Private Information Schemes Using Cyclic Codes

Ş. Bodur, E. Martínez-Moro, D. Ruano

Abstract—A Private Information Retrieval (PIR) scheme allows users to retrieve data from a database without disclosing to the server information about the identity of the data retrieved. A coded storage in a distributed storage system with colluding servers is considered in this work, namely the approach in [1] which considers an storage and retrieval code with a transitive group and provides binary PIR schemes with the highest possible rate. Reed-Muller codes were considered in [1], cyclic codes are considered in this work since they have a transitive group and can be defined over a binary field as well. We show that binary PIR schemes provide a larger constellation of PIR parameters and they outperform the ones coming from Reed-Muller codes in some cases.

Index Terms—Private information retrieval, cyclic codes, Reed-Muller codes.

I. INTRODUCTION

Many protocols protect the user and server from third parties while accessing the data. Nevertheless, no security measure protects the user from the server. As a result of this demand, Private Information Retrieval (PIR) protocols emerged [2]. They allow users to retrieve data from a database without disclosing to the server information about the identity of the data retrieved. We consider in this work that data is stored in a Distributed Storage System (DSS), since, if data is stored in a single database, one can only guarantee information theoretic privacy by downloading the full database, which has a high communication cost.

Shah et al. [3] have shown that privacy is guaranteed when one bit more than the requested file size is downloaded, but it requires many servers. In case of non-response or a fail from some servers, the PIR scheme should allow servers to communicate with each other. Hence, it is natural to assume that the servers may collude, that is, they may inform each other of their input from the user. A scheme that addresses the situation where any $t$ servers may collude is called a $t$-private information retrieval scheme and it was considered in [4], [5]. This approach, that we consider in this work, uses Coding Theory and the security and performance depend on the parameters of linear codes and their star products (also called Shur products).

The PIR scheme maximum possible rate, was examined without collusion in [6] and with collusion in [7]. The PIR capacity obtained without colluding servers and using a Maximum Distance Separable (MDS) code was given in [8]. In [4], a PIR scheme with colluding server capacity for Generalized Reed Solomon (GRS) codes is given. Their PIR scheme rate is based on the minimum distance of a star product of the storage code and the retrieval code.

The use of a GRS, or an MDS code, requires working over a big base field. In order to address this issue, since binary base fields are desirable for practical implementations, [1] provided a PIR scheme that is based on binary Reed-Muller (RM) codes. They observed that the scheme reaches the highest possible rate if the codes used to define the PIR scheme have a transitive automorphism group, which is the case of RM codes.

In this work we propose to use cyclic codes to construct PIR schemes in the same fashion as [1]. Cyclic codes have also a transitive automorphism group and they can be defined over a binary (or small) finite field as well. Moreover, the star product of two cyclic codes is a cyclic code and its parameters can be computed [9]. Namely, the star product of two cyclic codes is given by the sum of their generating sets and we can compute its dimension and estimate its minimum distance considering cyclotomic cosets.

The main contributions of this work are given in Section V. In order to show the goodness of this family of codes for PIR schemes, we first provide pairs of cyclic codes $C$ and $D$, the storage code and retrieval code, such that the parameters of $C$, $D$, $D^\perp C \ast D$ and $(C \ast D)^\perp$ are - at the same time- optimal or the best known. As we will recall in Section II their parameters determine the performance of the PIR scheme defined by $C$ and $D$. Since a punctured RM code is a cyclic code, we may obtain retrieve PIR schemes using punctured RM codes by using cyclic codes. Moreover, we show that by using cyclic codes we obtain a larger constellation of possible parameters of binary PIR schemes. The construction of PIR schemes and the computations of their parameters follow from a detailed analysis of cyclotomic cosets. Then we focus in the privacy and in the rate of a PIR scheme since the upload cost in a PIR scheme can be neglected [8]. More concretely, in case that the storage code $C$ has dimension 2, we obtain binary PIR schemes that greatly outperform the ones of obtained using RM codes, more concretely they protect against a more significant number of colluding servers. Finally, we compare our schemes with shortened RM codes and we show that in this case the PIR schemes using cyclic codes outperform them as well, namely they offer more privacy for a fixed rate.

II. GENERAL PRIVATE INFORMATION SCHEME

This section reviews some basic definitions of linear codes and briefly recalls PIR schemes (see [1], [4], [5] for further details). We denote by $\mathbb{F}_q$ the finite field with $q$ elements. A linear code $C$ is a linear subspace of $\mathbb{F}_q^n$. We denote its parameters by $[n, k, d]$. In this section, we will consider a general $q$-ary linear code $C$ of length $n$. The set of parity-check equations of a linear code $C$ is given by the set of all linear combinations of the columns of the parity-check matrix $H$ of $C$. The set of all linear combinations of the rows of $H$ is the set of all parity-check equations of $C$. The minimum distance of a linear code $C$ is the smallest weight of a non-zero codeword in $C$. The length of a linear code $C$ is the number of its columns. The dimension of a linear code $C$ is the number of its rows. The maximum distance separable (MDS) code is a code whose minimum distance is equal to the length of the code. A code is called cyclic if it is closed under cyclic shifts. The cyclic code is a linear code that is generated by a single cyclic shift of a single codeword. The cyclic code is a linear code that is generated by a single cyclic shift of a single codeword. The cyclic code is a linear code that is generated by a single cyclic shift of a single codeword.

This work was supported in part by Grant PGC2018-096446-B-C21 funded by MCIN/AEI/10.13039/501100011033 and by “ERDF A way of making Europe”, by Grant RYC-2016-20208 funded by MCIN/AEI/10.13039/501100011033 and by “ESF Investing in your future”, and by Grant CONTPR-2019-385 funded by Universidad de Valladolid and Banco Santander.

Ş. Bodur, E. Martínez-Moro, D. Ruano are with IMUVA-Mathematics Research Institute, Universidad de Valladolid, Spain. e-mail: seyma.bodur,edgar.martinez,diego.ruano@uva.es
Definition 1. Given two linear codes $C$ and $D$ of length $n$ over $\mathbb{F}_q$, we define their star product $C \star D$ as the linear code in $\mathbb{F}_q^n$ spanned by the set \{ $c \star d : c \in C, d \in D$ \}, where $\star$ denotes the component-wise product $c \star d = (c_1d_1, \ldots, c_nd_n)$.

A. PIR Schemes

A PIR scheme consists of three stages: Data Storage, File Request and Response Process. In the Data Storage Process, files are uploaded to a DSS. In the File Request, users decide the file they want to retrieve, called the desired file, and according to it, they select a random codeword that is sent to the servers. In the final Response Process, servers multiply the files corresponding to the random codeword received and then send this matrix back to the user. It should be noted that the servers do not have any information about the file requested by the user.

1) Data Storage Process: We have $r$-files, each file has $r$-rows, $k$-columns, and elements of files in $\mathbb{F}_q$. Since the number of files is $r$, the total file can be understood as $rp \times k$ matrix denoted by $A$, and each file is denoted by $a^i$, where $i \in \{1, \ldots, r\}$.

The files are stored in a DSS. In order to upload the files into the servers, files are encoded by a $k$-dimensional storage code $C \subseteq \mathbb{F}_q^k$ with parameters $[n, k, d]$. Concerning encoding, we multiply the matrix $A$, which covers all files, by $G_C$ the generator matrix of the linear code $C$ and we obtain the matrix $Y := A \cdot G_C$. Since, $A$ is an $rp \times k$ matrix, $Y$ has $rp$ rows and $n$ columns.

2) Request and Response Process: Let assume that the user wants to retrieve the file $a^i$. Then the user chooses a random query $Q^i$ and sends this query to the servers. Each server computes the inner product of $Q^i$ and $Y_j$, where $j$ is the server index, i.e., $j$th-server computes $\langle Q^i, Y_j \rangle$. Then, servers send back the response vectors to the user.

The file is divided into parts, and a part is obtained in each round. With the final round, all parts of the file are completed. For getting the whole file, all the parts of the file from several servers are gathered. Therefore, the PIR rate is defined as the ratio of the information obtained during the process to the downloaded information.

If $t$-colluding servers communicate with each other and they cannot access any information about the desired file, it is said that the PIR scheme is resistant to $t$-colluding servers. The following theorem is the key for finding the number of colluding servers and the system’s PIR rate.

Theorem 2 (\cite{3}). If the automorphism groups of $C$ and $C \star D$ are transitive on the set $\{1, \ldots, n\}$, then there exists a PIR scheme with rate $\frac{\dim(C \star D)}{n} - 1$ that resists a $(d_{D,1} - 1)$-collusion attack (that is, the privacy is $t = d_{D,1} - 1$).

We will compare Reed-Muller codes, considered in \cite{3}, and cyclic codes in Section IV. For this reason, the next section gives a brief exposition of RM codes.

III. REED-MULLER CODES

The binary $r^{th}$ order Reed-Muller Code, denoted by $RM(r, m)$, is defined to be $RM(r, m) := \{ ev(f) : f \in \mathbb{F}_2[x_1, \ldots, x_m], \deg(f) \leq r \}$, where $ev(f)$ is the evaluation of $f$ at all points in $\mathbb{F}_2^m$.

Remark 3. $RM(r, m)$ is a linear code of length $n = 2^m$, dimension $k = \sum_{i=0}^{r} \binom{m}{i}$, and minimum distance $2^{m-r}$ \cite{10}. One has that $RM(r_1, m) \star RM(r_2, m) = RM(r_1 + r_2, m)$, where $r_1 + r_2 \leq m$.

Since if we shorten or puncture RM codes at the position evaluating at 0 we obtain cyclic codes \cite{11}, the following two definitions will be helpful in Section IV.

Definition 4. The shortened code at the position evaluating at 0 of a binary Reed-Muller code is denoted by the linear $[2^m - 1, k - 1, 2^{m-r}]$ code $C_s$. The punctured code at the position evaluating at 0 of a binary Reed-Muller code is denoted by the linear $[2^m - 1, k, 2^{m-r} - 1]$ code $C^s$.

IV. CYCLIC CODES

In this section, we will be concerned with basic cyclic code definitions and compute the star product of two cyclic codes.

Definition 5. A $[n, k]$ linear code $C$ is said to be cyclic if every cyclic shift of a codeword $c = (c_0, c_1, \ldots, c_{n-1}) \in C$ is also codeword in $C$, that is $c = (c_{n-1}, c_0, \ldots, c_{n-2}) \in C$.

Theorem 6 (\cite{12}). A linear code $C$ is a cyclic code if and only if $C$ is isomorphic, as $\mathbb{F}_q$ linear spaces, to an ideal in the ring $R_n = \mathbb{F}_q[x]/(x^n - 1)$.

Definition 7. Let $C$ be a cyclic code in $R_n$. We call $g(x)$ the generator polynomial of $C$ if there exists a unique monic polynomial $g(x)$ such that $C = \langle g(x) \rangle$. Clearly, $g(x)$ is a divisor of $x^n - 1$ in $\mathbb{F}_q[x]$.

Definition 8. A set $J \subseteq \{ 0, 1, \ldots, n - 1 \}$ is said to be primitive if $C = \langle g(x) \rangle$ if $J = \{ j \in \mathbb{Z}/n\mathbb{Z} \mid g(x^j) = 0 \}$ and $I$ is called the generating set of $C = \langle g(x) \rangle$ if $I = \{ j \in \mathbb{Z}/n\mathbb{Z} \mid g(x^j) \neq 0 \}$.

Remark 9. Let $g(x)$ be a generator polynomial of cyclic code $C$ then $g(x) = \prod_{j \in I}(x - \alpha^j)$ and $g(x) = \frac{x^n - 1}{\prod_{j \in I}(x - \alpha^j)}$. Furthermore, $\dim(C) = n - |J| = |I|$.

Definition 10. The cyclotomic coset containing $s$, denoted by $U_s$, is defined to be the set $\{ s, sq, \ldots, s^q \}$ mod $n$ where $q$ is the smallest integer such that $q^i \equiv 1 \pmod{n}$.

We have the following result about the star product of cyclic codes.

Theorem 11. Let $I_1$ and $I_2$ be the generating sets of the cyclic codes $C$ and $D$, respectively. The star product of $C \star D$ is generated by $g_{C \star D} = \frac{x^n - 1}{\prod_{i \in I_1 + I_2}(x - \alpha^i)}$, where $+$ is the Minkowski sum on sets, that is, $I_1 + I_2 := \{ i_1 + i_2 \mid i_1 \in I_1, i_2 \in I_2 \}$.

Proof. The proof follows in a similar way to \cite{9} Theorem III.3.

□

Proposition 12 (BCH Bound). Let $J$ be a defining set of cyclic code $C$ with minimum distance $d$. If $J$ contains $\delta - 1$
consecutive elements \(\{i, \ldots, i + \delta - 2\} \subseteq J\), where \(i, \delta \in \mathbb{Z}/n\mathbb{Z}\), then \(d \geq \delta\).

The following example illustrates the previous statements on cyclic codes.

**Example 13.** Set \(q = 2\), \(n = 31\). Let \(I_C\) be a generating set and let \(J_C\) be a defining set of the code \(C\). Let us compute some cyclotomic cosets with respect to 2, modulo 31:

\[ U_0 = \{0\}, U_1 = \{1, 2, 4, 8, 16\}, U_3 = \{3, 6, 12, 17, 24\}. \]

Let \(J_C = U_0 \cup U_1 \cup U_2\), hence \(\{0, 1, 2, 3, 4\}\) is a set of consecutive roots which means that the BCH bound of \(C\) is equal to 6. The dimension \(k = |J_C| = 31 - 11 = 20\). Therefore, the parameters of this code are \([31, 20, 6]\).

V. PIR SCHEMES FROM CYCLIC CODES

In this section, we focus on cyclic codes towards obtaining a PIR scheme over small fields and compute the code parameters with cyclotomic cosets. First, we will analyze the codes obtained from computer search, their cyclotomic cosets, PIR rates, and the number of colluding servers.

The formulation of the amount of colluding servers and the PIR rate is given in Theorem 2. This theorem is valid for PIR schemes arising from cyclic codes since the automorphism group of a cyclic code is also transitive [13]. Table II gives some cyclic codes, the corresponding rate of the PIR scheme arising from them, and the number of colluding servers of these codes, that is, the privacy parameter \(t\).

| \(C\) | \(D\) | \(D^*\) | \(C \ast D\) | \((C \ast D)^*\) | Privacy | Rate |
|---|---|---|---|---|---|---|
| \([127, 8, 63]\) | \([127, 26, 61]\) | \([127, 126, 105]\) | \([127, 14, 65]\) | \([127, 14, 65]\) | 4/127 | 12 |
| \([127, 8, 64]\) | \([127, 27, 62]\) | \([127, 127, 106]\) | \([127, 14, 66]\) | \([127, 14, 66]\) | 4/127 | 12 |
| \([127, 8, 58]\) | \([127, 28, 63]\) | \([127, 128, 107]\) | \([127, 14, 67]\) | \([127, 14, 67]\) | 4/127 | 12 |
| \([127, 14, 65]\) | \([127, 29, 64]\) | \([127, 129, 108]\) | \([127, 14, 68]\) | \([127, 14, 68]\) | 4/127 | 12 |
| \([127, 21, 48]\) | \([127, 21, 48]\) | \([127, 112, 90]\) | \([127, 15, 55]\) | \([127, 15, 55]\) | 5/15/127 | 8 |

TABLE I

**Computer search experiments**

For instance, the first row in Table II considers \(C\) as an storage code with parameters \([127, 8, 63]\) and \(D\) as a retrieval code with parameters \([127, 26, 61]\). Applying Theorem 2, we can conclude that this system is secure against 9-colluding servers since \(d(D^*) = 10\) and that PIR’s rate is \(r = \frac{\text{dim}(C \ast D)^*}{n} = 14/127\).

We have obtained the codes in Table II by computer search. Consider the codes in the first row, the generating set of \(C\) consists of the union of the cyclotomic cosets \(U_1\) and \(U_{31}\), and the one of \(D\) consists of \(U_0, U_2, U_{23}, U_27, U_{31}\). As mentioned before, the generating set of star products of cyclic codes are given by the Minkowski sum of their generating sets. Hence, the generating set of \(C \ast D\) consists of all cyclotomic cosets except \(U_{13}\) and \(U_{47}\).

Table II classifies the codes in Table II according to the best known linear codes in database [14], which gives lower and upper bounds on the parameters of linear codes. As it is shown in the table, their parameters are the best known or optimal.

A. Comparison with Punctured RM and Shortened RM Codes

In this part, we will show why cyclic codes may provide better performance than RM codes. Even though a RM code \(C\) code is not cyclic, \(C^\ast\) and \(C^\ast\) are cyclic codes [11]. Therefore, we compare the PIR rate and privacy given by a cyclic code with the corresponding punctured and shortened RM codes.

First, let us focus on the comparison with punctured RM codes. For length 127, we fixed as storage code a [127, 8, 63] cyclic code and collected the star product of some codes in Table III. We remark that the BCH bound of all these codes equals their minimum distance.

| \(C\) | \(D\) | \(D^*\) | \(C \ast D\) | \((C \ast D)^*\) | Privacy | Rate |
|---|---|---|---|---|---|---|
| \([127, 8, 63]\) | \([127, 127, 105]\) | \([127, 14, 65]\) | \([127, 14, 65]\) | \([127, 14, 65]\) | 4/127 | 12 |
| \([127, 8, 64]\) | \([127, 127, 106]\) | \([127, 14, 66]\) | \([127, 14, 66]\) | \([127, 14, 66]\) | 4/127 | 12 |
| \([127, 8, 58]\) | \([127, 127, 107]\) | \([127, 14, 67]\) | \([127, 14, 67]\) | \([127, 14, 67]\) | 4/127 | 12 |
| \([127, 14, 65]\) | \([127, 127, 108]\) | \([127, 14, 68]\) | \([127, 14, 68]\) | \([127, 14, 68]\) | 4/127 | 12 |
| \([127, 21, 48]\) | \([127, 127, 90]\) | \([127, 21, 48]\) | \([127, 15, 55]\) | \([127, 15, 55]\) | 5/15/127 | 8 |

TABLE III

**Comparison with punctured, RM codes**

Unbold rows in Table III display the parameters of those codes obtained by the star product of two cyclic codes, equivalent to the punctured RM. Bold rows are obtained by the star product of cyclic code and the fixed code \(C\). Consequently, when the rate and the storage codes are fixed, cyclic codes provide the same parameters as punctured RM ones. However, for a fixed-length \(n\), the dimension of RM codes overgrow, so for a fixed \(C \ast D\), there are not many values that the dimensions of code \(C\) and \(D\) may take. Hence, the first advantage of using cyclic codes in the PIR scheme is to easily provide a larger constellation of parameters.

As an illustration of this fact, in the fourth and fifth rows of Table III the dimension of \(D\) can be 50 or 57 other than 64, or the \(D\) dimension can be 85 or 92, different than 99. Thus, we have different options for the same rate and privacy.

The following remark will show the method we used for obtaining the codes in Table III.

**Remark 14.** The \(r\)-th order punctured generalized RM code is the cyclic code length \(n = q^{m} - 1\) with generator polynomial

\[ g := \prod_{i=1}^{r}((x - \alpha^i), \text{ where } I = \{i: w_q(i) \leq (q - 1)c\}, \]

for some \(c \in \mathbb{Z}^+\).

Using Equation (1), we have created the unbolded row in Table III. If we add or remove some cyclotomic classes to the punctured RM code’s generating sets, we can get another cyclic code, which provides the same rate and privacy. For instance, in the third row, the generating set of \(C\) is comprised of \(U_0\) and \(U_1\), and \(D\) is comprised of \(U_0, U_1, U_3, U_5, U_9\). The generating set of \(D\) code in the second row consists of \(U_0, U_1, U_5, U_9\) by removing \(U_3\).
We also achieved a second advantage, more privacy, by reducing the dimension of the storage code \( C \), which is not equivalent to punctured Reed-Muller codes. We remark that the upload cost in the PIR scheme can be neglected [8], thus we focus on the value \( d(D^\perp) - 1 \), which provides privacy, and on \( \text{dim}(C \ast D)^+ / n \), which gives the PIR rate. Therefore, we can reduce the dimension of code \( C \).

Example 15. Consider \( C_{RM} \ast D_{RM} : [63, 57, 3] \), given by the star product of the punctured RM codes \( C_{RM} : [63, 7, 31], D_{RM} : [63, 42, 7] \). The scheme protects against \( d_{D^\perp} - 1 = 15 \) collusions. Consider now the cyclic code with parameters \([63, 51, 3]\), where the generating set of \( C \) consists of cyclotomic class \( U_{23} \), and the cyclic code \( D \) with parameters \([63, 51, 3]\). One has that \( C \ast D = C_{RM} \ast D_{RM} \). In this case \( d_{D^\perp} - 1 = 19 \). Therefore, our cyclic code proposal protects against a more significant number of colluding servers for the same rate.

Table [IV] considers more examples where the dimension of the storage code has been reduced. It displays the colluding server and rate reached when \( C \) with length \( n = 255 \) and dimension 2. Note that this scheme, obtained using a Punched RM code, protects against a maximum of 64 colluding servers where \( C : [255, 9, 127], D^\perp : [255, 36, 64] \). Moreover, the PIR rate of this scheme is equal to \( 8/255 \). In Table [V] the code pairs in all rows, except the first one, protect against more than 64-collusion and can even reach 73.

The cyclotomic cosets used for setting up the codes in Table [IV] are given in the Table [VI].

### Table IV
| \( C \)  | \( D \)  | \( D^\perp \) | \( (C \ast D)^+ \) | Privacy | Rate |
|-------|-------|-------|----------------|-------|------|
| 255, 2170 | 255, 192 | 255, 63, 65 | 255, 19 | 64 | 19/255 |
| 255, 2170 | 255, 198 | 255, 36, 64 | 255, 8 | 66 | 8/255 |
| 255, 2170 | 255, 199 | 255, 31, 65 | 255, 11 | 69 | 11/255 |
| 255, 2170 | 255, 201 | 255, 15, 67 | 255, 9 | 7 | 7/255 |
| 255, 2170 | 255, 202 | 255, 8, 68 | 255, 7 | 8 | 8/255 |
| 255, 2170 | 255, 204 | 255, 63, 74 | 255, 11 | 73 | 11/255 |

### Table V
| \( C \)  | \( D \)  |
|-------|-------|
| \( V_{68} \) | \( V \setminus \{ U_0, U_1, U_{11}, U_{17}, U_{23}, U_{29}, U_{35}, U_{41}, U_{47} \} \) |
| \( V \setminus \{ U_1, U_{13}, U_{25}, U_{37}, U_{49}, U_0, U_{11}, U_{22}, U_{33}, U_{44} \} \) | \( V \setminus \{ U_0, U_1, U_{11}, U_{17}, U_{23}, U_{29}, U_{35}, U_{41}, U_{47} \} \) |
| \( V \setminus \{ U_0, U_{11}, U_{17}, U_{23}, U_{29}, U_{35}, U_{41}, U_{47} \} \) | \( V \setminus \{ U_0, U_1, U_2, U_{11}, U_{12}, U_{13}, U_{22}, U_{31}, U_{32}, U_{33}, U_{34} \} \) |

### Table VI
| \( C \)  | \( D \)  | \( D^\perp \) | \( (C \ast D)^+ \) | Privacy | Rate |
|-------|-------|-------|----------------|-------|------|
| 127, 7, 64 | 127, 7, 64 | 127, 7, 64 | 127, 7, 64 | 64 | 127 |
| 127, 7, 64 | 127, 29, 34 | 127, 9, 170 | 127, 64, 148 | 64 | 127 |
| 127, 7, 64 | 127, 59, 21 | 127, 16, 193 | 127, 29, 31 | 15 | 127 |
| 127, 7, 64 | 127, 67, 21 | 127, 40, 234 | 127, 29, 31 | 15 | 127 |
| 127, 7, 64 | 127, 79, 11 | 127, 67, 21 | 127, 29, 31 | 15 | 127 |
| 127, 7, 64 | 127, 97, 1 | 127, 35, 32 | 127, 119, 1 | 127, 8, 32 | 31 | 127 |

One has that cyclic codes protect against one more colluding server than shortened RM, as it can be seen at Table [VI]. Moreover, we have a larger constellation of possible parameters. For instance, for a case rate equal to 29/127, a cyclic code protects against 15-collusion, but the shortened RM codes protect against 14-collusion.

### VI. Conclusion

By using cyclic codes, we provide binary PIR schemes with colluding servers in the fashion of \( \mathbb{I} \). We provide a family of optimal binary PIR schemes. Our PIR schemes have the advantage, with respect to PIR schemes from MDS codes, that they can be defined over a binary field. Moreover, they provide a larger constellation of parameters than the binary PIR schemes using Reed-Muller codes and they even outperform them in some cases.

### References

[1] R. Freij-Hollanti, O. W. Gnilke, C. Hollanti, A.-L. Hoerlemann-Trautmann, D. Karpuk, and I. Kubias, “t-private information retrieval schemes using transitive codes,” IEEE Transactions on Information Theory, vol. 65, no. 4, pp. 2107–2118, 2019.

[2] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, “Private information retrieval,” J. ACM, vol. 45, no. 6, pp. 965–982, 1998.

[3] N. B. Shah, K. V. Rashmi, and K. Ramchandran, “One extra bit of download ensures perfectly private information retrieval,” in 2014 IEEE International Symposium on Information Theory, 2014, pp. 856–860.

[4] R. Freij-Hollanti, O. W. Gnilke, C. Hollanti, and D. A. Karpuk, “Private information retrieval from coded databases with colluding servers,” SIAM J. Appl. Algebra Geom., vol. 1, no. 1, pp. 647–664, 2017.

[5] R. Tajeddine, O. W. Gnilke, and S. el Rouayheb, “Private information retrieval from mds coded data in distributed storage systems,” IEEE Transactions on Information Theory, vol. 64, pp. 7081–7093, 2018.

[6] H. Sun and S. A. Jafar, “The capacity of private information retrieval,” IEEE Trans. Inform. Theory, vol. 63, no. 7, pp. 4075–4088, 2017.

[7] ——, “The capacity of robust private information retrieval with colluding databases,” IEEE Trans. Inform. Theory, vol. 64, no. 4, part 1, pp. 2361–2370, 2018.

[8] K. Banawan and S. Ulukus, “The capacity of private information retrieval from coded databases,” IEEE Transactions on Information Theory, vol. 64, no. 3, pp. 1945–1956, 2018.

[9] I. Casco, “On squares of cyclic codes,” IEEE Trans. Inform. Theory, vol. 65, no. 2, pp. 1034–1047, 2019.

[10] F. J. MacWilliams and N. J. A. Sloane, “The theory of error-correcting codes,” 1977.

[11] A. D. Yardi and R. Pellikaan, “On shortened and punctured cyclic codes,” ArXiv, vol. abs/1705.09859, 2017.

[12] J. van Lint, Introduction to Coding Theory, ser. Graduate Texts in Mathematics. Springer Berlin Heidelberg, 1998.

[13] W. C. Huffman and V. Pless, Fundamentals of Error-Correcting Codes. Cambridge University Press, 2003.

[14] M. Grassl, “Bounds on the minimum distance of linear codes and quantum codes,” Online available at http://www.codetables.de, 2007, accessed on 2021-11-03.