,\rho))ANS_2(y_1, q_2(j,\rho)) \cdots ANS_2(y_{a-1}, q_2(j,\rho)).$$

The user now can compute $x_i = \phi(j, \rho, ANS_1(y_k, q_k(j,\rho)), ANS_2(y_k, q_k(j,\rho)))$.

The user sends $\ell = n/a - 1$ bits to each database and each server returns a bits. ▮

Theorem 3.2 may be optimal for constant a but it is not optimal for nonconstant a: there is a is a 2-server 1-round protocol where queries and answers are both length $O(n^{1/3})$ [4]. The best known lower bound for 2-server 1-round protocols with a-bit answers is $\Omega(n/2^{6a})$ [7].

4 Open Problem

We would like to find tight bounds on m for the case of 2-server 1-round protocols with m-bit queries and a-bit answers. In particular, for constant a, is the upper bound in Theorem 3.2 tight?

5 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 for helpful discussions and Nan Wang 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.

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

[6] 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.

[7] I. Kerenidis and R. de Wolf. Exponential lower bound for 2-query locally decodable codes. In *Proc. of the 35th ACM Sym. on Theory of Computing*, 2003.

[8] 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.

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