Some results on query processes and reconstruction functions for unconditionally secure 2-server 1-round binary private information retrieval protocols

D. R. Stinson
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, N2L 3G1, Canada
dstinson@cacr.math.uwaterloo.ca

R. Wei
Department of Computer Science
Lakehead University
Thunder Bay ON, P7B 5E1, Canada
wei@ccc.cs.lakeheadu.ca

June 12, 2006

Abstract

In this paper, we investigate query processes and reconstruction functions for unconditionally secure 2-server 1-round binary private information retrieval (PIR) schemes. We begin by formulating a simplified model for PIR schemes which is equivalent to the usual model. We show that a query is equivalent to a boolean function of two variables, and we give a precise characterization of the boolean functions that can be used as "query pairs" to the two servers. We also consider several notions of "privacy" and we make a few remarks about the communication complexity of PIR schemes.

1 Introduction

Constructions and bounds for unconditionally secure private information retrieval (PIR) schemes have been discussed by many researchers (see, e.g., [1, 2, 3, 4, 5, 6, 7]). In this note, we consider 2-server 1-round binary private information retrieval, which is the basic situation of PIR.

We begin with a discussion of the model, and then a bit later we give formal definitions in this model. There are two servers, $\mathcal{S}_1$ and $\mathcal{S}_2$. Both servers have the same secret information, $X = (x_1, \ldots, x_n) \in (\mathbb{Z}_2)^n$. A user $\mathcal{U}$ wishes to determine one of the x_i's, say x_j. However, $\mathcal{U}$ does not want either of $\mathcal{S}_1$ or $\mathcal{S}_2$ to be able to determine which x_j the user $\mathcal{U}$ is seeking.

Below we enumerate the main steps in a PIR protocol. This is the special case of the "standard" model considered in [5] where we have two servers and one-bit responses.

1. Suppose that $\mathcal{U}$ wishes to determine the value of x_j. $\mathcal{U}$ sends a *query* Q_1 to $\mathcal{S}_1$ and another query Q_2 to $\mathcal{S}_2$. The *query pair* (Q_1, Q_2) is chosen at random from a (multi)set of possible query pairs, which is denoted by $\mathcal{Q}^j$.

2. Given the query Q_i $(i = 1, 2)$, the server $\mathcal{S}_i$ computes the *response* R_i, which is a deterministic function of Q_i and X. Thus $R_i = f_i(Q_i, X)$ where f_i is the *response function* of $\mathcal{S}_i$, $i = 1, 2$. In the binary case, each response R_i is required to be a single bit, i.e., $R_i \in \mathbb{Z}_2$, $i = 1, 2$. After computing R_i, $\mathcal{S}_i$ transmits R_i to $\mathcal{U}$ $(i = 1, 2)$.

3. Given two responses R_1 and R_2, $\mathcal{U}$ attempts to infer the value of x_j. More precisely, $\mathcal{U}$ has a *reconstruction function* rec which, when given an index j, two suitable queries, and their responses, computes the value of x_j. Thus we desire that
$$\mathsf{rec}(j, Q_1, R_1, Q_2, R_2) = x_j,$$
where R_i is response of $\mathcal{S}_i$ given query Q_i, $i = 1, 2$.

4. The scheme is *private*, which means that the query Q_1 should not reveal the value of j to $\mathcal{S}_1$, and the query Q_2 should not reveal the value of j to $\mathcal{S}_2$. (Various types of privacy will be defined formally in Section 4.)

Example 1.1. We describe a PIR scheme from [5]. On input j, $\mathcal{U}$ chooses a random subset $Q_1 \subseteq \{1, \dots, n\}$ such that $|Q_1|$ is even. Then $\mathcal{U}$ defines $Q_2 = Q_1 \Delta \{j\}$, where Δ denotes the symmetric difference of two sets. The responses are defined to be $R_i = \sum_{j \in Q_i} x_j \bmod 2$, for $i = 1, 2$. Finally, $\mathcal{U}$ computes $x_j = R_1 + R_2 \bmod 2$. □

1.1 Our Contributions

The results in this paper are summarized as follows. In sections 2 and 3, we show how the above-described model can be simplified considerably. Our main observation is that the queries, response functions and reconstruction function can all be assumed to have certain special forms, without loss of generality. More specifically, a query is shown to be equivalent to a boolean function of n boolean variables, and a reconstruction function is shown to be equivalent to a boolean function of two variables. We go on to give a complete description of all the possible query pairs that can be used in PIR schemes, based on the corresponding reconstruction functions. As far as we know, this is the first paper that has provided a detailed study of general (i.e., nonlinear) query pairs and reconstruction functions.

We give formal definitions of three flavours of privacy for PIR in Section 4. The three notions of privacy are termed "perfect privacy", "strong privacy" and "weak privcay" (strong privacy is the usual notion of privacy considered in most previous papers). In Section 5, we briefly discuss some bounds on the communication complexity of PIR which achieve the different levels of privacy. In particular, we give a complete characterization of schemes achieving perfect privacy, and we give an improved construction of schemes achieving strong privacy.

2 A Simplified Model for Private Information Retrieval

In the general model for PIR described in the introduction, the response R_i (computed by $\mathcal{S}_i$, given a query Q_i) is determined by a fixed response function f_i. More specifically, the response is computed as $R_i = f_i(Q_i, X)$. We assume that the functions f_1 and f_2 are public.

Our first observation is that any query $Q = Q_i$ to $\mathcal{S}_i$ is equivalent to a boolean function f_Q of n variables, i.e., $f_Q : (\mathbb{Z}_2)^n \to \mathbb{Z}_2$. The boolean function f_Q, corresponding to the query Q, is defined as follows:
$$f_Q(X) = f_i(Q, X)$$
for all $X \in (\mathbb{Z}_2)^n$. Therefore, we can stipulate without loss of generality that a query Q_i is just a boolean function of n variables, and the response R_i is just the evaluation of Q_i at X, i.e., $R_i = Q_i(X)$. We will use this formulation throughout the rest of this paper.

We will denote the ring of all boolean functions of n boolean variables $x_1, \ldots, x_n$ by $\mathbb{B}(x_1, \ldots, x_n)$. As usual, we define $(f+g)(X) = f(X) + g(X) \bmod 2$ and $(fg)(X) = f(X)g(X)$. Addition of functions can be viewed as a logical "exclusive-or" and multiplication of functions can be viewed as a logical "and". $\mathbb{B}(x_1, \ldots, x_n)$ can also be viewed as the quotient ring $\mathbb{Z}_2[x_1, \ldots, x_n]/({x_1}^2 - x_1, \ldots, {x_n}^2 - x_n)$.

Here are some fundamental properties and definitions relating to $\mathbb{B}(x_1, \ldots, x_n)$.

1. $\mathbb{B}(x_1, \ldots, x_n)$ is a commutative ring. That is, $f(X)g(X) = g(X)f(X)$ for all functions f, g and for all X.

2. $\mathbb{B}(x_1, \ldots, x_n)$ is a idempotent ring. That is, $f^2 = f$ for all functions $f \in \mathbb{B}(x_1, \ldots, x_n)$. (Note that f^2 is the function defined as $f^2(X) = f(X)f(X)$. That is, the exponent means multiplication of a function by itself.) For example, $(x_1 + x_2x_3)^2 = x_1 + x_2x_3$.

3. $|\mathbb{B}(x_1, \ldots, x_n)| = 2^{2^n}$.

4. Every $f \in \mathbb{B}(x_1, \ldots, x_n)$ can be expressed in a unique way as a polynomial in the following form:
$$f(x_1, \ldots, x_n) = \sum_{T \subseteq \{1, \ldots, n\}} a_T \prod_{i \in T} x_i,$$
where $a_T \in \mathbb{Z}_2$ for all $T \subseteq \{1, \ldots, n\}$.

5. $\mathbb{B}(x_1, \ldots, x_n)$ contains zero-divisors. Note that we have $fg = 0$ if $x_j | f$ and $(x_j + 1) | g$ for some j (where 0 denotes the constant function which always takes on the value 0). For example, if $f = x_1 + x_1x_2$ and $g = x_1x_3 + x_3$, then $fg = 0$.

6. $fg = 1$ if and only if $f = g = 1$ (where 1 denotes the constant function which always takes on the value 1).

7. The *jth projection function* is the boolean function π_j defined by $\pi_j(x_1, \ldots, x_n) = x_j$.

Given two boolean functions $f_1, f_2 \in \mathbb{B}(x_1, \ldots, x_n)$, define $\mathbb{B}(f_1, f_2)$ to be the subring of $\mathbb{B}(x_1, \ldots, x_n)$ generated by f_1 and f_2. The following lemma is obvious.

Lemma 2.1. *Suppose $f_1, f_2 \in \mathbb{B}(x_1, \ldots, x_n)$. Then*
$$\mathbb{B}(f_1, f_2) = \{f = a_0 + a_1f_1 + a_2f_2 + a_{12}f_1f_2 : a_0, a_1, a_2, a_{12} \in \mathbb{Z}_2\}.$$

Remark 2.1. From the above lemma, it is easy to see that $|\mathbb{B}(f_1, f_2)| \leq 16$ for any f_1, f_2.

Suppose that Q_1 and Q_2 are two queries. As discussed above, both Q_1 and Q_2 are boolean functions of n variables. We write $\{Q_1, Q_2\} \rightsquigarrow x_j$ if it always possible to infer the value of x_j when given the values $R_1 = Q_1(X)$ and $R_2 = Q_2(X)$. More formally, $\{Q_1, Q_2\} \rightsquigarrow x_j$ if the following property holds: there do not exist two n-tuples $X = (x_1, \ldots, x_n)$ and $X' = (x'_1, \ldots, x'_n)$, with $x_j \neq x'_j$, such that $(Q_1(X), Q_2(X)) = (Q_1(X'), Q_2(X'))$. If $\{Q_1, Q_2\} \rightsquigarrow x_j$, we say that the query pair $\{Q_1, Q_2\}$ *determines* x_j. Our next theorem provides a simple and efficient way to decide if $\{Q_1, Q_2\} \rightsquigarrow x_j$.

Theorem 2.2. *$\{Q_1, Q_2\} \rightsquigarrow x_j$ if and only if $\pi_j \in \mathbb{B}(Q_1, Q_2)$.*

Proof. Suppose that $\pi_j \in \mathbb{B}(Q_1, Q_2)$. Then

$$\pi_j = a_0 + a_1 Q_1 + a_2 Q_2 + a_{12} Q_1 Q_2,$$

where $a_0, a_1, a_2, a_{12} \in \mathbb{Z}_2$. Hence,

$$\begin{aligned} x_j &= \pi_j(X) \\ &= a_0 + a_1 Q_1(X) + a_2 Q_2(X) + a_{12} Q_1(X) Q_2(X) \\ &= a_0 + a_1 R_1 + a_2 R_2 + a_{12} R_1 R_2. \end{aligned}$$

Conversely, suppose that $\{Q_1, Q_2\} \rightsquigarrow x_j$. For all four ordered pairs $(y_1, y_2) \in (\mathbb{Z}_2)^2$, define

$$h(y_1, y_2) = \begin{cases} z & \text{if } R_1 = y_1, R_2 = y_2 \Rightarrow x_j = z \\ 0 & \text{if there does not exist } X \in (\mathbb{Z}_2)^n \text{ such that } R_1 = y_1, R_2 = y_2. \end{cases}$$

The boolean function h can be expressed in the form

$$h(y_1, y_2) = a_0 + a_1 y_1 + a_2 y_2 + a_{12} y_1 y_2,$$

where $a_0, a_1, a_2, a_{12} \in \mathbb{Z}_2$. Now, for any $X \in (\mathbb{Z}_2)^n$, we have that

$$\begin{aligned} a_0 + a_1 R_1 + a_2 R_2 + a_{12} R_1 R_2 &= h_j(R_1, R_2) \\ &= x_j, \end{aligned}$$

where $R_i = Q_i(X)$, $i = 1, 2$. Hence, $\pi_j = a_0 + a_1 Q_1 + a_2 Q_2 + a_{12} Q_1 Q_2$, and therefore $\pi_j \in \mathbb{B}(Q_1, Q_2)$. □

In the above proof, the boolean function h can be used as the reconstruction function for x_j in the PIR protocol. Therefore, a reconstruction function rec can be assumed without loss of generality to have the following form:

$$\mathsf{rec}(j, Q_1, R_1, Q_2, R_2) = h(R_1, R_2),$$

where $h : (\mathbb{Z}_2)^2 \to \mathbb{Z}_2$.

We illustrate with a small example.

Example 2.1. Suppose $n = 3$, $Q_1(X) = x_1 x_2 + x_1 x_2 x_3 + x_1$, and $Q_2(X) = x_1 x_2$. Then it can be verified that

$$Q_1(X) Q_2(X) = x_1 x_2 + x_1 x_2 x_3 + x_1 x_2 = x_1 x_2 x_3.$$

It is easy to check that

$$Q_1(X) Q_2(X) + Q_1(X) + Q_2(X) = x_1 x_2 x_3 + x_1 x_2 + x_1 x_2 x_3 + x_1 + x_1 x_2 = x_1.$$

Therefore, $Q_1 Q_2 + Q_1 + Q_2 = \pi_1$ and hence $\pi_1 \in \mathbb{B}(Q_1, Q_2)$.

Theorem 2.2 then states that $\{Q_1, Q_2\} \rightsquigarrow x_1$. In fact, x_1 can be reconstructed from R_1 and R_2 by using the formula

$$x_1 = h_1(R_1, R_2) = R_1 R_2 + R_1 + R_2.$$

□

Summarizing, we have shown that we can assume the following simplifications without loss of generality:

- a query is equivalent to a boolean function of n variables
- a response is just the evaluation of the query function on input X
- a reconstruction function (if it exists) is equivalent to a boolean function of two variables
- the desired value x_j is obtained by evaluating the reconstruction function using the two responses as inputs.

3 Valid Query Pairs

In this section, we investigate how to construct query pairs (Q_1, Q_2) such that $\{Q_1, Q_2\} \rightsquigarrow x_j$. Such a query pair is said to be *valid* for x_j. By Theorem 2.2, this is equivalent to finding $Q_1, Q_2 \in \mathbb{B}(x_1, \ldots, x_n)$ so that $\pi_j \in \mathbb{B}(Q_1, Q_2)$.

Except in "degenerate" cases, there are 16 functions in $\mathbb{B}(Q_1, Q_2)$. Two of these functions are the constant functions 0 and 1, which obviously are never equal to π_j. Fourteen functions remain to be considered, which are listed in Table 1. For each function $h : (\mathbb{Z}_2)^2 \to \mathbb{Z}_2$, we specify in Table 1 the conditions under which the projection function π_j is expressible as $h(Q_1, Q_2)$.

Conditions 1 to 6 (i.e., the ones corresponding to the linear functions) in Table 1 are obvious. We now prove the validity of condition 7.

Lemma 3.1. *The function $\pi_j = Q_1Q_2$ if and only if*

$$Q_1 = x_j(Q_1' + 1) + Q_1' \quad \textit{and} \quad Q_2 = x_j(Q_2' + 1) + Q_2',$$

where $Q_1', Q_2' \in \mathbb{B}(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$ and $Q_1'Q_2' = 0$.

Proof. Write $Q_1 = x_jQ_1'' + Q_1'$ and $Q_2 = x_jQ_2'' + Q_2'$, where $Q_1', Q_1'', Q_2', Q_2'' \in \mathbb{B}(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$. Then, we have that

$$\begin{aligned} Q_1Q_2 &= (x_jQ_1'' + Q_1')(x_jQ_2'' + Q_2') \\ &= x_j(Q_1''Q_2'' + Q_1''Q_2' + Q_1'Q_2'') + Q_1'Q_2'. \end{aligned}$$

It follows that $\pi_j = Q_1Q_2$ if and only if

$$x_j(Q_1''Q_2'' + Q_1''Q_2' + Q_1'Q_2'' + 1) = Q_1'Q_2'. \tag{1}$$

Equation (1) is easily seen to be equivalent to the two equations

$$Q_1'Q_2' = 0 \tag{2}$$

and

$$Q_1''Q_2'' + Q_1''Q_2' + Q_1'Q_2'' = 1. \tag{3}$$

Equation (3) can be rewritten as

$$(Q_1'' + Q_1')(Q_2'' + Q_2') + Q_1'Q_2' = 1, \tag{4}$$

Table 1: Valid query pairs for a bit x_j

	reconstruction function h	conditions for $h(Q_1, Q_2) = x_j$
1.	Q_1	$Q_1 = x_j$
2.	Q_2	$Q_2 = x_j$
3.	$Q_1 + 1$	$Q_1 = x_j + 1$
4.	$Q_2 + 1$	$Q_2 = x_j + 1$
5.	$Q_1 + Q_2$	$Q_2 = x_j + Q_1$
6.	$Q_1 + Q_2 + 1$	$Q_2 = x_j + Q_1 + 1$
7.	$Q_1 Q_2$	$Q_1 = x_j(Q'_1 + 1) + Q'_1$ $Q_2 = x_j(Q'_2 + 1) + Q'_2$ $Q'_1 Q'_2 = 0$
8.	$Q_1 Q_2 + Q_1$	$Q_1 = x_j(Q'_1 + 1) + Q'_1$ $Q_2 = x_j(Q'_2 + 1) + Q'_2 + 1$ $Q'_1 Q'_2 = 0$
9.	$Q_1 Q_2 + Q_2$	$Q_1 = x_j(Q'_1 + 1) + Q'_1 + 1$ $Q_2 = x_j(Q'_2 + 1) + Q'_2$ $Q'_1 Q'_2 = 0$
10.	$Q_1 Q_2 + Q_1 + Q_2$	$Q_1 = x_j(Q'_1 + 1)$ $Q_2 = x_j(Q'_2 + 1)$ $Q'_1 Q'_2 = 0$
11.	$Q_1 Q_2 + 1$	$Q_1 = x_j(Q'_1 + 1) + 1$ $Q_2 = x_j(Q'_2 + 1) + 1$ $Q'_1 Q'_2 = 0$
12.	$Q_1 Q_2 + Q_1 + 1$	$Q_1 = x_j(Q'_1 + 1) + 1$ $Q_2 = x_j(Q'_2 + 1)$ $Q'_1 Q'_2 = 0$
13.	$Q_1 Q_2 + Q_2 + 1$	$Q_1 = x_j(Q'_1 + 1)$ $Q_2 = x_j(Q'_2 + 1) + 1$ $Q'_1 Q'_2 = 0$
14.	$Q_1 Q_2 + Q_1 + Q_2 + 1$	$Q_1 = x_j(Q'_1 + 1) + Q'_1 + 1$ $Q_2 = x_j(Q'_2 + 1) + Q'_2 + 1$ $Q'_1 Q'_2 = 0$

Notation: in this table, Q'_1 and Q'_2 always denote functions that do not involve the variable x_j, i.e., $Q'_1, Q'_2 \in \mathbb{B}(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$.

which, using (2), simplifies to

$$(Q_1'' + Q_1')(Q_2'' + Q_2') = 1. \tag{5}$$

Finally, (5) is satisfied if and only if

$$Q_1'' + Q_1' = Q_2'' + Q_2' = 1,$$

and the desired result follows. □

Example 3.1. Suppose we take $Q_1' = x_2x_3$ and $Q_2' = (x_2+1)x_3$. Then $Q_1'Q_2' = 0$. Applying Lemma 3.1, if we define

$$\begin{aligned} Q_1 &= x_1(x_2x_3+1) + x_2x_3 \\ &= x_1x_2x_3 + x_1 + x_2x_3, \quad \text{and} \\ Q_2 &= x_1((x_2+1)x_3+1) + (x_2+1)x_3 \\ &= x_1x_2x_3 + x_1x_3 + x_1 + x_2x_3 + x_3, \end{aligned}$$

then we have that $Q_1Q_2 = x_1$. □

Now we can prove the other conditions in Table 1 using the following theorem.

Theorem 3.2. *Let $e_1, e_2, e_3 \in \mathbb{Z}_2$. The function $\pi_j = (Q_1+e_1)(Q_2+e_2)+e_3$ if and only if*

$$Q_1 = (x_j+e_3)(Q_1'+1) + Q_1' + e_1 \quad \textit{and} \quad Q_2 = (x_j+e_3)(Q_2'+1) + Q_2' + e_2,$$

where $Q_1', Q_2' \in \mathbb{B}(x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_n)$ and $Q_1'Q_2' = 0$.

Proof. $(Q_1+e_1)(Q_2+e_2)+e_3 = \pi_j$ if and only if $(Q_1+e_1)(Q_2+e_2) = \pi_j + e_3$. From Lemma 3.1 we know that the latter equation holds if and only if

$$(Q_1+e_1) = (x_j+e_3)(Q_1'+1) + Q_1' \quad \text{and} \quad (Q_2+e_2) = (x_j+e_3)(Q_2'+1) + Q_2',$$

where $Q_1', Q_2' \in \mathbb{B}(x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_n)$ and $Q_1'Q_2' = 0$. The conclusion follows. □

3.1 An Additional Simplification

We define a boolean function $f(X) \in \mathbb{B}(x_1, \dots, x_n)$ to be *homogeneous* if $f(0, \dots, 0) = 0$. If $g(X)$ is not homogeneous, then the function $f(X) = g(X)+1$ is homogeneous. Clearly these two functions are equivalent in the sense that we can compute $f(X)$ if and only if we can compute $g(X)$.

The above observation allows us to restrict all query functions to be homogeneous, without loss of generality. If we impose this additional requirement, then it turns out that only require homogeneous reconstruction functions, as well.

Lemma 3.3. *Suppose that $Q_1, Q_2 \in \mathbb{B}(x_1, \dots, x_n)$ are both homogeneous, and suppose that $h(Q_1, Q_2) = \pi_j$ for some j. Then h is a homogeneous function of Q_1 and Q_2.*

Proof. We have that $Q_1(0, \dots, 0) = Q_2(0, \dots, 0) = 0$. Also,

$$0 = \pi_j(0, \dots, 0) = h(0, 0).$$

Therefore h is homogeneous. □

An immediate consequence of Lemma 3.3 is that we need only consider seven of the 14 possible reconstruction functions listed in Table 1, namely, functions $1, 2, 5, 7, 8, 9$ and 10.

Summarizing, we have the following.

Theorem 3.4. *Suppose that $Q_1, Q_2 \in \mathbb{B}(x_1, \ldots, x_n)$ are both homogeneous, and suppose that $h(Q_1, Q_2) = \pi_j$ for some j. Then h is one of the functions Q_1, Q_2, Q_1+Q_2, $Q_1Q_2+Q_1$, $Q_1Q_2+Q_2$, Q_1Q_2 or $Q_1Q_2 + Q_1 + Q_2$.*

4 Privacy

4.1 Notions of Privacy

So far, we have not considered the privacy requirement. In this section, we give a formal definition of a privacy in a PIR scheme. For $1 \leq j \leq n$, $\mathcal{U}$ will have a collection $\mathcal{Q}^j$ of query pairs such that $\{Q_1, Q_2\} \rightsquigarrow x_j$ for all $(Q_1, Q_2) \in \mathcal{Q}^j$. We will assume that every $\mathcal{Q}^j$ consists of ℓ query pairs. Note that a $\mathcal{Q}^j$ is allowed to contain "repeated" query pairs, so $\mathcal{Q}^j$ is a multiset.

Here is the process $\mathcal{U}$ uses to find the value of a random bit in X, making use of the simplifications described in the previous sections.

1. User $\mathcal{U}$ chooses $j \in \{1, \ldots, n\}$ uniformly at random.
2. $\mathcal{U}$ chooses $(Q_1, Q_2) \in \mathcal{Q}^j$ uniformly at random.
3. $\mathcal{U}$ sends query Q_1 to $\mathcal{S}_1$ and query Q_2 to $\mathcal{S}_2$.
4. $\mathcal{U}$ receives responses R_1 from $\mathcal{S}_1$ and R_2 from $\mathcal{S}_2$, where $R_i = Q_i(X)$, $i = 1, 2$.
5. $\mathcal{U}$ computes x_j from R_1 and R_2 using a suitable reconstruction function $h : (\mathbb{Z}_2)^2 \rightarrow \mathbb{Z}_2$, by computing $x_j = h(R_1, R_2)$.

Since each query pair is associated with a reconstruction function, we will denote a query pair (Q_1, Q_2) by a triple $(h; Q_1, Q_2) \in \mathcal{Q}^j$ in some situations. As previously discussed, we will only consider homogeneous reconstructions.

We will give three definitions of privacy of PIR protocols. The standard definition used in most PIR papers is what we term "strong privacy". We also give a stronger notion, "perfect privacy", and a weaker notion, "weak privacy". Both of theses definitions are new, as far as we know.

First, we define "perfect privacy" for a PIR. Roughly speaking, if a PIR provides perfect privacy, then the servers cannot get any information about the value of j during the query even if they know the reconstruction function associated with any query. (In general, we do not assume that the servers know the reconstruction function being used. This definition considers the situation where the servers somehow have obtained this additional information.) A formal definition is given as follows.

Definition 4.1. *For $1 \leq j \leq n$ and $i = 1, 2$, define*

$$\pi_i^*(\mathcal{Q}^j) = [(h; Q_i) : (h; Q_1, Q_2) \in \mathcal{Q}^j].$$

We say that a PIR provides perfect privacy *if*

$$\pi_i^*(\mathcal{Q}^1) = \pi_i^*(\mathcal{Q}^2) = \ldots = \pi_i^*(\mathcal{Q}^n)$$

for $i = 1, 2$, where equality denotes multiset equality.

Here are two additional types of privacy. “Strong privacy” means that a query provides no information as to the value of j, while “weak privacy” means that no possible value of j can be ruled out, given any query.

Definition 4.2. *For $1 \leq j \leq n$ and $i = 1, 2$, define*

$$\pi_i(\mathcal{Q}^j) = [Q_i : (h; Q_1, Q_2) \in \mathcal{Q}^j].$$

We say that a PIR provides strong privacy *if*

$$\pi_i(\mathcal{Q}^1) = \pi_i(\mathcal{Q}^2) = \ldots = \pi_i(\mathcal{Q}^n)$$

for $i = 1, 2$, where equality denotes multiset equality.

Definition 4.3. *For $1 \leq j \leq n$ and $i = 1, 2$, define*

$$\pi_i^d(\mathcal{Q}^j) = \{Q_i : (h; Q_1, Q_2) \in \mathcal{Q}^j\}.$$

We say that a PIR provides weak privacy *if*

$$\pi_i^d(\mathcal{Q}^1) = \pi_i^d(\mathcal{Q}^2) = \ldots = \pi_i^d(\mathcal{Q}^n)$$

for $i = 1, 2$, where equality denotes set equality.

Remark 4.1. Note that $\pi_i(\mathcal{Q}^j)$ is the multiset formed by taking all the queries to $\mathcal{S}_i$ from the collection $\mathcal{Q}^j$. Further, $\pi_i^d(\mathcal{Q}^j))$ is the set of distinct queries in the multiset $\pi_i(\mathcal{Q}^j)$.

We will see in Section 5 that a PIR scheme providing only weak privacy can have far fewer queries than a scheme that provides strong or perfect privcay.

4.2 Characterization of Perfect Privacy

In this section, we show that every reconstruction function must be linear if perfect privacy is achieved. First, we discuss some relationships between reconstruction functions and queries. We will use the following notations in this section:

- $Q^{(j)}$ denotes any function in $\mathbb{B}(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$,
- $Q^{(ij)}$ denotes any function in $\mathbb{B}(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$, etc.

Here is a useful preliminary lemma.

Lemma 4.1. *Suppose $(h; Q_1, Q_2) \in \mathcal{Q}^j$ where h is nonlinear and homogeneous. Let $Q_i = x_j Q_i^{(j)} + R_i^{(j)}$. Then either $R_i^{(j)} = 0$ or $Q_i^{(j)} + R_i^{(j)} = 0$ or 1, where $i = 1, 2$.*

Proof. The conclusion follows immediately from Table 1. □

Remark 4.2. Lemma 4.1 gives a method for a server to check whether a query can be associated with a nonlinear reconstruction function h. When a server receives a query Q_i, the server writes $Q_i = x_j Q_i^{(j)} + R_i^{(j)}$ for each possible j. Using Lemma 4.1, it is easy to determine whether h is (possibly) a nonlinear function.

Lemma 4.2. *Suppose $(h; Q_1, Q_2) \in \mathcal{Q}^j$ for some nonlinear function h. If $(h'; Q_1, Q_2') \in \mathcal{Q}^i$ and $(h''; Q_1', Q_2) \in \mathcal{Q}^i$, where $i \neq j$, then at least one of h' or h'' is linear.*

Proof. Suppose $h(Q_1, Q_2) = Q_1 Q_2$. Then

$$\begin{aligned} Q_1 &= x_j(Q_1^{(j)} + 1) + Q_1^{(j)} \\ &= x_j(x_i Q_1^{(ij)} + R_1^{(ij)} + 1) + x_i Q_1^{(ij)} + R_1^{(ij)} \\ &= x_i(x_j Q_1^{(ij)} + Q_1^{(ij)}) + x_j(R_1^{(ij)} + 1) + R_1^{(ij)}. \end{aligned}$$

Assume a pair $(h'; Q_1, Q_2') \in \mathcal{Q}^i$ is associated with a nonlinear function. According to Lemma 4.1, one of the following cases must occur:

1. $x_j(R_1^{(ij)} + 1) + R_1^{(ij)} = 0$,
2. $x_j Q_1^{(ij)} + Q_1^{(ij)} = x_j(R_1^{(ij)} + 1) + R_1^{(ij)} + 1$, or
3. $x_j Q_1^{(ij)} + Q_1^{(ij)} = x_j(R_1^{(ij)} + 1) + R_1^{(ij)}$.

It is easy to see that cases 1 and 3 are impossible. Hence, we consider case 2. In this case, we must have $R_1^{(ij)} = Q_1^{(ij)} + 1$. Let

$$\begin{aligned} Q_2 &= x_j(Q_2^{(j)} + 1) + Q_2^{(j)} \\ &= x_j(x_i Q_2^{(ij)} + R_2^{(ij)} + 1) + x_i Q_2^{(ij)} + R_2^{(ij)}. \end{aligned}$$

Since $Q_1^{(j)} Q_2^{(j)} = 0$, we have

$$(x_i Q_1^{(ij)} + Q_1^{(ij)} + 1)(x_i Q_2^{(ij)} + R_2^{(ij)}) = 0.$$

This gives us

$$x_i(Q_1^{(ij)} R_2^{(ij)} + Q_2^{(ij)}) + Q_1^{(ij)} R_2^{(ij)} + R_2^{(ij)} = 0,$$

so we have

$$Q_1^{(ij)} R_2^{(ij)} + Q_2^{(ij)} = Q_1^{(ij)} R_2^{(ij)} + R_2^{(ij)} = 0$$

and then

$$R_2^{(ij)} = Q_2^{(ij)}.$$

Therefore

$$Q_2 = x_i(x_j Q_2^{(ij)} + Q_2^{(ij)}) + x_j Q_2^{(ij)} + x_j + Q_2^{(ij)}.$$

From Lemma 4.1, if $(h''; Q_1', Q_2) \in \mathcal{Q}^i$, then h'' is linear.

In the above, we proved the conclusion when $h(Q_1, Q_2) = Q_1 Q_2$. When f is some other nonlinear function, the proof can be carried out in a similar way. □

The following lemma is obvious from the Table 1 and the definition of perfect privacy.

Lemma 4.3. *Suppose a PIR provides perfect privacy. If $(h; Q_1, Q_2) \in \mathcal{Q}^j$, where h is linear, then $Q_1 + Q_2 = x_j$.*

Proof. Recall that we just consider homogeneous reconstruction functions. If $h = Q_1$, then $Q_1 = x_j$ and $(h; Q_1) \notin \pi_1^*(\mathcal{Q}^i)$ for any $i \neq j$. The PIR does not achieve perfect privacy in this case. For the same reason, $h \neq Q_2$, and the conclusion follows. □

Theorem 4.4. *If a PIR provides perfect privacy, then every reconstruction function used in the scheme is linear.*

Proof. Lemma 4.2 tells us that if $(h; Q_1, Q_2) \in \mathcal{Q}^j$ for some nonlinear function h, then either $(h; Q_1) \notin \pi_1^*(\mathcal{Q}^i)$ or $(h; Q_2) \notin \pi_2^*(\mathcal{Q}^i)$. The conclusion follows. □

5 Communication Complexity

In general, we want to achieve privacy in such a way that the size of the query collections, namely, ℓ, is minimized. The generation of a random query pair requires $r = \lceil \log_2 \ell \rceil$ random bits. The number of random bits required to specify a query to $\mathcal{S}_i$ is $\log_2 |\pi_i^d(\mathcal{Q}^j)|$ (which is independent of j).

In this section, we consider bounds on ℓ. Recently, it was proven in [2] that $\ell \geq 2^{n-2}$ for any PIR that provides strong privacy. We also note that [5] and [6] proved that $\ell \geq 2^{n-1}$ under certain special conditions. Next, we give a bound for perfect privacy, which turns out to be an exact bound.

Theorem 5.1. *If a PIR provides perfect privacy, then the number of query bits required in the scheme is at least* 2^{n-1}.

Proof. In [6], it was proven that if the reconstruction functions are restricted to linear summations of the two inputs, then the lower bound for the number of query bits required in the scheme is 2^{n-1}. Combining this result with Lemma 4.3 and Theorem 4.4, the conclusion follows. □

Remark 5.1. In [5], it was shown that if the reconstruction functions are restricted to linear summations of the two inputs, and if the queries are linear functions of the n boolean variables, then the number of query bits required in the scheme is at least 2^{n-1}. In fact, the proof given in [5] can be adapted in a straightforward fashion to apply to the case of arbitrary queries.

The bound proven in Theorem 5.1 is an exact bound, because [5] constructed a PIR scheme with 2^{n-1} query bits that achieves perfect privacy. In fact, the scheme is the one that we presented in Example 1.1; it is easy to verify that it achieves perfect privacy and the query sets have size 2^{n-1}. This scheme is also the best previous construction for PIR schemes that achieve strong privacy. However, we will construct schemes that yield a small improvement. First, we present a small example.

Example 5.1. *Suppose* $n = 3$, *and the query sets* $\mathcal{Q}^1, \mathcal{Q}^2, \mathcal{Q}^3$ *are defined as follows:*

$\mathcal{Q}^1$	$\mathcal{Q}^2$	$\mathcal{Q}^3$
$(x_1 + x_2, x_2)$	$(x_1 + x_2, x_1)$	$(x_1 + x_2, x_3)$
$(x_1 + x_3, x_3)$	$(x_1 + x_3, x_2)$	$(x_1 + x_3, x_1)$
$(x_2 + x_3, x_1)$	$(x_2 + x_3, x_3)$	$(x_2 + x_3, x_2)$

This PIR achieves strong privacy.

This example can be generalized to any $n \geq 3$ as follows. Suppose Q is a linear function in $\mathbb{B}(x_1, \ldots, x_n)$. Then Q is equivalent to a subset of $\{1, \ldots, n\}$ (the elements of the subset are those i that the coefficient of x_i is 1). There are $2^{n-1} - 1$ non-empty even subsets which are denoted as E. For any fixed j and any $Q_1 \in E$, $Q_1 \neq x_1 + x_2 + x_3 + x_j$, define $Q_2 = Q_1 + x_j$. For $Q_1 = x_1 + x_2 + x_3 + x_j$, define $Q_2 = x_j$. It is readily checked that the result is a PIR with strong privacy. In this method, $\pi_2(\mathcal{Q}^j)$ contains all the odd subsets of $\{1, \ldots, n\}$ except $\{1, 2, 3\}$. So we have the following result.

Theorem 5.2. *For all $n \geq 3$, there exists a PIR for a database of size n with query set size $\ell = 2^{n-1} - 1$, which achieves strong privacy.*

Finally, we turn to weak privacy. A PIR achieving weak privacy can use considerably fewer query bits. We provide some simple examples of PIR achieving weak privacy with $O(\log n)$ query bits.

Theorem 5.3. *There exists a PIR scheme for a database of size $n \geq 2$ that achieves weak privacy and in which $|\mathcal{Q}^j| = 2n - 2$.*

Proof. Let $\mathcal{Q}^j = \{(x_i, x_j) : i \neq j\} \bigcup \{(x_j, x_i) : i \neq j\}$. Then $S(\pi_i(\mathcal{Q}^j))$ consists of all single variables. □

Theorem 5.4. *There exists a PIR for a database of size $n \geq 3$ that achieves weak privacy and in which $|\mathcal{Q}^j| = n(n-1)/2$.*

Proof. Let $\mathcal{Q}^j = \{(x_{i_1} + x_{i_2}, x_j) : j \neq i_1 \neq i_2 \neq j\} \bigcup \{(x_i + x_j, x_i) : i \neq j\}$. $S(\pi_1(\mathcal{Q}^j))$ consists of all sums of two variables and $S(\pi_2(\mathcal{Q}^j))$ consists of all single variables. □

Remark 5.2. The PIR constructed in Theorem 5.4 has the property that it achieves strong privacy with respect to $\mathcal{S}_1$, but only weak privacy with respect to $\mathcal{S}_2$.

6 Conclusion

We have presented a simplification of the model used for unconditionally secure 2-server PIR schemes. We hope that the simplified model and the characterization of valid query pairs will prove useful in the design and analysis of unconditionally secure PIR schemes. Finally, we note that it is straightforward to generalize Theorem 2.2 to s-server schemes, $s \geq 2$.

Acknowledgements

Research of the authors is supported NSERC discovery grant 203114-06 (DRS) and NSERC discovery grant 239135-06 (RW).

References

[1] A. Ambainis. Upper bound on the communication complexity of private information retrieval. *Lecture Notes in Computer Science* **1256** (1997), 401–407.

[2] R. Beigel, L. Fortnow and W. Gasarch, A nearly tight lower bound for privacy information retrieval protocols, *Electronic Colloquium on Computational Complexity*, Report No. 87, 2003.

[3] A. Beimel, Y. Ishai, E. Kushilevitz and J.-F. Raymond. Breaking the $O\left(n^{1/(2k-1)}\right)$ barrier for information-theoretic private information retrieval, *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science*, 2002, 261–270.

[4] C. Cachin, S. Micali and M. Stadler. Computationally private information retrieval with polylogarithmic communication, *Lecture Notes in Computer Science* **1592** (1999), 402–414.

[5] N. Chor, O. Goldreich, E. Kushilevitz and M. Sudan. Private information retrieval, *Journal of the ACM* **45** (1998), 965–982.

[6] I. Kerenidis and R. de Wolf. Exponential lower bound for 2-query locally decodable codes via a quantum argument, *Proceedings of the 35th ACM Symposium on Theory of Computing*, 2003, 106–115.

[7] E. Kushilevitz and R. Ostrovsky. Replication is not needed: Single database, computationally private information retrieval, *Proceedings of the 38th IEEE Symposium on Foundations of Computer Science* (1997), 364–373.