Lifted Reed-Solomon Codes and Lifted Multiplicity Codes

Lukas Holzbaur, Rina Polyanskaya, Nikita Polyanskii, Ilya Vorobyev, and Eitan Yaakobi

Abstract

Lifted Reed-Solomon and multiplicity codes are classes of codes, constructed from specific sets of m-variate polynomials. These codes allow for the design of high-rate codes that can recover every codeword or information symbol from many disjoint sets. Recently, the underlying approaches have been combined for the bi-variate case to construct lifted multiplicity codes, a generalization of lifted codes that can offer further rate improvements. We continue the study of these codes by first establishing new lower bounds on the rate of lifted Reed-Solomon codes for any number of variables m, which improve upon the known bounds for any $m \geq 4$. Next, we use these results to provide lower bounds on the rate and distance of lifted multiplicity codes obtained from polynomials in an arbitrary number of variables, which improve upon the known results for any $m \geq 3$. Specifically, we investigate a subcode of a lifted multiplicity code formed by the linear span of m-variate monomials whose restriction to an arbitrary line in $\mathbb{F}_q^m$ is equivalent to a low-degree univariate polynomial. We find the tight asymptotic behavior of the fraction of such monomials when the number of variables m is fixed and the alphabet size $q = 2^\ell$ is large.

Using these results, we give a new explicit construction of batch codes utilizing lifted Reed-Solomon codes. For some parameter regimes, these codes have a better trade-off between parameters than previously known batch codes. Further, we show that lifted multiplicity codes have a better trade-off between redundancy and the number of disjoint recovering sets for every codeword or information symbol than previously known constructions, thereby providing the best known PIR codes for some parameter regimes. Additionally, we present a new local self-correction algorithm for lifted multiplicity codes.

Index Terms

Lifted Codes, Multiplicity Codes, Lifted Multiplicity Codes, Batch Codes, Reed-Solomon Codes, Distributed Storage Systems, Disjoint Recovering Sets

I. Introduction

The concepts of *locality* and *availability* of codes have been subject to intensive studies. Informally, the locality of a code refers to the number of codeword symbols that need to be accessed in order to recover a single codeword or information symbol and availability is the number of such (disjoint) recovery sets. These properties are of interest in a variety of applications, such as load balancing in distributed data storage, cryptography, and low-complexity error correction/detection. Several different notions related to these parameters have been considered in the literature, including, but not limited to, locally recoverable codes (LRCs) [3], [4], locally decodable/correctable codes (LDCs/LCCs) [5], [6], relaxed LCCs [7] and LDCs [8], batch codes [9], PIR codes [10], and codes with the disjoint repair group property (DRGP) [11].

Reed-Muller (RM) codes are a popular class of codes that can provide strong locality and availability properties, as already exploited in the early majority-logic decoding algorithms [12]. These codes are defined as the evaluation of multi-variate polynomials up to a specific degree in all points of a multidimensional space. Their restriction to the evaluation points that fall on one line in this evaluation space can readily be seen to be equivalent to the evaluation of a univariate polynomial in the variable over the one-dimensional space spanned by this line. If the degree of this univariate polynomial is low, these positions form a codeword of a (non-trivial) Reed-Solomon (RS) code, another well-studied class of evaluation codes. This principle can be exploited to show locality and availability properties of the RM code, which have been subject to extensive study (see, e.g., [13]–[15]). However, the obvious drawback of RM codes with nice local recovery properties is their rather low rate of $R \leq 1/2$.

To overcome this issue of low rate, the concept of *lifted RS codes* was introduced in [16]. Instead of evaluating only multi-variate polynomials of a limited degree, as in RM codes, these codes consist of the evaluation of all polynomials that are

The results on lifted RS codes have partially been presented at the IEEE International Symposium on Information Theory (ISIT) 2020 [1] and parts of the results on lifted multiplicity codes have partially been presented at the IEEE Information Theory Workshop (ITW) 2020 [2].

L. Holzbaur's work was supported by the Technical University of Munich – Institute for Advanced Study, funded by the German Excellence Initiative and European Union 7th Framework Programme under Grant Agreement No. 291763 and the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under Grant No. WA3907/1-1. N. Polyanskii's work was supported by a grant from the Russian Science Foundation (grant no. 19-71-00137). E. Yaakobi's work was supported in part by the Israel Science Foundation under Grant No. 1817/18 and by the Technion Hiroshi Fujiwara Cyber Security Research Center and the Israel National Cyber Directorate.

L. Holzbaur is with the Institute for Communications Engineering, Technical University of Munich, Germany. R. Polyanskaya is with the Institute for Information Transmission Problems, Russian Academy of Sciences, Russia. N. Polyanskii is with the Center for Computational and Data-Intensive Science and Engineering, Skolkovo Institute of Science and Technology, Russia, and the Institute for Communications Engineering, Technical University of Munich, Germany. I. Vorobyev is with the Center for Computational and Data-Intensive Science and Engineering, Skolkovo Institute of Science and Technology, Russia. E. Yaakobi is with the Computer Science Department, Technion — Israel Institute of Technology, Israel.

Emails: lukas.holzbaur@tum.de, rev-rina@yandex.ru, nikita.polyansky@gmail.com, vorobyev.i.v@yandex.ru, yaakobi@cs.technion.ac.il

equivalent to the evaluation of a low-degree univariate polynomial when restricted to a line. Using this concept of lifting, which first appeared in [17] in the context of LDPC codes, [16] presents constructions of codes from multi-variate polynomials along with good bounds on the redundancy for the bi-variate case. These codes are of considerably higher rate than RM codes, while, broadly speaking, preserving the locality properties of the RM code. The main highlight of these codes is the construction of high-rate high-error LCCs. As a conceptual result, it was shown in [16] that any polynomial producing a codeword of the lifted RS code can be decomposed to a linear combination of *good* monomials whose restriction to lines are low-degree. Thus, the code rate is equal to the *fraction* of good monomials. We remark that the distance properties of these codes follow from the fact that each symbol has many disjoint recovering sets and, thus, the relative distance of lifted RS codes is similar to the one of RM codes.

Multiplicity codes [18] are another recently introduced class of codes with good locality properties based on RM codes. Here, each codeword symbol not only consists of the evaluation of a degree-restricted multi-variate polynomial, but it also contains the evaluation of all the derivatives of this polynomial up to some order. Similar to the concept of lifting, this generalization provides codes with significantly better rate than RM codes, while providing good locality properties. In particular, it was proved [18] that multiplicity codes represent a family of high-rate LCCs that have very efficient local decoding algorithms. The analysis of the rate of multiplicity codes is rather straightforward, whereas distance properties are implied by a bound on the number of points that a low-degree polynomial can vanish on with high multiplicity.

As both lifted RS codes and multiplicity codes are based on generalizations of RM codes, it is a natural question whether these techniques can be combined to further improve the parameters of the respective codes. Some progress in the study of these *lifted multiplicity codes* has recently been made in [11], [19]. In [19], the authors show asymptotic results for any number of variables. Paper [11] is devoted to improving the existing bounds on the required redundancy in the bi-variate case.

A. Our contribution

In this work we continue the study of lifted RS codes and lifted multiplicity codes by generalizing the results on the bi-variate case of [11], [16] to an arbitrary number of variables. Since lifted RS codes represent a specific class of lifted multiplicity codes, when derivatives are not taken into account, we focus on the description of lifted multiplicity codes in the following. Essentially, we investigate the same class of codes as defined in [11], [19]. Informally, the $[m, s, d, q]$ lifted multiplicity code consists of the evaluation (together with the derivatives up to the sth order) of polynomials from $\mathbb{F}_q[X_1, \dots, X_m]$ whose restriction to a line agrees with some polynomial of degree less than d on its first $s-1$ derivatives. Note that the condition $d < qs$ guarantees [11], [19] that the all-zero codeword is produced only by the zero polynomial and, therefore, we fix $d = qs - r$ for some integer r.

Following a standard approach, we consider a subcode of a lifted multiplicity code which is formed by the linear span of *good* monomials whose restriction to a line is equivalent to a low-degree polynomial. To count bad monomials, we first make use of the result for lifted RS codes ($s = 1$) derived in Section III and then extend it for larger s. Roughly speaking, we prove that there exists a one-to-$\binom{s+m-1}{m-1}$ correspondence between bad monomials for lifted RS codes and groups of bad monomials for lifted multiplicity codes. This enables us to find the exact asymptotic order of the number of bad monomials when q is large (for more details, see Section IV-B). Unfortunately, unlike lifted RS codes, there is no nice structural result saying that a good polynomial of a lifted multiplicity code can be decomposed into a linear combination of good monomials (for a counterexample see Appendix B). However, the fraction of good monomials serves as a lower bound on the rate of a lifted multiplicity code. Compared to prior works, our estimate for lifted RS codes is consistent with [16] for $m = 2$, with [20] for $m = 3$, and better than the result of [16] for any $m > 2$. As for lifted multiplicity codes, our estimate is consistent with [11] for $m = 2$ and better than the result of [19] for any $m \geq 2$.

Let $\binom{m}{\geq b}$ denote the number of ways to choose an (unordered) set of at least b elements from a fixed set of size m. Our main contribution is summarized in the following statement.

Theorem (Parameters of lifted multiplicity code)**.**
Code rate: *For powers of two q and $s < q$ and a positive integer $r < q$, the rate of the $[m, s, qs - r, q]$ lifted multiplicity code is*

$$1 - O_m\left(s^{-1}(q/r)^{\log \lambda_m - m}\right) \quad \text{as } q \to \infty,$$

where λ_m is the largest eigenvalue of the matrix

$$A_m := \begin{pmatrix} \binom{m}{\geq 1} & \binom{m}{0} & 0 & 0 & \dots & 0 \\ \binom{m}{\geq 3} & \binom{m}{2} & \binom{m}{1} & \binom{m}{0} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \binom{m}{\geq 2j+1} & \binom{m}{2j} & \binom{m}{2j-1} & \binom{m}{2j-2} & \dots & \binom{m}{2j-m+2} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \binom{m}{\geq 2m-1} & \binom{m}{2m-2} & \binom{m}{2m-3} & \binom{m}{2m-4} & \dots & \binom{m}{m} \end{pmatrix}.$$

TABLE I: The largest eigenvalue λ_m of A_m, the resulting convergence rate $m-\log(\lambda_m)$ derived in Section III, and the convergence rate p_m of [16] for different values of m.

m	λ_m	$m-\log(\lambda_m)$	p_m
2	3.0000	4.1504×10^{-1}	4.1504×10^{-1}
3	7.2361	1.4479×10^{-1}	1.1360×10^{-2}
4	15.5436	4.1747×10^{-2}	2.8233×10^{-3}
5	31.7877	9.6043×10^{-3}	4.6986×10^{-4}
6	63.9217	1.7653×10^{-3}	1.1742×10^{-4}
7	127.9763	2.6714×10^{-4}	2.9353×10^{-5}
8	255.9939	3.4467×10^{-5}	2.8664×10^{-8}
9	511.9986	3.8959×10^{-6}	2.6872×10^{-9}
10	1023.9997	3.9323×10^{-7}	3.3590×10^{-10}

Distance: *For $r,s<q$, the relative distance Δ of the $[m,s,qs-r,q]$ lifted multiplicity code is*

$$\Delta\geq\Delta_{min}\coloneqq\left\lceil\frac{r-s+1}{s}\right\rceil\frac{q-s}{q^2}.$$

For $s=o(r)$, $\Delta_{min}=\frac{r}{qs}(1+o(1))$.

Availability: *Each symbol of a codeword of the $[m,s,qs-s,q]$ lifted multiplicity code can be reconstructed in $\lfloor q/s\rfloor^{m-1}$ different ways, each of which involves a disjoint set of coordinates of the codeword with cardinality $s^{m-1}(q-1)$.*

Local self-correction: *For $s^{m-2}=o(\log q)$ and $r<q$, let $\mathbf{y}$ be a noisy version of a codeword $\mathbf{c}$ of the $[m,s,qs-r,q]$ lifted multiplicity code such that the relative distance $\Delta(\mathbf{y},\mathbf{c})<\alpha\Delta_{min}$ with $0<\alpha<1/4$. Then for any $i\in[q^m]$, there exists a randomized algorithm $\mathfrak{A}$ that makes at most $(q-1)s^{m-1}$ queries to $\mathbf{y}$ and reconstructs c_i correctly with probability at least $1-2\alpha+o(1)$.*

We have several additional remarks and comments illustrating the contribution of our paper.

- The advantage of moving from lifted RS codes to lifted multiplicity codes is that the redundancy improves by a factor of s (the order of derivatives), while the number of repair groups gets worse by a factor of s^{m-1} and the logarithm of the alphabet size gets bigger by a factor of $\binom{s+m-1}{m}$. This means that lifted multiplicity codes cover more parameters of codes with good locality properties than lifted RS codes. For a relevant comparison, see the remarks after Lemmas 7-8.
- Let us demonstrate the improvement in the rate of the $[m,s,qs-r,q]$ lifted multiplicity codes compared to the rate of the multiplicity code of order-s evaluations of degree $qs-r$ polynomials in m variables over $\mathbb{F}_q$ [18, Lemma 7]. Both types of codes have the same estimate on the relative distance $\Delta\geq\frac{r}{qs}(1+o(1))$. However, the rate of the multiplicity code is

$$\frac{\binom{qs-r+m}{m}}{\binom{s+m-1}{m}q^m}<\left(\frac{qs-r+m}{(s+1/3)q}\right)^m\leq1-\Omega_m\left(s^{-1}\right),$$

  which is smaller than the rate of lifted multiplicity codes as $\log\lambda_m<m$. Here, we point out that for large m, we are able to find the technical parameter λ_m numerically only. We depict some values of λ_m in Table I. This parameter stands for the exponential growth of the number of bad monomials. The inequality $\log\lambda_m<m$ follows from [16] implicitly, as the true exponent $\log\lambda_m$ was estimated by $m-p_m<m$, where $p_m:=-\log\left(1-2^{-m\lceil\log m\rceil}\right)/\lceil\log m\rceil$. On the other hand, it is possible to estimate $\log\lambda_m$ from the other side as follows

$$p_m\leq m-\log\lambda_m\leq-\log(1-2^{-m}) \tag{1}$$

  and, thus, $m-\log\lambda_m>0$ vanishes as $m\to\infty$.
- Observe that if a good polynomial and its derivatives do not vanish on a point, then it is still possible that the restrictions of the polynomial to some lines containing this point are equivalent to the zero polynomial. This fact was overlooked in [19] when proving the distance property of lifted multiplicity codes. However, we can always say that the restriction of the polynomial to at least $(q-s)q^{m-2}$ lines crossing this point is equivalent to a non-zero univariate polynomial of degree less than $qs-r$ and, thus, the minimum distance of the code is at least $1+\lceil r/s-1\rceil(q-s)q^{m-2}$ (for more details, see Section IV-B).
- Note that the self-correction algorithm for multiplicity codes from [18] works well for lifted multiplicity codes. However, for small enough s, we present a slightly different local self-correction algorithm which requires $s\,5^m$ times less locality. Here we combine two ideas: 1) for recovering of the evaluation of a polynomial and its derivatives up to the sth order at a point, it is sufficient to know directional derivatives for s^{m-1} lines containing the point whose directional vectors $(1,v_2,\dots,v_m)$ form a subcube $1\times Q_2\times\cdots\times Q_m$ with $Q_i\subset\mathbb{F}_q$, $|Q_i|=p$; 2) every $(m-1)$-uniform hypergraph with q vertices in each part with at least εq^{m-1} hyperedges contains a copy of $(m-1)$-uniform clique with s vertices in each part (for more details, see Section VII-B).

B. Outline

The remainder of the paper is organized as follows. In Section II, we give rigorous definitions of lifted RS codes and lifted multiplicity codes along with some auxiliary notation. The rate of lifted RS codes can be determined by computing the fraction of so-called *good monomials*, for which we will derive tight asymptotic formulas in Section III. Using the latter result, in Section IV, we derive bounds on the rate and distance of lifted multiplicity codes. In Sections V, VI, and VII, we apply the results of Sections III and IV results to PIR codes, batch codes, and LCCs, respectively. Finally, we conclude with Section VIII.

II. Preliminaries

A. Notation

We start by introducing some notation that is used throughout the paper. For some functions $f(x)$ and $g(x)$, we write $f(x) = O(g(x))$ and $f(x) = \Omega(g(x))$ as $x \to \infty$ if there exists some real x_0 and C such that $|f(x)| \leq C|g(x)|$ and $|f(x)| \geq C|g(x)|$ for $x \geq x_0$, respectively. If both equalities $f(x) = O(g(x))$ and $f(x) = \Omega(g(x))$ hold, we use the notation $f(x) = \Theta(g(x))$. Also, we write $f(x) = o(g(x))$ as $x \to \infty$ if for every positive ε there exists some real x_0 such that $|f(x)| \leq \varepsilon|g(x)|$ for $x \geq x_0$. In these notations, we use a subscript, such as $O_m(f(x))$, if the parameter m is to be regarded as fixed.

Let $[n]$ be the set of integers from 1 to n. We use uppercase letters such as T and X to denote variables. A vector is denoted by bold letters, e.g., $\mathbf{d}$ is a vector over a field or a ring and $\mathbf{X}$ is a vector of variables. Let $q = 2^\ell$ and $\mathbb{F}_q$ be a field of size q. We write $\log x$ to denote the logarithm of x in base two. By $\mathbb{Z}_{\geq}$ and $\mathbb{Z}_n$ denote the set of non-negative integers and the set of integers between 0 and $n-1$, respectively. In what follows, we fix m to be a positive integer representing the number of variables. For $\mathbf{d} = (d_1, \ldots, d_m) \in \mathbb{Z}_q^m$ and $\mathbf{X} = (X_1, \ldots, X_m)$, let $\mathbf{X}^{\mathbf{d}}$ denote the monomial $\prod_{i=1}^m X_i^{d_i}$ from $\mathbb{F}_q[\mathbf{X}]$. Let $\deg(\mathbf{d})$ be the sum of components of $\mathbf{d} \in \mathbb{Z}_{\geq}^n$ and $|\mathbf{d}|$ be the number of non-zero components of $\mathbf{d}$. Additionally, we define $\deg_q(\mathbf{d}) := \sum_{i=1}^m \lfloor d_i/q \rfloor$. For a vector $\mathbf{i} \in \mathbb{Z}_{\geq}^m$, let $[\mathbf{X}^{\mathbf{i}}]f(\mathbf{X})$ denote the coefficient of $\mathbf{X}^{\mathbf{i}}$ in the polynomial $f(\mathbf{X})$. For $f(\mathbf{X}) \in \mathbb{F}_q[\mathbf{X}]$, we define $\deg(f)$ to be the maximal $\deg(\mathbf{i})$ for $\mathbf{i}$ such that $[\mathbf{X}^{\mathbf{i}}]f(\mathbf{X})$ is non-zero.

Let us define a partial order relation on $\mathbb{Z}_q$. For two integers $a = \sum_{i=0}^{\ell-1} a^{(i)}2^i$ and $b = \sum_{i=0}^{\ell-1} b^{(i)}2^i$ with $a^{(i)}, b^{(i)} \in \{0,1\}$ we write $a \leq_2 b$ if $a^{(i)} \leq b^{(i)}$ for all $i \in \{0, \ldots, \ell-1\}$. We denote $a = (a^{(\ell-1)}, ..., a^{(0)})_2$. For vectors $\mathbf{d}, \mathbf{d}' \in \mathbb{Z}_q^m$, we write $\mathbf{d} \leq_2 \mathbf{d}'$ if $d_i \leq_2 d_i'$ for all $i \in [m]$.

Define an operation $(\mathrm{mod}_s^*\ q)$ that takes a non-negative integer and maps it to the element from $\mathbb{Z}_{qs}$ as follows

$$a\ (\mathrm{mod}_s^*\ q) := \begin{cases} a, & \text{if } a \in \mathbb{Z}_s, \\ b \in \mathbb{Z}_{qs} \setminus \mathbb{Z}_s, & \text{if } a \notin \mathbb{Z}_s,\ a = b\ (\mathrm{mod}\ qs - s). \end{cases}$$

If $s = 1$, we drop the index and write $(\mathrm{mod}^*\ q)$ instead of $(\mathrm{mod}_1^*\ q)$. It can be readily seen that if $a\,(\mathrm{mod}^* q) = b$, then $T^a = T^b\ (\mathrm{mod}\ T^q - T)$ in $\mathbb{F}_q[T]$. A similar equivalence for $(\mathrm{mod}_s^*\ q)$ will be defined in Section II-C.

For a function $f : \mathbb{F}_q^m \to \mathbb{F}_q$ and a set $S \subset \mathbb{F}_q^m$ let $f|_S$ denote the restriction of f to the domain S. Abbreviate the set of all lines in $\mathbb{F}_q^m$ by

$$\mathcal{L}_m := \left\{(\mathbf{w} + \mathbf{v}T)|_{T \in \mathbb{F}_q} \text{ for } \mathbf{w}, \mathbf{v} \in \mathbb{F}_q^m\right\}.$$

We note that a multivariate polynomial restricted to a line is an univariate polynomial and the degree of the latter does not depend on the parameterization of the line, i.e., the degree of the univariate polynomial obtained by restricting to a line $L = (\mathbf{w} + \gamma_1\mathbf{v} + \gamma_2\mathbf{v}T)|_{T \in \mathbb{F}_q}$ with $\gamma_1 \in \mathbb{F}_q$ and $\gamma_2 \in \mathbb{F}_q^*$ is independent of the choice of γ_1 and γ_2. Denote the set of univariate polynomials of degree less than d by

$$\mathcal{F}_q(d) := \{f(T) \in \mathbb{F}_q[T] :\ \deg(f) < d\}.$$

B. Lifted Reed-Solomon codes

Let us recall the definition of lifted Reed-Solomon codes introduced in [16].

Definition 1 (Lifted Reed-Solomon code, [16])**.** For integers $m \geq 1$ and $d < q$, the *m-dimensional lift of a Reed-Solomon code* (or the $[m, d, q]$ *lifted RS code*) is the code

$$\left\{(f(\mathbf{a}))|_{\mathbf{a} \in \mathbb{F}_q^m} :\ f(\mathbf{X}) \in \mathbb{F}_q[\mathbf{X}] \text{ such that } \forall L \in \mathcal{L}_m :\ f|_L \in \mathcal{F}_q(d)\right\}.$$

Remark. Note that the one-dimensional lift of a Reed-Solomon code represents the ordinary Reed-Solomon code of length q and dimension d. Also, we observe that the $[m, d, q]$ lifted RS code includes all codewords of the m-variate RM code of order $d-1$ over $\mathbb{F}_q$.

In Appendix C-A we provide a simple example which demonstrates that there exist polynomials contained in the lifted RS code, that are not of low-degree, i.e., not contained in the respective RM code.

Definition 2 (d^*-bad and good monomials)**.** Given a positive integer $d < q$, we say that a monomial $\mathbf{X}^{\mathbf{d}}$ with $\mathbf{d} \in \mathbb{Z}_q^m$ is d^*-*bad* over $\mathbb{F}_q[\mathbf{X}]$ if there exists at least one $\mathbf{i} \in \mathbb{Z}_q^m$ such that $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) \pmod{q} \in \{d, d+1, \ldots, q-1\}$. A monomial is said to be d^*-*good* if it is not d^*-bad.

A characterization of lifting was established in [16]. We make use of this result for lifted Reed-Solomon codes.

Lemma 1 (Follows from [16, Section 2])**.** *The $[m, d, q]$ lifted RS code is equivalently defined as the evaluation of polynomials from the linear span of d^*-good monomials over $\mathbb{F}_q[\mathbf{X}]$.*

Lemma 1 suggests a way to compute the dimension of the $[m, q, d]$ lifted RS code, namely one needs to estimate the size of the set of d^*-good m-variate monomials over $\mathbb{F}_q[\mathbf{X}]$. We carry out a careful analysis of the latter in Section III.

C. Lifted multiplicity codes

Definition 3. For $f(\mathbf{X}) \in \mathbb{F}_q[\mathbf{X}]$ and a vector $\mathbf{i} \in \mathbb{Z}_{\geq}^m$, the $\mathbf{i}$th *(Hasse) derivative* of f, denoted by $f^{(\mathbf{i})}(\mathbf{X})$, is the coefficient $[\mathbf{Y}^{\mathbf{i}}]g(\mathbf{X}, \mathbf{Y})$, where the polynomial $g(\mathbf{X}, \mathbf{Y}) \coloneqq f(\mathbf{X} + \mathbf{Y}) \in \mathbb{F}_q[\mathbf{X}, \mathbf{Y}]$. Therefore, we have

$$g(\mathbf{X}, \mathbf{Y}) = \sum_{\mathbf{i} \in \mathbb{Z}_{\geq}^m} f^{(\mathbf{i})}(\mathbf{X}) \mathbf{Y}^{\mathbf{i}}.$$

For an $\mathbf{x} \in \mathbb{F}_q^m$, an integer $s \geq 1$, and a polynomial $f(\mathbf{X}) \in \mathbb{F}_q[\mathbf{X}]$, we write $f^{(<s)}(\mathbf{x}) \in \mathbb{F}_q^{\binom{s+m-1}{m}}$ to denote the vector containing $f^{(\mathbf{i})}(\mathbf{x})$ for all $\mathbf{i} \in \mathbb{Z}_{\geq}^m$ so that $\deg(\mathbf{i}) < s$. In what follows, we assume that s is a power of two.

We recall two well-known properties of the Hasse derivative which will imply the linearity of lifted multiplicity codes over $\mathbb{F}_q$.

Proposition 1. *Let $f(\mathbf{X}), g(\mathbf{X}) \in \mathbb{F}_q[\mathbf{X}]$, $\lambda \in \mathbb{F}_q$ and let $\mathbf{i} \in \mathbb{Z}_{\geq}^m$. Then we have*

1) $f^{(\mathbf{i})}(\mathbf{X}) + g^{(\mathbf{i})}(\mathbf{X}) = (f+g)^{(\mathbf{i})}(\mathbf{X})$.
2) $(\lambda f)^{(\mathbf{i})}(\mathbf{X}) = \lambda f^{(\mathbf{i})}(\mathbf{X})$.

Definition 4. We say that two univariate polynomials $f(X), g(X) \in \mathbb{F}_q[X]$ are equivalent up to order s if $f^{(<s)}(x) = g^{(<s)}(x)$ for all $x \in \mathbb{F}_q$. To indicate such an equivalence, we write $f(X) \equiv_s g(X)$.

The following statement shows the smallest possible degree of an equivalent polynomial.

Proposition 2 (Lemma 12 in [11])**.** *Let q be a power of two. For every univariate polynomial $f(X)$, there exists a unique degree-at-most $sq-1$ polynomial $g(X)$ such that $f(X) \equiv_s g(X)$. Moreover, if s is a power of two, then $f(X) = g(X) \pmod{X^{qs} + X^s}$ and for all i such that $\deg(f) - qs + s < i < qs$, we have $[X^i]f(X) = [X^i]g(X)$.*

If s is a power of two and $a \pmod{q}_s^* = b$, then $T^a \equiv_s T^b$. Now we give a well-known result about the multiplicities of a multi-variate polynomial.

Lemma 2 (Follows from [21])**.** *Let $f(\mathbf{X})$ be a non-zero polynomial of degree at most d. Then the number of points $\mathbf{x} \in \mathbb{F}_q^m$ such that $f^{(\mathbf{i})}(\mathbf{x}) = 0$ for all $\mathbf{i} \in \mathbb{Z}_{\geq}^m$ with $\deg(\mathbf{i}) < s$ is at most $\lfloor dq^{m-1}/s \rfloor$.*

Definition 5 (Lifted multiplicity code [11])**.** For integers $m \geq 1$ and $d < qs$, the $[m, s, d, q]$ *lifted multiplicity* code over $\mathbb{F}_q^{\binom{s+m-1}{m}}$ of length q^m is defined as

$$\left\{ \left(f^{(<s)}(\mathbf{a}) \right)\Big|_{\mathbf{a} \in \mathbb{F}_q^m} \; : \; \begin{array}{l} f(\mathbf{X}) \in \mathbb{F}_q[\mathbf{X}] \text{ such that} \\ f|_L \equiv_s g(T) \;\forall\; L = L(T) \in \mathcal{L}_m \\ \text{for some } g \in \mathcal{F}_q(d) \end{array} \right\}.$$

Remark. Multiplicity codes, as defined in [18], consist of the evaluations of multi-variate polynomials of degree $< d$. These polynomials trivially fulfill the condition that their restriction to every line $L \in \mathcal{L}_m$ is a polynomial of degree $< d$. It follows that the $[m, s, d, q]$ multiplicity code is a subcode of the $[m, s, d, q]$ lifted multiplicity code and thereby that the dimension of a lifted multiplicity code is lower bounded by the dimension of the corresponding multiplicity code. However, for many parameters, lifting increases the rate of the multiplicity code, as we formally show in Section IV. To provide some further intuition, we also give an example for this improvement in Appendix C-B.

Definition 6 ($(d,s)^*$-bad and good monomials)**.** Given positive integers s and d, we say that a monomial $\mathbf{X}^{\mathbf{d}}$ with $\mathbf{d} \in \mathbb{Z}_{qs}^m$ and $\deg_q(\mathbf{d}) \leq s-1$ is $(d,s)^*$-*bad* over $\mathbb{F}_q[\mathbf{X}]$ if there exists at least one $\mathbf{i} \in \mathbb{Z}_{qs}^m$ such that $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) \pmod{q}_s^* \in \{d, d+1, \ldots, qs-1\}$. A monomial $\mathbf{X}^{\mathbf{d}}$ with $\mathbf{d} \in \mathbb{Z}_{qs}^m$ and $\deg_q(\mathbf{d}) \leq s-1$ is said to be $(d,s)^*$-*good* if it is not $(d,s)^*$-bad.

Let $\mathcal{F}_q(m, s, d)$ be the collection of $(d,s)^*$-good m-variate monomials from $\mathbb{F}_q[\mathbf{X}]$.

Proposition 3. *For $s \leq q$ and $d < qs$, the cardinality of the $[m, s, d, q]$ lifted multiplicity code is $\geq q^{|\mathcal{F}_q(m,s,d)|}$.*

Proof. The full proof of this technical statement is given in Appendix A. There we show that different linear combinations of good monomials produce different codewords and that these codewords are contained in the $[m, s, d, q]$ lifted multiplicity code. Thus, the lower bound on the dimension of the code follows directly from the number of good monomials $|\mathcal{F}_q(m, s, d)|$. ■

Remark. Observe that for $s = 1$, Definition 5 gives exactly the code spanned by the evaluation of good monomials, i.e., the statement of Proposition 3 holds with equality. This case corresponds to lifted RS codes, for which this equivalence first appeared in [16], as restated in Lemma 1. Therefore, we will also refer to the $[m, 1, d, q]$ lifted multiplicity code as the $[m, d, q]$ lifted RS code in the following. In Appendix B, we provide some codewords of a lifted multiplicity code with $s \geq 2$, which are not included in the subcode spanned by the evaluation of good monomials, thereby showing that the statement of Proposition 3 does not hold with equality in general.

III. Analysis of lifted RS codes

In this section, we investigate the code dimension of lifted RS codes. For this purpose, we first introduce the concept of $(q-r)$-bad monomials (slightly different from $(q-r)^*$-bad monomials) and derive an explicit evaluation formula to count the number of such monomials when the parameter $r \leq m$ is fixed and the field size $q = 2^\ell$ is scaled. To emphasize that we scale q independently of r, we do not denote the maximum degree by d in the following, but instead explicitly write $q - r$. Second, we show how to use the evaluation formula to derive a bound on the number of $(q-r)^*$-bad monomials for arbitrary $r \leq q$. Our estimate improves upon the result presented in [16, Sections 3.2, 3.4] for $m \geq 3$ and is consistent with the result for $m = 3$ provided in [20].

A. Computing the number of $(q-r)$-bad monomials

Let us introduce a terminology useful for establishing the number of d^*-bad monomials. Let $r \leq \min(m, q)$ be a fixed positive integer.

Definition 7 ($(q-r)$-bad monomial)**.** We say that a monomial $\mathbf{X}^{\mathbf{d}}$ with $\mathbf{d} \in \mathbb{Z}_q^m$ is $(q-r)$-*bad* over $\mathbb{F}_q[\mathbf{X}]$ if there exists at least one $\mathbf{i} \in \mathbb{Z}_q^m$ such that $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) \pmod q = (q-r)$.

Remark. The difference with Definition 2 is, roughly speaking, in the modulo operation, namely $\pmod q$ is used in Definition 7, whereas $\pmod{q-1}$ is used in Definition 2.

Let $S_j(\ell)$ denote the set of tuples $\mathbf{d} \in \mathbb{Z}_q^m$, $q = 2^\ell$, for which there exists $\mathbf{i} \leq_2 \mathbf{d}$ with $\deg(\mathbf{i}) = (q-r) + jq = (2^\ell - r) + j2^\ell$ and $s_j(\ell)$ be the cardinality of $S_j(\ell)$. We note that $S_j(\ell)$ also depends on r, however, we omit this in our notion as we fix r and scale only $\ell = \log q$. Also, the evaluation formula we provide does not depend on r. Clearly, $s_j(\ell) = 0$ for $j \geq m$ as the maximal $\deg(\mathbf{i})$ over admissible $\mathbf{i}$ is $m(q-1)$ which is smaller than $(q-r) + mq$. Therefore, we aim to compute $\sum_{i=0}^{m-1} s_i(\ell)$ since the number of $(q-r)$-bad monomials over $\mathbb{F}_q$ is bounded by this value from one side and by $s_0(\ell)$ from the other side.

Example. For $q = 4$, $r = 1$ and $m = 2$ the set $S_0(2)$ is

$$\begin{array}{rcl} S_0(2) & = & \{\ (3,0),\ (2,1),\ (3,1),\ (1,2),\ (3,2),\ (0,3),\ (1,3),\ (2,3),\ (3,3)\ \} \\ & & \quad\downarrow \quad\downarrow \quad\downarrow \quad\downarrow \quad\downarrow \quad\downarrow \quad\downarrow \quad\downarrow \quad\downarrow \\ & \mathbf{i}\ : & \ (3,0)\ (2,1)\ (3,0)\ (1,2)\ (3,0)\ (0,3)\ (1,2)\ (2,1)\ (3,0) \end{array}.$$

It is easy to check that for any $\mathbf{d} \in S_0(2)$ and the corresponding $\mathbf{i}$ it holds that $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) = (q-r) + jq = (4-1) + 0 \cdot 4 = 3$. The cardinality of the set is $s_0(2) = |S_0(2)| = 9$. For these parameters the only $\mathbf{d}$ with $\deg(\mathbf{d}) \geq q - r = 3$ that is not $(q-r)$-bad is $\mathbf{d} = (2, 2)$.

Before presenting our main technical result, we establish two important preliminary results.

Lemma 3. *If $\mathbf{d} \in S_j(\ell)$ for a non-negative integer j, then $\mathbf{d} \in S_l(\ell)$ for any non-negative integer $l < j$.*

Proof. As $\mathbf{d} \in S_j(\ell)$, there exists some $\mathbf{i}$ such that $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) = (q-r) + jq = (2^\ell - r) + j2^\ell$. We shall prove that there exists $\mathbf{i}'$ such that $\mathbf{i}' \leq_2 \mathbf{i}$ and $\deg(\mathbf{i}') = (2^\ell - r) + l2^\ell$. This is sufficient for showing $\mathbf{d} \in S_l(\ell)$. To this end, we provide an iterative procedure that takes an arbitrary $\mathbf{i} \in \mathbb{Z}_q^m$ with $\deg(\mathbf{i}) \geq j2^\ell$ and outputs $\mathbf{a} \leq_2 \mathbf{i}$ with $\deg(\mathbf{a}) = \deg(\mathbf{i}) - (j-l)2^\ell$ for $l \in [j]$. The procedure goes from the leading bits to the least significant ones and replaces some ones in the binary representations of $\mathbf{i} = (i_1, \ldots, i_m)$ by zeros.

1) **Step 1.** Let us initialize $\mathbf{a} \leftarrow \mathbf{i}$ and $\Delta \leftarrow (j-l)$ and $h \leftarrow \ell$.
2) **Step 2.** If $h = 0$, output $\mathbf{a}$. Else, let $h \leftarrow h - 1$ and $\Delta \leftarrow 2\Delta$. Compute $\delta = \Delta - \sum_{\xi=1}^{m} a_\xi^{(h)}$. If $\delta > 0$, let $\Delta \leftarrow \Delta - \delta$ and $a_\xi^{(h)} \leftarrow 0$ for all $\xi \in [m]$. Repeat Step 2. Else, let m' satisfy $\Delta - \sum_{\xi=1}^{m'} a_\xi^{(h)} = 0$ and let $a_\xi^{(h)} \leftarrow 0$ for all $\xi \in [m']$. Output $\mathbf{a}$.

According to the procedure, we output the correct $\mathbf{a}$ if we do the else-part in Step 2 at some point. Assume to the contrary that this does not happen. This means that we output the all-zero tuple at the end. However, $\Delta = (j-l)2^\ell - \deg(\mathbf{i}) > 0$ at the final step which contradicts with $\deg(\mathbf{i}) \geq j2^\ell$. This completes the proof. ■

Example. Consider the parameters $q = 2^\ell = 4$, $m = 2$, $r = 2$, $j = 1$, and $l = 0$. For the element $\mathbf{d} = (3,3) \in S_1(2)$ and $\mathbf{i} = (3,3) = (11,11)_2$ with $\mathbf{i} \leq_2 \mathbf{d}$ we will find the corresponding $\mathbf{a}$ with $\mathbf{a} \leq_2 \mathbf{i}$ and $\deg(\mathbf{a}) = \deg(\mathbf{i}) - (j-l)2^\ell = 2$.

1) **Step 1.** Initialize $\mathbf{a} \leftarrow (3,3)$ and $\Delta \leftarrow j - l = 1$ and $h \leftarrow \ell = 2$.
2) **Step 2.** Let $h \leftarrow h - 1 = 1$ and $\Delta \leftarrow 2\Delta = 2$. Compute $\delta = \Delta - \sum_{\xi=1}^{m} a_\xi^{(h)} = 0$. Since $\delta \not> 0$ we choose $m' = 2$ to satisfy $\Delta - \sum_{\xi=1}^{m'} a_\xi^{(h)} = 0$ and set $a_1^{(1)} \leftarrow 0$, $a_2^{(1)} \leftarrow 0$ to obtain $\mathbf{a} = (01,01)_2 = (1,1)$.

As $\mathbf{a} \leq_2 \mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{a}) = q - r = 2$ it follows that $\mathbf{d} \in S_0(2)$.

Let us introduce some auxiliary functions. We define two maps $F_{\mathrm{drop}} : \mathbb{Z}_{2^\ell} \to \mathbb{Z}_{2^{\ell-1}}$ and $F_{\mathrm{lead}} : \mathbb{Z}_{2^\ell} \to \mathbb{Z}_2$ that take an integer $a = \sum_{i=0}^{\ell-1} a^{(i)} 2^i$ and output $a - 2^{\ell-1} a^{(\ell-1)}$ and $a^{(\ell-1)}$, respectively (we either drop the leading bit in the binary representation of a or output it). We extend the maps F_{drop} and F_{lead} to $\mathbb{Z}_{2^\ell}^m$ in a straightforward manner by applying functions to each component of a vector $\mathbf{a} \in \mathbb{Z}_{2^\ell}^m$, that is

$$F_{\mathrm{drop}}(\mathbf{a}) = (F_{\mathrm{drop}}(a_1), \ldots, F_{\mathrm{drop}}(a_m)),$$
$$F_{\mathrm{lead}}(\mathbf{a}) = (F_{\mathrm{lead}}(a_1), \ldots, F_{\mathrm{lead}}(a_m)).$$

For an integer a, we denote $\max(a, 0)$ by $(a)^+$.

Lemma 4. *If $\mathbf{d} \in S_j(\ell+1)$ for a non-negative integer j, then $F_{\mathrm{drop}}(\mathbf{d})$ belongs to $S_0(\ell), S_1(\ell), \ldots, S_{(2j+1-|F_{\mathrm{lead}}(\mathbf{d})|)^+}(\ell)$.*

Proof. By definition, if $\mathbf{d} \in S_j(\ell+1)$, then there exists some $\mathbf{i} \in \mathbb{Z}_{2^{\ell+1}}^m$ with $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) = (2^{\ell+1} - r) + j2^{\ell+1}$. If the leading bits in $\mathbf{i}$ are dropped, then the sum of components of $F_{\mathrm{drop}}(\mathbf{i})$ is

$$\begin{aligned}\deg(F_{\mathrm{drop}}(\mathbf{i})) &= \deg(\mathbf{i}) - |F_{\mathrm{lead}}(\mathbf{i})|2^\ell \\ &= (2^\ell - r) + (2j + 1 - |F_{\mathrm{lead}}(\mathbf{i})|)|2^\ell.\end{aligned}$$

Since we also have the property $F_{\mathrm{drop}}(\mathbf{i}) \leq_2 F_{\mathrm{drop}}(\mathbf{d})$, we obtain that $F_{\mathrm{drop}}(\mathbf{d})$ belongs to $S_{2j+1-|F_{\mathrm{lead}}(\mathbf{i})|}(\ell)$. Additionally, we note that $|F_{\mathrm{lead}}(\mathbf{i})| \leq \min(2j+1, |F_{\mathrm{lead}}(\mathbf{d})|)$ as $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) = (2^\ell - r) + j2^\ell$. From this and Lemma 3, we conclude that $r(\mathbf{d})$ belongs to $S_0(\ell)$, $S_1(\ell)$, …, $S_{(2j+1-|F_{\mathrm{lead}}(\mathbf{d})|)^+}(\ell)$. This completes the proof. ∎

With these results established, we are now ready to give the key technical statement required for the estimation of the rate of lifted RS codes. Recall that $\binom{b}{\geq a}$ denotes the number of ways to choose an (unordered) subset of at least a elements from a fixed set of b elements. For $a < 0$ or $a > b$, we assume that $\binom{b}{a} = 0$.

Proposition 4. *The system of recurrence relations*

$$\begin{pmatrix} s_0(\ell+1) \\ s_1(\ell+1) \\ \vdots \\ s_j(\ell+1) \\ \vdots \\ s_{m-1}(\ell+1) \end{pmatrix} = A_m \begin{pmatrix} s_0(\ell) \\ s_1(\ell) \\ \vdots \\ s_j(\ell) \\ \vdots \\ s_{m-1}(\ell) \end{pmatrix}$$

holds true, where the square $m \times m$ matrix A_m is given by

$$A_m := \begin{pmatrix} \binom{m}{\geq 1} & \binom{m}{0} & 0 & 0 & \cdots & 0 \\ \binom{m}{\geq 3} & \binom{m}{2} & \binom{m}{1} & \binom{m}{0} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \binom{m}{\geq 2j+1} & \binom{m}{2j} & \binom{m}{2j-1} & \binom{m}{2j-2} & \cdots & \binom{m}{2j-m+2} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \binom{m}{\geq 2m-1} & \binom{m}{2m-2} & \binom{m}{2m-3} & \binom{m}{2m-4} & \cdots & \binom{m}{m} \end{pmatrix}.$$

Proof. To begin, first note that we can uniquely encode $\mathbf{d} \in \mathbb{Z}_{2^{\ell+1}}^m$ by the pair $(F_{\mathrm{lead}}(\mathbf{d}), F_{\mathrm{drop}}(\mathbf{d}))$. Let us define the set $\mathrm{Pair}(j)$ of size $s_j(\ell+1)$ as

$$\mathrm{Pair}(j) = \{(F_{\mathrm{lead}}(\mathbf{d}), F_{\mathrm{drop}}(\mathbf{d})) : \quad \mathbf{d} \in S_j(\ell+1)\}.$$

For $w \in \{0, \ldots, m\}$, we define the set $T^{(w)}(j)$ as follows

$$T^{(w)}(j) = \{(\mathbf{v}, \mathbf{y}) : \ \mathbf{v} \in \mathbb{Z}_2^m, \mathbf{y} \in S_{(2j+1-w)^+}(\ell), |\mathbf{v}| = w\}.$$

Clearly, for different $w \in \{0, \ldots, m\}$, the sets $T^{(w)}(j)$ are pairwise disjoint, and the size of $T^{(w)}(j)$ is

$$|T^{(w)}(j)| = \binom{m}{w} s_{(2j+1-w)^+}(\ell),$$

where we used the notation $s_j(\ell) = |S_j(\ell)|$. In the remaining proof, we show that the disjoint union of $T^{(w)}(j)$ coincides with $\mathrm{Pair}(j)$, that is

$$\mathrm{Pair}(j) = \bigsqcup_{w\in\{0,\dots,m\}} T^{(w)}(j). \tag{2}$$

Note that $(2j+1-w)^+ = 0$ for $w \geq 2j+1$ and, thus, $|T^{(w)}(j)| = \binom{m}{w} s_0(\ell)$ for $w \geq 2j+1$. Combining this observation, equality (2) and the fact $s_i(\ell) = 0$ for $i \geq m$ would lead to the required relation

$$\begin{aligned} s_j(\ell+1) = &\binom{m}{\geq 2j+1} s_0(\ell) + \binom{m}{2j} s_1(\ell) \\ &+ \binom{m}{2j-1} s_2(\ell) + \cdots + \binom{m}{2j-m+3} s_{m-2}(\ell) \\ &+ \binom{m}{2j-m+2} s_{m-1}(\ell). \end{aligned}$$

First, we check one direction of equation (2) – namely, each element in $\mathrm{Pair}(j)$ is covered by the union. Let $(F_{\mathrm{lead}}(\mathbf{d}), F_{\mathrm{drop}}(\mathbf{d})) \in \mathrm{Pair}(j)$ for some $\mathbf{d} \in S_j(\ell+1)$. By denoting $w = |F_{\mathrm{lead}}(\mathbf{d})|$ and applying Lemma 4, we get that $F_{\mathrm{drop}}(\mathbf{d}) \in S_{(2j+1-w)^+}(\ell)$. Therefore, $(F_{\mathrm{lead}}(\mathbf{d}), F_{\mathrm{drop}}(\mathbf{d})) \in T^{(w)}(j)$.

Second, we show that each element in $T^{(w)}(j)$ is included in $\mathrm{Pair}(j)$. Let $(\mathbf{v}, \mathbf{y}) \in T^{(w)}(j)$. Construct $\mathbf{d} \in \mathbb{Z}_{2^{\ell+1}}^m$ to satisfy $F_{\mathrm{lead}}(\mathbf{d}) = \mathbf{v}$ and $F_{\mathrm{drop}}(\mathbf{d}) = \mathbf{y}$. By definition, we have that $|\mathbf{v}| = w$ and $\mathbf{y} \in S_{(2j+1-w)^+}(\ell)$. The latter means that there exists an $\mathbf{i}$ such that $\mathbf{i} \leq_2 \mathbf{y}$ and $\deg(\mathbf{i}) = (2^\ell - r) + (2j+1-w)^+ 2^\ell$. Construct $\mathbf{i}' \in \mathbb{Z}_{2^{\ell+1}}^m$ such that $F_{\mathrm{drop}}(\mathbf{i}') = \mathbf{i} \leq_2 \mathbf{y} = F_{\mathrm{drop}}(\mathbf{d})$ and $F_{\mathrm{lead}}(\mathbf{i}') \leq_2 \mathbf{v} = F_{\mathrm{lead}}(\mathbf{d})$ and $|F_{\mathrm{lead}}(\mathbf{i}')| = \min(2j+1, w)$. Thus, we obtain that $\mathbf{i}' \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}') = (2^{\ell+1} - r) + j2^{\ell+1}$. This completes the proof. ■

Definition 8 (Largest eigenvalue λ_m)**.** Let A_m be as in Proposition 4 and Λ be the set of its eigenvalues. We define λ_m to be the largest element from Λ.

It is well known that the eigenvalues of a matrix are upper and lower bounded by the largest and smallest sum of its rows or columns, respectively. It follows directly from the structure of A_m that $2^{m-1} \leq \lambda_m \leq 2^m$. For the readers convenience, we provide λ_m and $m - \log \lambda_m$ for $2 \leq m \leq 10$ in Table I.

Note that the order of $s_j(\ell)$ is the maximum value in the matrix A_m^ℓ, the ℓth power of A_m. The exponential growth rate of the matrix powers A_m^ℓ as $\ell \to \infty$ is controlled by λ_m^ℓ. Since all elements of A_m^{m-1} are positive (except the mth row which has all zeros but the last entry), the matrix A_m has only one eigenvalue of maximum modulus by Perron-Frobenius theorem for non-negative matrices (e.g., see [22, Theorem 8.5.2]). Finally, we obtain the following statement.

Corollary 1. *For an integer $r \leq m$, the number of $(q-r)$-bad monomials is $\Theta_m(\lambda_m^\ell) = \Theta_m(q^{\log \lambda_m})$ as $q \to \infty$.*

B. Computing the number of $(q-r)^*$-bad monomials

Now let $r \leq q$ (the restriction $r \leq m$ is no longer necessary, i.e., r could be very large). By Definition 2, a monomial $\mathbf{X}^{\mathbf{d}}$ is $(q-r)^*$-bad if there exists an $\mathbf{i} \in \mathbb{Z}_q^m$ such that $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) \pmod{q}^* \in \{q-r, q-r+1, \dots, q-1\}$. The latter condition is equivalent to

$$\deg(\mathbf{i}) = q - r_0 + (q-1)j = (q - r_0 - j) + qj$$

for some $r_0 \in [r]$ and $j \in \mathbb{Z}_m$. Let us drop the $\lceil \log(r+m) \rceil$ least significant bits in every component of $\mathbf{d}$ and $\mathbf{i}$ to obtain some $\mathbf{d}'$ and $\mathbf{i}'$ from $\mathbb{Z}_{q'}^m$ with $q' = 2^{\ell'}$ and $\ell' = \ell - \lceil \log(r+m) \rceil$. Then we have that $\mathbf{i}' \leq_2 \mathbf{d}'$ and

$$(q' - m) + jq' \leq \deg(\mathbf{i}') \leq \lfloor \deg(\mathbf{i}) / 2^{\ell - \ell'} \rfloor \leq (q'-1) + jq'.$$

Therefore, by Definition 7, we have that $\mathbf{X}^{\mathbf{d}'}$ is $(q'-r')$-bad over $\mathbb{F}_{q'}[\mathbf{X}]$ for some positive integer $r' \leq m$. By simple counting arguments and Corollary 1, the following statement is implied.

Lemma 5. *For an integer $r < q = 2^\ell$, the number of $(q-r)^*$-bad monomials is $\Theta_m(r^{m - \log \lambda_m} q^{\log \lambda_m})$ as $\ell \to \infty$.*

Proof. The number of $(q-r)^*$-bad monomials can be bounded by the number of $(q'-r')$-bad monomials with $r' \leq m$ multiplied by the number of ways to choose $m\lceil \log(r+m) \rceil$ bits. By Corollary 1, it can be estimated as

$$m 2^m (r+m)^m O_m\left(q'^{\log \lambda_m}\right) = O_m\left(r^{m - \log \lambda_m} q^{\log \lambda_m}\right),$$

where the factor m comes from the number of choices for the parameter $r' \in [m]$ and $2^m (r+m)^m \geq 2^{m\lceil \log(r+m) \rceil}$ is the number of ways to choose $m\lceil \log(r+m) \rceil$ bits.

Now let us elaborate on showing that the number of $(q-r)^*$-bad monomials is $\Omega_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right)$. Take all $(q'-1)$-bad monomials $\mathbf{X}^{\mathbf{d}'}$ over $\mathbb{F}_{q'}[\mathbf{X}]$ with the property that there exists $\mathbf{i}' \leq_2 \mathbf{d}'$ such that $\deg(\mathbf{i}') = q'-1$. By Proposition 4 and Corollary 1, the number of such monomials can be bounded as $\Omega_m(q'^{\log\lambda_m})$. Define

$$\ell_0 \coloneqq \lceil \log(m+r) \rceil - \lfloor \log r \rfloor.$$

Then we concatenate every component d'_j of $\mathbf{d}' = (d'_1, \ldots, d'_m)$ with the all-one string of length ℓ_0 and an arbitrary binary string of length $\lfloor \log r \rfloor$. The total number of obtained tuples $\mathbf{d} \in \mathbb{Z}_q^m$ is then

$$2^{m\lfloor \log r \rfloor}\Omega_m\left(q'^{\log\lambda_m}\right) = \Omega_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right).$$

For every resulting tuple $\mathbf{d}$, the monomial $\mathbf{X}^{\mathbf{d}}$ is also $(q-r)^*$-bad over $\mathbb{F}_q[\mathbf{X}]$. Indeed, we can construct an appropriate $\mathbf{i}$ based on $\mathbf{i}'$. To see this, we concatenate every component i'_j (except i'_1) with the all-zero string of length $\lceil \log(r+m) \rceil$, and i'_1 with the all-one string of length ℓ_0 and the all-zero string of length $\lfloor \log r \rfloor$.

Then we have $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i})$ can be easily bounded as $q - r \leq \deg(\mathbf{i}) \leq q-1$. This completes the proof. ■

Example. Consider the parameters $q' = 2^{\ell'} = 4$, $m = 2$, $r = 2$, and $q = 2^{\ell' + \lceil \log(r+m) \rceil} = 16$. As shown in the previous example, we have $\mathbf{d}' = (1,3) \in S_0(\ell')$ with $\mathbf{i}' \leq_2 \mathbf{d}'$ for $\mathbf{i}' = (1,2)$. The binary representations of $\mathbf{d}'$ and $\mathbf{i}'$ are given by

$$\mathbf{d}' = (01, 11)_2$$
$$\mathbf{i}' = (01, 10)_2$$

Concatenating the all-one string of length $\ell_0 = \lceil \log(m+r) \rceil - \lfloor \log r \rfloor = 1$ followed by arbitrary strings of length $\lfloor \log r \rfloor = 1$ to the components of $\mathbf{d}'$ gives the tuples

$$\mathbf{d}_1 = (0110, 1110)_2$$
$$\mathbf{d}_2 = (0110, 1111)_2$$
$$\mathbf{d}_3 = (0111, 1110)_2$$
$$\mathbf{d}_4 = (0111, 1111)_2 \ .$$

The $\mathbf{i}$ such that $\mathbf{i} \leq \mathbf{d}_j,\ j = 1,2,3,4$, can be found by concatenating every component i'_j except for i'_1 with $\lceil \log(r+m) \rceil = 2$ zeros and i_1 with $\ell_0 = 1$ one and $\lfloor \log r \rfloor = 1$ zero, to obtain

$$\mathbf{i} = (0110, 1000)_2 \ .$$

The degree of $\mathbf{i}$ is $\deg(\mathbf{i}) = 14 \geq q - r$.

C. Code rate and distance of lifted RS codes

Theorem 1. *For a power of two q, the rate R and the relative distance δ of the $[m, q-r, q]$ lifted RS code are*

$$R = 1 - \Theta_m\left((q/r)^{\log\lambda_m - m}\right), \quad \delta \geq \frac{r}{q} \quad \text{as } q \to \infty.$$

Proof of Theorem 1. To estimate the code rate of $[m, q-r, q]$ lifted RS codes, it suffices to compute the fraction of $(q-r)^*$-good monomials. By Lemma 1 and 5, the rate is

$$1 - \Theta_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right)q^{-m} = 1 - \Theta_m\left((q/r)^{\log\lambda_m - m}\right)$$

as $q \to \infty$. To estimate the relative distance of the code, we first note that the lifted RS code is linear. Suppose that $(f(\mathbf{a}))|_{\mathbf{a}\in\mathbb{F}_q^m}$ is a non-zero codeword. Let us say that $f(\mathbf{w}_0) \neq 0$. Then for any $\mathbf{v} \in \mathbb{F}_q^m \setminus \{\mathbf{0}\}$, the polynomial $f(\mathbf{w}_0 + \mathbf{v}T)$ is equivalent to a non-zero univariate polynomial of degree at most $q-r-1$. Thus, $f(\mathbf{w}_0 + \mathbf{v}t) \neq 0$ for at least $r+1$ different values $t \in \mathbb{F}_q$ and $f(\mathbf{a})$ is non-zero for at least $1 + rq^{m-1}$ values $\mathbf{a} \in \mathbb{F}_q^m$. This completes the proof. ■

IV. Analysis of lifted multiplicity codes

Using the results on the rate of lifted RS codes, we now move to estimating the rate and minimal distance of lifted multiplicity codes (recall that lifted RS codes are trivial lifted multiplicity codes with $s = 1$). In the following, we impose the constraint $s \geq m$ on the parameters, which helps with dropping the modulo operation in the definition of bad monomials. Then by applying the known results for lifted RS codes, we show how to find the asymptotics of the number of bad monomials when m is fixed and q is large. Our estimate continues the study of two-dimensional lifts initiated in [11] and is consistent with the result for the case of $m = 2$ presented there.

A. Computing the number of $(qs-r,s)^*$-bad monomials

In this section, we show that the number of $(qs-r,s)^*$-bad monomials can be well approximated by "$\binom{s+m}{m-1}$ times the number of $(q-r,1)^*$-bad monomials".

First, we recall the known estimate for the number of $(q-r,1)^*$-bad monomials when the number of variables is fixed and the alphabet size is large, as established in Section III, in the notation of lifted multiplicity codes.

Corollary 2. *For an integer $r<q=2^\ell$, the number of $(q-r,1)^*$-bad monomials is $\Theta_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right)$ as $\ell\to\infty$, with λ_m as in Definition 8. Moreover, the number of $\mathbf{d}\in\mathbb{Z}_q^m$ such that there exists an $\mathbf{i}\in\mathbb{Z}_q^m$ with $\mathbf{i}\le_2\mathbf{d}$ and*

1) $\deg(\mathbf{i}) \pmod q \in\{q-r,q-r+1,\dots,q-1\}$ is $\Theta_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right)$ as $\ell\to\infty$.

2) $\deg(\mathbf{i})\in\{q-r,q-r+1,\dots,q-1\}$ is also $\Theta_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right)$ as $\ell\to\infty$.

Let $s\ge m$ be a power of two and $1\le r<q$. First, we show that for such a choice of parameters, the modulo operation in Definition 6 can be dropped. By Proposition 2, for $f(X)\in\mathbb{F}_q[X]$ with

$$\deg(f)\le(s-1)q+m(q-1)=(m+s-1)q-m,$$

we have that $[X^i](f(X) \pmod{X^{qs}+X^s}) = [X^i]f(X)$ for all $i\in\{qs-r,qs-r+1,\dots,qs-1\}$ as

$$(m+s-1)q-m-qs+s=(m-1)q-m+s<qs-r.$$

Therefore, by Definition 6, a monomial $\mathbf{X}^{\mathbf{d}}$ with $\mathbf{d}\in\mathbb{Z}_{qs}^m$ and $\deg_q(\mathbf{d})\le s-1$ is $(qs-r,s)^*$-bad if there exists a vector $\mathbf{i}$ such that $\mathbf{i}\le_2\mathbf{d}$ and $\deg(\mathbf{i})\in\{qs-r,qs-r+1,\dots,qs-1\}$.

Let a monomial $\mathbf{X}^{\mathbf{d}}$ be $(qs-r,s)^*$-bad. Then every component of $\mathbf{d}$ can be represented as $d_j=\hat{d}_jq+d'_j$ with $d'_j\in\mathbb{Z}_q$ and $\hat{d}_j\in\mathbb{Z}_s$ for all $j\in[m]$. As deduced above, there exists an $\mathbf{i}\in\mathbb{Z}_{qs}^m$ such that $\mathbf{i}\le_2\mathbf{d}$ and $\deg(\mathbf{i})\in\{qs-r,qs-r+1,\dots,qs-1\}$. Therefore, after representing $i_j=\hat{i}_jq+i'_j$, we obtain that $\mathbf{i}'\le_2\mathbf{d}'$ and $\deg(\mathbf{i}') \pmod q\in\{q-r,q-r+1,\dots,q-1\}$. Let us also check that $s-m\le\deg(\hat{\mathbf{d}})\le s-1$. To show $\deg(\hat{\mathbf{d}})\ge s-m$, we just note that

$$\deg(\mathbf{i})\le\deg(\mathbf{d})=\deg(\hat{\mathbf{d}})q+\deg(\mathbf{d}')\le\deg(\hat{\mathbf{d}})q+(q-1)m.$$

Thus, if $\deg(\hat{\mathbf{d}})<s-m$, we have that $\deg(\mathbf{i})\le(s-1)q-m<qs-r$ which contradicts the property $\deg(\mathbf{i})\in\{qs-r,qs-r+1,\dots,qs-1\}$. Note that $\deg(\hat{\mathbf{d}})=\deg_q(\mathbf{d})$, therefore $\deg(\hat{\mathbf{d}})\le s-1$. Finally, we arrive at the following statement.

Lemma 6. *For an integer $m<r<q=2^\ell$ and a power of two $s\ge m$, the number of $(qs-r,s)^*$-bad monomials is*

$$\Theta_m\left(s^{m-1}r^{m-\log\lambda_m}q^{\log\lambda_m}\right)\quad\text{as }\ell\to\infty.$$

Proof. As noted above, for every $(qs-r,s)^*$-bad monomial $\mathbf{X}^{\mathbf{d}}$, $\mathbf{d}$ can be uniquely decomposed to the pair $(\hat{\mathbf{d}},\mathbf{d}')$, where $s-m\le\deg(\hat{\mathbf{d}})\le s-1$ and for $\mathbf{d}'\in\mathbb{Z}_q^m$, there exists an $\mathbf{i}'\le_2\mathbf{d}'$ with $\deg(\mathbf{i}') \pmod q\in\{q-r,q-r+1,\dots,q-1\}$. Thus, Corollary 2 yields that the number of $(qs-r,s)^*$-bad monomials for $\ell\to\infty$ can be bounded by

$$\left(\sum_{j=1}^{m}\binom{s-j+m-1}{m-1}\right)O_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right)=O_m\left(s^{m-1}r^{m-\log\lambda_m}q^{\log\lambda_m}\right).$$

It remains to show that this estimate is asymptotically tight. To see this, consider all possible $\mathbf{d}'\in\mathbb{Z}_q^m$ such that there exists $\mathbf{i}'\in\mathbb{Z}_q^m$ with $\mathbf{i}'\le_2\mathbf{d}'$ and $\deg(\mathbf{i}')=q-r'\in\{q-r,q-r+1,\dots,q-1\}$. By Corollary 2 the number of such $\mathbf{d}'$ can be estimated as

$$\Omega_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right).$$

Now we take a look on all possible $\hat{\mathbf{d}}\in\mathbb{Z}_s^m$ such that $\deg(\hat{\mathbf{d}})=s-1$. We can estimate the number of such $\hat{\mathbf{d}}$ by $\binom{s+m-2}{m-1}$. For any such $\hat{\mathbf{d}}$, we define $\mathbf{d}\in\mathbb{Z}_{qs}^m$ to be such that $d_j=\hat{d}_jq+d'_j$ and note that $\mathbf{X}^{\mathbf{d}}$ is $(qs-r,s)^*$-bad as for $\mathbf{i}$ with $i_j=\hat{d}_jq+i'_j$, we have $\mathbf{i}\le_2\mathbf{d}$ and

$$\deg(\mathbf{i})=q\deg(\hat{\mathbf{d}})+\deg(\mathbf{i}')=q(s-1)+q-r'=qs-r',$$

which belongs to $\{qs-r,qs-r+1,\dots,qs-1\}$. Therefore, the number of $(qs-r,s)^*$-bad monomials is

$$\binom{s+m-2}{m-1}\Omega_m\left(r^{m-\log\lambda_m}q^{\log\lambda_m}\right)=\Omega_m\left(s^{m-1}r^{m-\log\lambda_m}q^{\log\lambda_m}\right).$$

This completes the proof. ∎

B. Rate and distance of lifted multiplicity codes

Theorem 2 (Rate and distance of lifted multiplicity codes)**.** *For powers of two s, q and integers r and m with $m \leq s \leq q$ and $r \leq q$, the rate of the $[m, s, qs - r, q]$ lifted multiplicity code is*

$$1 - O_m\left(s^{-1}(q/r)^{\log \lambda_m - m}\right) \quad \text{as } q \to \infty.$$

The relative distance Δ of the $[m, s, qs - r, q]$ lifted multiplicity code is

$$\Delta \geq \Delta_{min} := \left\lceil \frac{r - s + 1}{s} \right\rceil \frac{q - s}{q^2}.$$

For $s = o(r)$, $\Delta_{min} = \frac{r}{qs}(1 + o(1))$.

Proof of Theorem 2. By Proposition 3, we can obtain the lower bound on the rate of the lifted multiplicity code by computing the fraction of $(qs - r, s)^*$-good monomials. Thus, by Lemma 6, the rate is

$$1 - \frac{O_m\left(s^{m-1} r^{m - \log \lambda_m} q^{\log \lambda_m}\right)}{\binom{s+m-1}{m} q^m} = 1 - O_m\left(s^{-1}(q/r)^{\log \lambda_m - m}\right).$$

Now we estimate the distance of the $[m, s, qs - r, q]$ lifted multiplicity code. Consider a codeword which is the evaluation of some non-zero polynomial f. Let $\mathbf{w}_0 \in \mathbb{F}_q^m$ be a coordinate such that $f^{(<s)}(\mathbf{w}_0)$ is not all-zero. In what follows, we prove the existence of a set S, $|S| \geq (q - s)q^{m-1}$, of lines containing this point such that for any $L \in S$ polynomial $f|_L$ doesn't vanish for at least $\lceil r/s \rceil$ points. More explicitly, assume that for some $\mathbf{i}_0 \in \mathbb{Z}_{\geq}^m$ with $\deg(\mathbf{i}_0) = i_0 < p$, $f^{(\mathbf{i}_0)}(\mathbf{w}_0) \neq 0$. Let a line L be parameterized by $\mathbf{w}_0 + T\mathbf{v}$ with $\mathbf{v} = (1, v_2, \ldots, v_m)$, $v_i \in \mathbb{F}_q$. Define $g_{\mathbf{v}}(T) := f|_L = f(\mathbf{w}_0 + T\mathbf{v})$. By the definition of Hasse derivatives, we have

$$g_{\mathbf{v}}(T) = \sum_{\mathbf{i} \in \mathbb{Z}_{\geq}^m} f^{(\mathbf{i})}(\mathbf{w}_0 + T\mathbf{v}) T^{\deg(\mathbf{i})} \mathbf{v}^{\mathbf{i}}$$

and, thus,

$$g_{\mathbf{v}}^{(i_0)}(0) = \sum_{\mathbf{i}:\ \deg(\mathbf{i}) = i_0} f^{(\mathbf{i})}(\mathbf{w}_0) \mathbf{v}^{\mathbf{i}}.$$

Since $f^{(\mathbf{i}_0)}(\mathbf{w}_0) \neq 0$, we can think about the right-hand side of the above equality as a non-zero polynomial in $v_2, \ldots, v_m$ of degree at most s. This yields that there exist at most sq^{m-2} different $\mathbf{v} = (1, v_2, \ldots, v_m) \in \mathbb{F}_q^m$ such that $g_{\mathbf{v}}^{(i_0)}(0) = 0$. Thus, for at least $(q - s)q^{m-2}$ different lines L containing the point $\mathbf{w}_0$, the univariate polynomial $f|_L \neq 0$. By the definition of $[m, s, qs - r, q]$ lifted multiplicity codes, for any line L, $f|_L$ agrees with some univariate polynomial of degree at most $qs - r - 1$ on its first $s - 1$ derivatives. By Lemma 2, if $g_{\mathbf{v}}(T) = f|_L \neq 0$, there exist at least $\lceil (r+1)/s \rceil$ points on which $f|_L$ doesn't vanish with high multiplicity, i.e., for at least $\lceil (r + 1)/s \rceil$ different $t \in \mathbb{F}_q$, $g_{\mathbf{v}}^{(j)}(t) \neq 0$ for some $j < s$. This implies that the number of non-zero positions of the codeword produced by f is at least

$$1 + \left\lceil \frac{r+1}{s} - 1 \right\rceil (q - s)q^{m-2}.$$

Since the lifted multiplicity code is $\mathbb{F}_q$-linear, the distance of the lifted multiplicity code can be bounded by the same value. This completes the proof. ■

V. PIR CODES

In this section, we show that lifted multiplicity codes have the best known trade-off between the number of information symbols and the required redundancy for private information retrieval (PIR) codes.

A. Preliminaries and prior work

The defining property of a k-PIR code is that for every message symbol, there exist k mutually disjoint sets of coded symbols from which the message symbol can be uniquely recovered. PIR codes were suggested in [10] to decrease storage overhead in PIR schemes preserving both privacy and communication complexity. Formally, this family of codes is defined as follows.

Definition 9 (PIR code, [10])**.** Let $F : \Sigma^n \to \Sigma^N$ be a map that encodes a string $x_1, \ldots, x_n$ to $c_1, \ldots, c_N$ and $\mathcal{C}$ be the image of F. The code $\mathcal{C}$ will be called a *k-PIR code* (or *$[N, n, k]_{|\Sigma|}^P$ code*) over the alphabet Σ if for every $i \in [n]$, there exist k mutually disjoint sets $R_1, \ldots, R_k \subset [N]$ (referred to as *recovering sets*) and functions $g_1, \ldots, g_k$ such that for all $\mathbf{c} \in \mathcal{C}$ and for all $j \in [k]$, $g_j(\mathbf{c}|_{R_j}) = x_i$, where $\mathbf{c}|_R$ is the projection of $\mathbf{c}$ onto coordinates indexed by R.

The definition of a *code with the disjoint repair group property* (DRGP) [11] is similar to Definition 9, except that we should recover all codeword symbols instead of only information symbols. For $\mathbb{F}_q$-linear codes, any systematically encoded code with the DGRP directly gives a PIR code. In what follows, we summarize the results for PIR codes since the best known bounds for DRGP codes hold for PIR codes as well.

The main figure of merit when studying PIR codes is the value of N, given n and k. Denote by $N_q^{\mathrm{P}}(n,k)$ the value of the smallest N such that there exists an $[N,n,k]_q^P$ code. For the binary case, we will remove q from these and subsequent notations. Since it is known that for sublinear k and fixed q, $\lim_{n\to\infty} N_q^{\mathrm{P}}(n,k)/n = 1$, [10], [16], we evaluate these codes by their redundancy and define $r_q^{\mathrm{P}}(n,k) := N_q^{\mathrm{P}}(n,k) - n$. In order to have a better understanding of the asymptotic behavior of the redundancy, the value of $r_q^{\mathrm{P}}(n,k)$ is usually studied for either constant $k = O(1)$ or polynomial $k = \Theta(n^{\varepsilon})$, $\varepsilon \geq 0$.

Constructions of PIR codes with fixed k were first suggested in [10], [23]. In particular, it can be seen that for $k = 2$, $r_q^{\mathrm{P}}(n,2) = 1$, and for any fixed $k \geq 3$, $r_q^{\mathrm{P}}(n,k) = \Theta(\sqrt{n})$ [10], [24], [25]. There are several constructions of PIR codes [11], [23], [26]–[28] and based on them, it is already possible to deduce some results on the asymptotic behavior of $r_q^{\mathrm{P}}(n,k)$. For example, the constructions of *one-step majority logic decodable codes* from [28] assure that $r^{\mathrm{P}}(n,n^{\varepsilon}) = O(n^{0.5+\varepsilon})$ for all $\varepsilon \geq 0$. In [27] the authors discussed partially lifted codes and their application to non-binary PIR codes. More results for PIR codes were achieved in [26] by using multiplicity codes and array codes. The construction [11] of PIR codes is based on bi-variate lifted multiplicity codes. Constructions of PIR codes based on tri-variate lifted RS codes were investigated in [20]. Finally, the paper [29] introduced the so-called wedge-lifted codes to construct PIR codes.

Lemma 7. *The redundancy of non-binary PIR codes satisfies:* *Based on*

1) $r_q^{\mathrm{P}}(n,k) = \Theta_k(\sqrt{n})$ *for linear PIR codes with fixed* $k \geq 3$, *[10], [24], [25].* *[Steiner systems]*
2) $r_q^{\mathrm{P}}(n,n^{\varepsilon}) = O(n^{\delta(\varepsilon)})$ *for* $0 \leq \varepsilon < 1$, *where* $\delta(\varepsilon) = 1 - \frac{1}{\lfloor 2/(1-\varepsilon)\rfloor} + \frac{\varepsilon}{\lfloor 2/(1-\varepsilon)\rfloor - 1}$, *[26].* *[mult. code]*
3) $r_q^{\mathrm{P}}(n,n^{0.25}) = O(n^{0.714})$, *[27].* *[partially lifted codes]*
4) $r_q^{\mathrm{P}}(n,n^{\varepsilon}) = O(n^{\frac{1}{2}+\varepsilon(\log 3-1)})$ *for* $0 \leq \varepsilon < \frac{1}{2}$, *[11].* *[lifted mult. codes with $m = 2$]*
5) $r_q^{\mathrm{P}}(n,n^{1-1/m}) = O(n^{1+\log\left(1-2^{-m\lceil \log m\rceil}\right)/(m\lceil \log m\rceil)})$ *for an integer* $m \geq 2$, *[16].* *[lifted RS codes]*
6) $r_q^{\mathrm{P}}(n,n^{2/3}) = O(n^{\log_8(5+\sqrt{5})})$, *[20].* *[lifted RS codes with $m = 3$]*

Remark. From our results (c.f. Theorem 4), it follows that given n and $k = n^{\varepsilon}$, with $0 < \varepsilon \leq 1 - \frac{1}{m}$, the redundancy of non-binary k-PIR codes based on m-variate lifted multiplicity codes is $O(n^{\delta_{LM}(\varepsilon,m)})$, where

$$\delta_{LM}(\varepsilon,m) := \frac{m-1}{m} + \frac{1+\log\lambda_m - m}{m-1}\,\varepsilon. \tag{3}$$

We remark that for $m = 2$, the same $\delta_{LM}(\varepsilon,m)$ was first derived in [11] and gives the best estimate on $r(n,n^{\varepsilon})$ with $0 < \varepsilon \leq \frac{1}{2}$. For further comparison, we provide the relevant results for the best known families of non-binary PIR codes in the same form. For $0 \leq \varepsilon \leq 1 - \frac{1}{m}$, the required redundancy of n^{ε}-PIR codes based on m-variate multiplicity codes (c.f. [26]) and m-variate lifted RS codes is $O(n^{\delta_M(\varepsilon,m)})$, and $O(n^{\delta_{LRS}(\varepsilon,m)})$, respectively, where $\delta_M(\varepsilon,m) := \frac{m-1}{m} + \frac{1}{m-1}\varepsilon$ and $\delta_{LRS}(\varepsilon,m) := \delta_{LM}(\frac{m-1}{m},m)$. Clearly, $\delta_{LM}(\varepsilon,m) < \delta_M(\varepsilon,m)$ for all $0 < \varepsilon \leq 1 - \frac{1}{m}$ as $\log\lambda_m - m < 0$ (c.f. (1)).

Let us illustrate the improvement compared to Lemma 7 and consider the case of $m = 3$ and $\frac{1}{2} < \varepsilon \leq \frac{2}{3}$. Then, we have $\delta_{LM}(\varepsilon,3) = \frac{2}{3} + 0.4276\varepsilon$, $\delta_M(\varepsilon,3) = \frac{2}{3} + \frac{1}{2}\varepsilon$ and $\delta_{LRS}(\varepsilon,3) = 0.9517$. The latter is given in item 6 of Lemma 7. The proposed bound $\delta_{LM}(\varepsilon,3)$ coincides with $\delta_{LRS}(\varepsilon,3)$ for $\varepsilon = \frac{2}{3}$ and outperforms all known bounds for $\frac{1}{2} < \varepsilon < \frac{2}{3}$.

In Figure 1, we compare our results for non-binary PIR codes based on the most suitable m-variate lifted multiplicity codes to the known results summarized in Lemma 7. Table II in Appendix D gives the ranges in which each bound is best among all known results. It can be verified that for any $\varepsilon \in (\frac{1}{2},1) \setminus \{\frac{2}{3}\}$, our bounds based on lifted multiplicity codes improve the state-of-art results.

Lemma 8. *The redundancy of binary PIR codes satisfies:* *Based on*

1) $r^{\mathrm{P}}(n,k) = \Theta_k(\sqrt{n})$ *for linear PIR codes with fixed* $k \geq 3$, *[10], [24], [25].* *[Steiner system]*
2) $r^{\mathrm{P}}(n,n^{1-1/m}) = O(n^{1+\log\left(1-2^{-m\lceil \log m\rceil}\right)/(m\lceil \log m\rceil)}\log n)$ *for an integer* $m \geq 2$, *[16].* *[lifted RS codes]*
3) $r^{\mathrm{P}}(n,n^{\varepsilon}) = O(n^{0.5+\varepsilon})$ *for* $0 \leq \varepsilon < 1/2$, *[26], [28].* *[array and one-step majority logic dec. codes]*
4) $r^{\mathrm{P}}(n,n^{\varepsilon}) = O(n^{\delta(\varepsilon)})$ *for* $0 \leq \varepsilon < 1$, *where* $\delta(\varepsilon) = \min_{m \geq \lceil 1/(1-\varepsilon)\rceil} \left\{1 - \frac{m(1-\varepsilon)-1}{2m(m-1)}\right\}$, *[26].* *[binary image of mult. codes]*
5) $r^{\mathrm{P}}(n,n^{\varepsilon}) = O(n^{\frac{3}{4}+\varepsilon(\log 3-\frac{3}{2})})$ *for* $0 \leq \varepsilon < \frac{1}{2}$, *[11].* *[binary image of lifted mult. codes with $m = 2$]*
6) $r^{\mathrm{P}}(n,n^{2/3}) = O(n^{\log_8(5+\sqrt{5})}\log n)$, *[20].* *[binary image of lifted RS codes with $m = 3$]*
7) $r^{\mathrm{P}}(n,n^{1/(2a)}) = O(n^{0.5+\log(2-2^{-a})/(2a)})$ *for integers* $a \geq 1$, *[29].* *[wedge-lifted codes]*

Remark. The codes constructed in [1], [16], [20], [27] are q-ary codes of length $N = q^m$. To obtain a binary PIR code each symbol can be converted to $\log q = \log N^{\frac{1}{m}} = \frac{1}{m}\log N = \Theta(\log n)$ symbols, hence the additional factor of $\log(n)$ in Lemma 8 compared to Lemma 7. Clearly, the image of every recovery set of a q-ary symbol is also a recovery set for bit of the image of this symbol, so the number of mutually disjoint recovering sets is at least as large as in for the non-binary code.

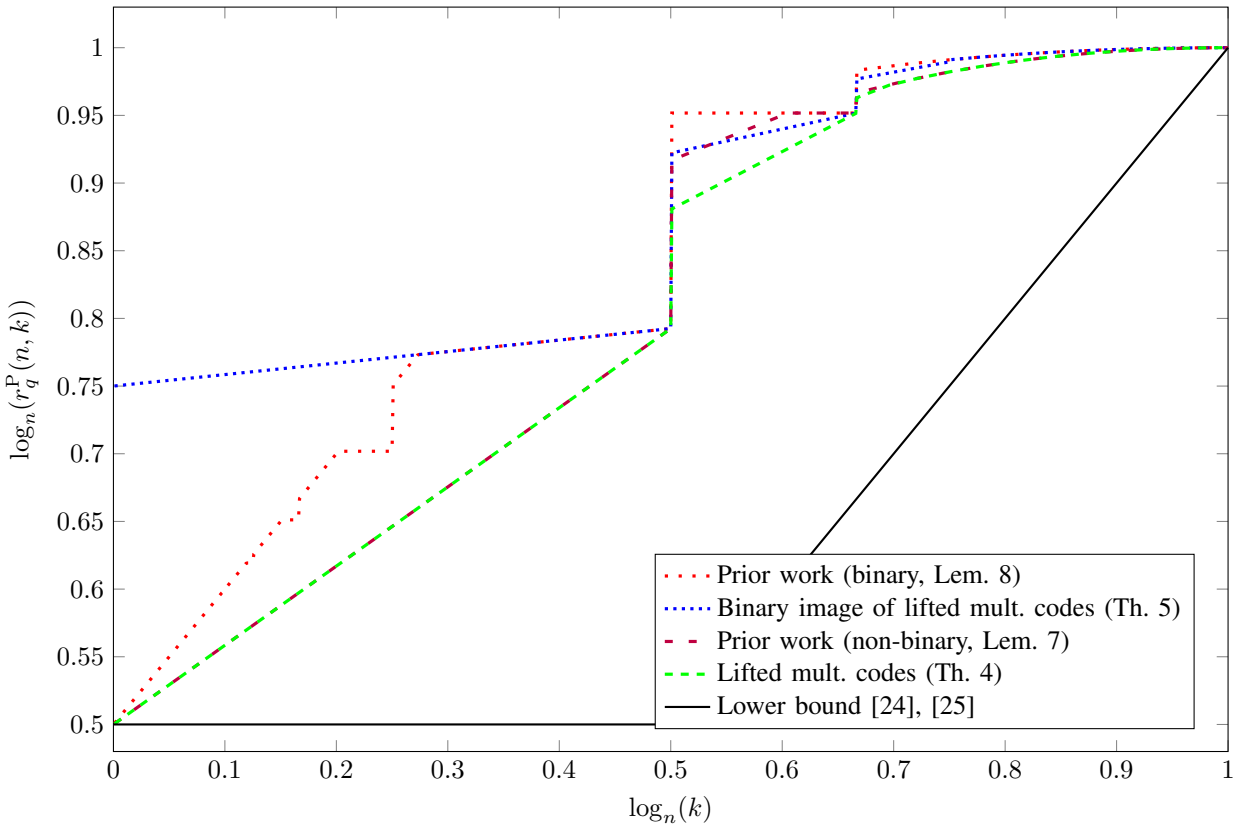

Fig. 1: Comparison of parameters of binary and non-binary PIR codes based on lifted multiplicity codes to the upper and lower bounds on the minimal redundancy of [11], [20], [24]–[27]. For $\log_n(k) \leq 0.5$ the results of Theorem 4 and Theorem 5 recover the results from [11].

We provide the relevant results for the best known families of binary PIR codes in the same form. For $0 \leq \varepsilon \leq (m-1)/m$, the required redundancy of binary n^{ε}-PIR codes based on m-variate lifted multiplicity codes (cf. Theorem 5), m-variate multiplicity codes, and m-variate lifted RS codes is $O(n^{\delta'_{LM}(\varepsilon)+o(1)})$, $O(n^{\delta'_{M}(\varepsilon)+o(1)})$, and $O(n^{\delta'_{LRS}(\varepsilon)+o(1)})$, respectively, where $\delta'_{LM}(\varepsilon,m) := \frac{2m-1}{2m} + \frac{1+2\log\lambda_m - 2m}{2m-2}\varepsilon$, $\delta'_{M}(\varepsilon,m) := \frac{2m-1}{2m} + \frac{1}{2m-2}\varepsilon$ and $\delta'_{LRS}(\varepsilon,m) := \delta'_{LM}(\frac{m-1}{m},m)$. Therefore, computing the bounds for small m and employing the inequality (1) for large m, we can range these three families of binary n^{ε}-PIR codes with $\varepsilon > 2/3$ as follows

$$\min_{m \geq \lceil \frac{1}{1-\varepsilon} \rceil} \delta'_{LM}(\varepsilon,m) < \min_{m \geq \lceil \frac{1}{1-\varepsilon} \rceil} \delta'_{M}(\varepsilon,m) < \min_{m \geq \lceil \frac{1}{1-\varepsilon} \rceil} \delta'_{LRS}(\varepsilon,m).$$

We remark that for $m = 2$ the same $\delta_{LM}(\varepsilon,m)$ was first derived in [11].

The binary image of lifted multiplicity codes requires the minimal redundancy among the best binary PIR codes, as given in Lemma 8, in the range $\varepsilon \in (0.273, 1)$. Our bounds provide a strict improvement for $\varepsilon \in (\frac{1}{2}, 1) \setminus \{2/3\}$. A more detailed comparison to the known constructions is given in Table III in Appendix D.

B. PIR and DRGP codes from lifted multiplicity codes

In this section, we apply our results on lifted multiplicity codes established in Section IV to PIR codes and codes with the disjoint repair group property. Our results improve the constructions of these codes based on ordinary multiplicity codes [26]. First, recall that a systematically encoded linear code with the DRGP property directly gives a PIR code, i.e., for linear codes the DRGP property is strictly stronger than that of PIR codes. The linear codes constructed from lifted multiplicity codes in the following have the DGRP property, but as the focus here are PIR codes, we state the results for this code class.

First, let us recall a known result for recovering the evaluation $f^{(<s)}(\mathbf{w}_0)$ for an arbitrary polynomial.

Lemma 9 (Follows from [26, Theorem 14])**.** *Let $f(\mathbf{X}) \in \mathbb{F}_q[\mathbf{X}]$ and a line L be parameterized as $\mathbf{w}_0 + T\mathbf{v}$. Define $g_{\mathbf{v}}(T) := f|_L = f(\mathbf{w}_0 + T\mathbf{v})$. Let a family of sets $Q_2, \ldots, Q_m$, $Q_i \subset \mathbb{F}_q$, $|Q_i| = s$, be given. If for all directions of the form $\mathbf{v} = (1, v_2, \ldots, v_m)$, $v_i \in Q_i$, and all $0 \leq j < s$, values $g_{\mathbf{v}}^{(j)}(0)$ are known, then it is possible to reconstruct $f^{(<s)}(\mathbf{w}_0)$.*

Next we prove that lifted multiplicity codes satisfy the definition of k-PIR codes for appropriate k.

Theorem 3 (Lifted multiplicity codes are PIR codes)**.** *Let q and s be powers of two and $m \geq 2$ be an integer such that $m \leq s \leq q$. The $[m, s, qs-s, q]$ lifted multiplicity code is a k-PIR code for $k = (q/s)^{m-1}$.*

Proof. For any line L parameterized by $\mathbf{w}_0 + T\mathbf{v}$ and a polynomial f producing a codeword of the $[m, s, qs-s, q]$ lifted multiplicity code, the polynomial $g_{\mathbf{v}}(T) := f|_L$ is equivalent up to order to s to a univariate polynomial $h(T)$ of degree at most $sq - s - 1$. By reading $g_{\mathbf{v}}^{(j)}(t)$ for all $0 \leq j < s$, $t \in \mathbb{F}_q \setminus \{0\}$, we can reconstruct polynomial $h(T)$ in $O(qs \log(qs))$ time (cf. [30]) and get the values $h^{(j)}(0) = g_{\mathbf{v}}^{(j)}(0)$ for all $0 \leq j < s$.

For an integer $i \in [q/s]$, let Q_i be a subset of $\mathbb{F}_q$ of size s so that $Q_i \cap Q_j = \emptyset$ for $j \neq i$. Let us index codeword symbols by elements of $\mathbb{F}_q^m$, i.e., $(c_1, \ldots, c_{q^m}) = (c_{\mathbf{a}})|_{\mathbf{a} \in \mathbb{F}_q^m}$, where $c_{\mathbf{a}} = f^{(<s)}(\mathbf{a})$. Fix an arbitrary vector $(i_2, \ldots, i_m) \in [q/s]^{m-1}$. By Lemma 9, for $\mathbf{w}_0 \in \mathbb{F}_q^m$, a possible recovering set for $c_{\mathbf{w}_0}$ is simply

$$\left\{\mathbf{w}_0 + \mathbf{v}t : t \in \mathbb{F}_q \setminus \{0\},\, v_1 = 1,\, v_j \in Q_{i_j} \text{ for } j \in [m] \setminus \{1\}\right\}.$$

Thus, for $c_{\mathbf{w}_0}$, we can construct at least $(q/s)^{m-1}$ mutually disjoint recovering sets. ■

Theorem 4 (Non-binary PIR codes)**.** *Given an integer $m \geq 2$, for any real ε with $0 < \varepsilon < \frac{m-1}{m}$ and a power of two q, there exists an n^{ε}-PIR code of length $N = q^m$ and dimension n over Σ such that the redundancy, $N - n$, and the alphabet size, $|\Sigma|$, satisfy*

$$N - n = O_m\left(n^{(m-1)/m + (\log \lambda_m - m + 1)\varepsilon/(m-1)}\right), \quad |\Sigma| = q^{\Theta_m(q^{m - \varepsilon m^2/(m-1)})}.$$

In other words, for $0 < \varepsilon < 1$, the polynomial growth of the minimal redundancy of n^{ε}-PIR codes with dimension n is

$$\log_n\left(r_{|\Sigma|}(n, n^{\varepsilon})\right) \leq \min_{m \geq \lceil 1/(1-\varepsilon) \rceil} \left(\frac{m-1}{m} + \frac{1 + \log \lambda_m - m}{m-1}\varepsilon\right).$$

Proof. Take $s = \Theta_m(q^{1 - \varepsilon m/(m-1)})$. For simplicity of notation, we assume that s is a power of two. By Theorem 3, there exists a k-PIR code with $k = (q/s)^{m-1} = \Theta_m(N^{\varepsilon}) = \Theta_m(n^{\varepsilon})$ over $\mathbb{F}_q^{\binom{s+m-1}{m}}$ of length $N = q^m$ and redundancy at most

$$\begin{aligned} N - n &= O_m\left(q^m s^{-1} (q/s)^{\log \lambda_m - m}\right) \\ &= O_m\left(q^{\varepsilon m/(m-1) + (m-1)} q^{\varepsilon m/(m-1)(\log \lambda_m - m)}\right) \\ &= O_m\left(n^{(m-1)/m + (\log \lambda_m - m + 1)\varepsilon/(m-1)}\right). \end{aligned}$$

■

We now transform the non-binary codes constructed in Theorem 4 into binary PIR codes.

Theorem 5 (Binary PIR codes)**.** *Given a positive integer m, for any real ε with $0 < \varepsilon < \frac{m-1}{m}$ and an integer n sufficiently large, there exists a binary n^{ε}-PIR code of length N and dimension n such that the redundancy, $N - n$, satisfies*

$$N - n = O_m\left(n^{(m-1/2)/m + \varepsilon(1/2 + \log \lambda_m - m)/(m-1)} \log n\right).$$

In other words, for $0 < \varepsilon < 1$, the polynomial growth of the minimal redundancy of binary n^{ε}-PIR codes with dimension n is

$$\log_n\left(r(n, n^{\varepsilon})\right) \leq \min_{m \geq \lceil 1/(1-\varepsilon) \rceil} \left(\frac{2m-1}{2m} + \frac{1 + 2\log \lambda_m - 2m}{2m-2}\varepsilon\right).$$

Proof. Let $\mathcal{C}$ be a non-binary PIR code as in Theorem 4. We construct the binary PIR code $\overline{\mathcal{C}}$ from $\mathcal{C}$ by converting each symbol of the alphabet of size $|\Sigma| = q^{\Theta_m(q^{m - \varepsilon m^2/(m-1)})}$ to

$$\log |\Sigma| = \Theta_m(q^{m - \varepsilon m^2/(m-1)} \log q) = \Theta_m(N^{1 - \varepsilon m/(m-1)} \log N) = \Theta_m(n^{1 - \varepsilon m/(m-1)} \log n)$$

bits. Denote the length and dimension of the binary code by $\overline{N}$ and $\overline{n}$, respectively. Thus, $\overline{n} = \Theta_m(n^{2 - \varepsilon m/(m-1)} \log n)$ and $\overline{N} = \Theta_m(n^{2 - \varepsilon m/(m-1)} \log n)$. Therefore, $n = \Theta_m(\overline{n}^{(m-1)/(2m-2-\varepsilon m)} / \log \overline{n})$. Denote by $\overline{r} = \overline{N} - \overline{n} = (N - n) \log |\Sigma|$ the redundancy and by $\overline{k}$ the availability parameter of the new code.

First, we note that the availability parameter of $\overline{\mathcal{C}}$ is at least that of $\mathcal{C}$. Indeed, we know that each bit in $\overline{\mathcal{C}}$ is a bit among $\log |\Sigma|$ bits representing some symbol in $\mathcal{C}$. For each recovering set of a symbol in $\mathcal{C}$, we get a corresponding recovering set for any bit from the image of this symbol in $\overline{\mathcal{C}}$. Therefore, $\overline{k} \geq k = n^{\varepsilon} \geq \Theta_m(\overline{n}^{\varepsilon(m-1)/(2m-2-\varepsilon m)} / (\log \overline{n})^{\varepsilon})$. Define $\overline{\varepsilon} = \varepsilon(m-1)/(2m-2-\varepsilon m)$. Then $\overline{k} = \Omega_m(\overline{N}^{\overline{\varepsilon}} / \log \overline{n})$ and $\varepsilon = (2m-2)\overline{\varepsilon}/(m - 1 + \overline{\varepsilon} m)$

Second, we rewrite the redundancy $\overline{r}$ in terms of $\overline{n}$ and $\overline{\varepsilon}$ as

$$\begin{aligned}\overline{r} = \overline{N} - \overline{n} &= O_m\left(n^{(m-1)/m+(\log\lambda_m - m+1)\varepsilon/(m-1)} n^{1-\varepsilon m/(m-1)}\log n\right)\\ &= O_m\left(n^{(2m-1)/m+(\log\lambda_m-2m+1)\varepsilon/(m-1)}\log n\right)\\ &= O_m\left(\overline{n}^{(m-1)(2m-1)/(2m^2-2m-2\varepsilon m^2)+(\log\lambda_m-2m+1)\varepsilon/(2m-2-\varepsilon m)}\log\overline{n}\right)\\ &= O_m\left(\overline{n}^{(m-1/2)/m+\overline{\varepsilon}(1/2+\log\lambda_m-m)/(m-1)}\log\overline{n}\right).\end{aligned}$$

■

VI. Batch codes

In this section we apply bounds on the rate of lifted RS codes from Section III to obtain a new construction of batch codes with improved redundancy. Additionally, using an idea from [26], we apply our results on PIR codes from Section V to obtain bounds on the redundancy of batch codes.

A. Preliminaries and prior work

By definition, PIR codes provide k non-intersecting recovery sets for *any single* information symbol. Batch codes generalize this property by requiring that *any k-tuple* of information symbols (with repetition) can be recovered from non-intersecting subsets of codeword symbols. Batch codes were originally motivated by different applications such as load-balancing in storage and cryptographic protocols [9]. In this work, we consider a special notion of batch codes, namely *primitive multiset batch codes* (for a more general study on the different notions of batch codes the reader is referred to [31]). Formally, this class of codes is defined as follows.

Definition 10 (Batch code, [9])**.** Let $F : \Sigma^n \to \Sigma^N$ be a map that encodes a string $x_1, \ldots, x_n$ to $c_1, \ldots, c_N$ and $\mathcal{C}$ be the image of F. The code $\mathcal{C}$ will be called a *k-batch code* (or $[N, n, k]^B_{|\Sigma|}$ *code*) over the alphabet Σ if for every multiset of symbols $\{x_{i_1}, \ldots, x_{i_k}\}$, $i_j \in [n]$, there exist k mutually disjoint sets $R_1, \ldots, R_k \subset [N]$ (referred to as *recovering sets*) and functions $g_1, \ldots, g_k$ such that for all $\mathbf{c} \in \mathcal{C}$ and for all $j \in [k]$, $g_j(\mathbf{c}|_{R_j}) = x_{i_j}$.

Several explicit and non-explicit constructions of these codes have been proposed, employing methods based on generalizations of Reed-Muller (RM) codes [9], [32], unbalanced expanders [9], graph theory [33], array and multiplicity codes [26], bi-variate lifted multiplicity codes and finite geometries [32]. For large $k = \Omega(n)$, batch codes are closely related to constant-query *locally correctable codes* and it is known [5], [34] that their rate approaches zero. On the other hand, when $k = O(1)$ is fixed, there exist explicit batch code constructions with the code rate very close to one [35].

Denote by $N_q^{\mathrm{B}}(n, k)$ the smallest N such that that there exists an $[N, n, k]_q^B$ code. Because of the above motivation, we classify batch codes by the required redundancy $r_q^{\mathrm{B}}(n, k) := N_q^{\mathrm{B}}(n, k) - n$. In this paper, we will be concerned with the regime of sublinear k, i.e., $k = n^\varepsilon$ with $n \to \infty$ and $0 \le \varepsilon \le 1$. For $q = 2$, we remove q in the subsequent notations. Several achievability results, i.e., upper bounds on the smallest achievable $r_q^{\mathrm{B}}(n, k)$, have been shown. We summarize the best presently known results that provide the smallest $r_q^{\mathrm{B}}(n, n^\varepsilon)$ for both binary and non-binary batch codes in the following statements.

Lemma 10. *The redundancy of non-binary batch codes satisfies* *Based on*

1) $r_q^{\mathrm{B}}(n, k) = O_k(\sqrt{n})$ *for linear batch codes with fixed* k, $3 \le k \le 5$, *[26], [35].* *[array construction]*
2) $r_q^{\mathrm{B}}(n, n^\varepsilon) = O(n^{3/4+\varepsilon/2})$ *for* $0 < \varepsilon < \frac{1}{2}$, *[26], [32].* *[lift. mult. codes and mult. codes with $m = 2$]*
3) $r_q^{\mathrm{B}}(n, n^\varepsilon) = O(n^{\delta(\varepsilon)})$ *for* $0 \le \varepsilon \le 1$, *where* $\delta(\varepsilon) = \min\limits_{m > \frac{2}{1-\varepsilon}} \left\{1 - \frac{1}{m} + \frac{1+\varepsilon}{2m-2}\right\}$, *[26].* *[mult. codes]*
4) $r_q^{\mathrm{B}}(n, n^\varepsilon) = O(n^{\frac{3\varepsilon+1}{2}}\log n)$ *for* $0 < \varepsilon < 1/3$, *[32].* *[finite geometry design]*

Non-binary batch codes obtained from lifted RS codes, as given in Theorem 7, require the minimal redundancy among all known non-binary n^ε-batch codes in the range $\varepsilon \in (0.432, 0.582]$ and those obtained from PIR codes, as given in Theorem 9, are best for $\varepsilon \in [0.582, 1)$. For a more detailed comparison, see Table IV in Appendix D.

Lemma 11. *The redundancy of binary batch codes satisfies* *Based on*

1) $r^{\mathrm{B}}(n, k) = O_k(\sqrt{n})$ *for linear batch codes with fixed* k, $3 \le k \le 5$, *[26], [35].* *[array construction]*
2) $r^{\mathrm{B}}(n, n^\varepsilon) = O(n^{\log_4(3)+(2-\log_2(3))\varepsilon}\log n)$ *for* $0 < \varepsilon < \frac{1}{2}$, *[32].* *[binary image of lifted mult. codes with $m = 2$]*
3) $r^{\mathrm{B}}(n, n^\varepsilon) = O(n^{\delta(\varepsilon)}\log n)$ *for* $0 \le \varepsilon \le 1$, *where* $\delta(\varepsilon) = \min\limits_{m > \frac{2}{1-\varepsilon}} \left\{1 - \frac{m(1-\varepsilon)-2}{4m(m-1)}\right\}$, *[26].* *[binary image of mult. codes]*
4) $r^{\mathrm{B}}(n, n^\varepsilon) = O(n^{\frac{3\varepsilon+1}{2}}\log n)$ *for* $0 < \varepsilon < 1/3$, *[32].* *[finite geometry design]*

Binary batch codes obtained from lifted RS codes, as given in Theorem 8, require the minimal redundancy among all known binary n^ε-batch codes in the range $\varepsilon \in (0.41, 0.648]$ and those obtained from PIR codes, as given in Theorem 10, are best for $\varepsilon \in [0.648, 1)$. A more detailed comparison is given in Table V in Appendix D.

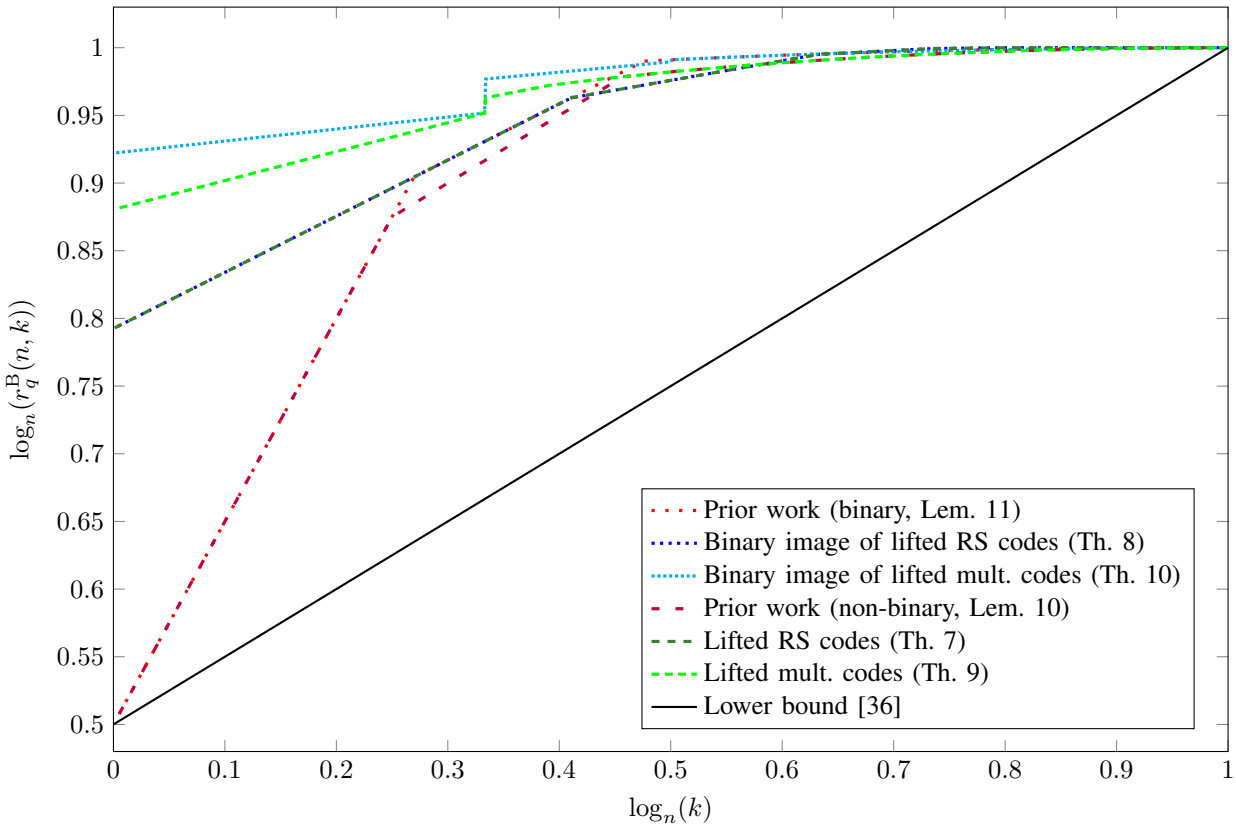

Fig. 2: Comparison of bounds on the parameters of batch codes based on m-variate lifted RS and lifted multiplicity codes for different values of m to the upper and lower bounds of [24]–[26], [32], [35].

On the other hand, the only non-trivial converse bound on the redundancy of systematic linear batch codes, yielding that $r_q^{\mathrm{B}}(n,k) = \Omega(\sqrt{nk})$, was recently shown in [36].

We illustrate the trade-off between parameters of batch codes in Figure 2.

B. Batch codes from lifted RS codes

In this section, a new construction of binary batch codes is presented. To this end, we first provide a construction of non-binary k-batch codes of length n based on the m-dimensional lift of an RS code. After that, we compute the parameters of this construction in the asymptotic regime for the availability parameter $k = n^{\varepsilon}$ with real $\varepsilon \in [\frac{m-2}{m}, \frac{m-1}{m}]$. Finally, we show how to convert this construction into a binary batch code.

We now show a one-way connection between lifted RS codes and batch codes.

Theorem 6. *Fix integers q, m and $r < q$. The $[m, q-r, q]$ lifted RS code has the following properties:*

1) *The length of the code is q^m.*
2) *The rate of the code is $1 - \Theta\left((q/r)^{\log \lambda_m - m}\right)$ as $q \to \infty$.*
3) *The code is a k-batch code for $k = q^{m-2}r$.*

Proof of Theorem 6. The first property follows from Definition 1. The second property is implied by Theorem 1.

To prove the third property, we first note that a lifted RS code is a linear code over $\mathbb{F}_q$ and it can be encoded systematically. Let $\mathbf{y}$ be a codeword of the $[m, d, q]$ lifted RS code. Since every coordinate of $\mathbf{y}$ is simply the evaluation $f(\mathbf{a})$ for some $\mathbf{a} \in \mathbb{F}_q^m$, we can index coordinates of our code by elements $\mathbf{a}$ from $\mathbb{F}_q^m$.

Now we shall prove a slightly stricter condition than required for k-batch codes, namely for every multiset of symbols $\{y_{\mathbf{a}_1}, \ldots, y_{\mathbf{a}_k}\}$, there exist mutually disjoint sets $R_1, \ldots, R_k \subset \mathbb{F}_q^m$ and some functions $g_1, \ldots, g_k$ such that $y_{\mathbf{a}_i} = g_i(\mathbf{y}|_{R_i})$. Let us prove the existence of $R_1, \ldots, R_k$ by using the inductive procedure described below.

To reconstruct $y_{\mathbf{a}_1}$, we take an arbitrary line L_1 in $\mathbb{F}_q^m$ containing $\mathbf{a}_1$ and let $R_1 = L_1 \setminus \{\mathbf{a}_1\}$. As the restriction of polynomial f to a line L_1 has degree less than $q-r$ by definition of lifted RS codes, we can interpolate $f|_{L_1}$ by reading evaluations of f at some $q-r$ points on the line L_1 and evaluate $f|_{L_1}$ at point $\mathbf{a}_1$. Suppose that for $k' < k$, symbols $\{y_{\mathbf{a}_1}, \ldots, y_{\mathbf{a}_{k'}}\}$ can be reconstructed by using recovering sets $R_1, \ldots, R_{k'}$, where R_i is a subset of a line L_i from the space $\mathbb{F}_q^m$. Since the number of lines passing through the point $\mathbf{a}_{k'+1}$ is larger than q^{m-1} and the total number of points already employed for recovering

$\{y_{\mathbf{a}_1}, \ldots, y_{\mathbf{a}_{k'}}\}$ is at most qk', we conclude that there exists a line $L_{k'+1}$ among q^{m-1} ones such that the cardinality of the intersection

$$\left| L_{k'+1} \bigcap \left\{ \bigcup_{i \in [k']} L_i \right\} \right| \leq \frac{qk'}{q^{m-1}} < \frac{qk}{q^{m-1}} = r.$$

Therefore, we can reconstruct $\mathbf{y}_{\mathbf{a}_{k'+1}}$ by reading evaluations of f at some $q - r$ unused points on $L_{k'+1}$, interpolating the univariate polynomial $f|_{L_{k'+1}}$ of degree less than $q - r$ and evaluating the latter at point $\mathbf{a}_{k'+1}$.

Thus, the required multiset of codeword symbols can be determined by this procedure. This completes the proof. ■

In the next statement we show a connection between parameters of the non-binary batch code constructed in Theorem 6.

Theorem 7. *Given a positive integer m, for any real ε with $\frac{m-2}{m} \leq \varepsilon < \frac{m-1}{m}$ and a power of two q, there exists a n^ε-batch code of length $N = q^m$ and dimension n over $\mathbb{F}_q$ such that the redundancy, $N - n$, satisfies*

$$N - n = O_m\left(n^{(m-\log \lambda_m)\varepsilon + ((m-1)\log \lambda_m/m - m + 2)}\right).$$

Proof of Theorem 7. Let $r = \lceil q^{m\varepsilon - m + 2} \rceil \geq n^{\varepsilon - (m-2)/m}$. By Theorem 6, there exists a k-batch code with $k = rq^{m-2} \geq q^{m\varepsilon} \geq n^\varepsilon$ over $\mathbb{F}_q$ of length $N = q^m$ and redundancy at most

$$\begin{aligned} N - n &= O_m\left(r^m \lambda_m^{\ell - \log r}\right) \\ &= O_m\left(2^{\ell m(m\varepsilon - m + 2)} \lambda_m^{\ell - \ell(m\varepsilon - m + 2)}\right) \\ &= O_m\left(n^{(m-\log \lambda_m)\varepsilon + ((m-1)\log \lambda_m/m - m + 2)}\right). \end{aligned}$$

■

Theorem 8. *Given a positive integer m, for any real ε with $\frac{m-2}{m} \leq \varepsilon \leq \frac{m-1}{m}$ and an integer n sufficiently large, there exists a binary n^ε-batch code of length N and dimension n such that the redundancy, $N - n$, satisfies*

$$N - n = O_m\left(n^{(m-\log \lambda_m)\varepsilon + ((m-1)\log \lambda_m/m - m + 2) + o(1)}\right).$$

Proof of Theorem 8. Let $\mathcal{C}$ be a non-binary batch code from Theorem 7. We construct the binary batch code $\mathcal{C}'$ from $\mathcal{C}$ by converting each symbol of the alphabet of size q to $\log q = \log N^{1/m} = \frac{1}{m} \log N = \Theta(\log n)$ bits. Denote the length, dimension of the binary code by N', n' respectively. Thus, $n' = \Theta(n \log n)$ and $N' = \Theta(N \log n)$. Therefore, $n = \Theta(n' / \log n')$. Denote by $r' = N' - n'$ the redundancy of the binary code and by k' be the availability parameter of the new code.

First, we note that the availability parameter of $\mathcal{C}'$ is at least that of $\mathcal{C}$. Indeed, we know that each bit in $\mathcal{C}'$ is a bit among $\log q$ bits representing some symbol in $\mathcal{C}$. For each recovering set of a symbol in $\mathcal{C}$, we have the corresponding recovering set for any bit from the image of this symbol in $\mathcal{C}'$. Therefore, $k' \geq k = n^\varepsilon \geq (n' / \log n')^\varepsilon$.

Second, we rewrite the redundancy r' in terms of n' as

$$\begin{aligned} r' = N' - n' &= O((N - n) \log n) \\ &= O_m\left(n'^{(m-\log \lambda_m)\varepsilon + ((m-1)\log \lambda_m/m - m + 2)} \log n'\right). \end{aligned}$$

As for any $\delta > 0$ and sufficiently large n we have $\log n < n^\delta$, the required statement is proved. ■

C. Batch codes from PIR codes

Batch codes from bi-variate lifted multiplicity codes were derived in [32], however, it is difficult to obtain such results from lifted multiplicity codes with a larger number of variables (for a similar discussion about batch codes from multiplicity codes, we refer the reader to [26]). However, by the generic connection between PIR and batch codes we are able to indirectly construct batch code from lifted multiplicity codes.

Recall the result from [26, Theorem 30] which relates the redundancy of batch and PIR codes

$$r_q^{\mathrm{B}}(n, n^\varepsilon) \leq r_q^{\mathrm{P}}(n, n^{\frac{1+\varepsilon}{2}}).$$

Combining this bound with Theorems 4 and 5 yields the following statements.

Theorem 9 (Non-binary batch codes)**.** *Given an integer $m \geq 3$, for any real ε with $0 < \varepsilon < \frac{m-2}{m}$ and a power of two q, there exists an n^ε-batch code of length $N = q^m$ and dimension n over Σ such that the redundancy, $N - n$, and the alphabet size, $|\Sigma|$, satisfy*

$$N - n = O_m\left(n^{(m-1)/m + (1+\log \lambda_m - m)(1+\varepsilon)/(2m-2)}\right), \quad |\Sigma| = q^{\Theta_m(q^{m - \varepsilon m^2/(m-1)})}.$$

In other words, for $0<\varepsilon<1$, the polynomial growth of the minimal redundancy of n^{ε}-batch codes with dimension n is

$$\log_n\left(r^{\mathrm{B}}_{|\Sigma|}(n,n^{\varepsilon})\right)\leq \min_{m\geq\lceil 2/(1-\varepsilon)\rceil}\left(\frac{m-1}{m}+\frac{(1+\log\lambda_m-m)(1+\varepsilon)}{2m-2}\right).$$

Theorem 10 (Binary batch codes)**.** *Given a positive integer m, for any real ε with $0<\varepsilon<\frac{m-2}{m}$ and an integer n sufficiently large, there exists a binary n^{ε}-batch code of length N and dimension n such that the redundancy, $N-n$, satisfies*

$$N-n=O_m\left(n^{(2m-1)/(2m)+\varepsilon(1+2\log\lambda_m-2m)(1+\varepsilon)/(4m-4)}\log n\right)$$

In other words, for $0<\varepsilon<1$, the polynomial growth of the minimal redundancy of binary n^{ε}-batch codes with dimension n is

$$\log_n\left(r^{\mathrm{B}}(n,n^{\varepsilon})\right)\leq \min_{m\geq\lceil 2/(1-\varepsilon)\rceil}\left(\frac{2m-1}{2m}+\frac{(1+2\log\lambda_m-2m)(1+\varepsilon)}{4m-4}\right).$$

VII. Locally correctable codes

In this section we show that a lifted multiplicity code is a locally correctable code (LCC) with certain parameters. Specifically, we provide a self-correction algorithm for lifted multiplicity codes.

A. Preliminaries and prior work

Unlike PIR codes, LCCs [5] explicitly require locality properties. Informally, a code is said to be locally correctable if given a vector that is sufficiently close to a codeword, each codeword coordinate can be recovered from a small subset of (possibly noisy) other positions with high probability. We give a formal definition of LCCs below.

Definition 11 (Locally correctable code.)**.** A code $\mathcal{C}$ of length N over an alphabet Σ is said to be (r,δ,ξ)-locally correctable if there exists a randomized correcting algorithm $\mathfrak{A}$ such that

1) For all $\mathbf{c}\in\mathcal{C}, i\in[N]$ and all vectors $\mathbf{y}\in\Sigma^N$ such that the relative distance $\Delta(\mathbf{y},\mathbf{c})\leq\delta$, we have $\Pr(\mathfrak{A}(\mathbf{y},i)=c_i)\geq 1-\xi$.
2) $\mathfrak{A}$ makes at most r queries to $\mathbf{y}$.

LDCs [6] are defined similar to LCCs, except that there the algorithm is required to recover message symbols instead of codeword symbols. Note, that for linear codes local correctability is a strictly stronger notion than local decodability, as a systematically encoded LCC is always an LDC.

LCCs have been constructed employing different approaches such as RM codes, lifted RS codes [16], multiplicity codes [18], and tensor codes [37], [38]. One typical question about LCCs is phrased as follows: given the high rate of a code (close to 1), how to get the query complexity as small as possible. The current state-of-the-art construction provided in [39] has the sub-polynomial (in length) query complexity. For an extensive discussion about other aspects of LCCs see [6], [40], [41] and the references therein.

B. LCCs from lifted multiplicity codes

One important ingredient to show the self-correction algorithm for lifted multiplicity codes is the following statement about hypergraphs. Recall that a *s-partite hypergraph* H is a pair $H=(V,E)$, where V is the vertex set that can be partitioned into sets $V_1,\dots,V_s$ so that each edge in the edge set E consists of a choice of precisely one vertex from each part. By $K_l^{(s)}$ denote a *complete s-partite hypergraph*, whose parts are all of equal size l.

Theorem 11 (Follows from [42, Theorem 1])**.** *Let $n>sl$, $l>1$. Then every s-partite hypergraph with n vertexes and at least $n^{s-1/l^{s-1}}$ hyperedges contains a copy of $K_l^{(s)}$.*

Theorem 12. *Let m be a fixed positive integer. For $s^{m-2}=o(\log q)$ and a real $\alpha<1/4$, the $[m,s,qs-r,q]$ lifted multiplicity code is a $((q-1)s^{m-1},\alpha\Delta_{min},2\alpha+o(1))$-locally correctable code, where $\Delta_{min}:=\left\lceil\frac{r-s+1}{s}\right\rceil\frac{q-s}{q^2}$.*

Remark. It is worth mentioning that the self-correction algorithm for multiplicity codes from [18], which has the query complexity $(q-1)5^m(s+1)^m$, also works well for lifted multiplicity codes. In our algorithm, we impose a stronger requirement on the order of derivatives: $s^{m-2}=o(\log q)$ for our algorithm and $s\leq q/5-1$ for the algorithm from [18]. However, our proposed algorithm has the query complexity $(q-1)s^{m-1}$, which implies a slightly better running time. For instance, the complexity of our algorithm is $\Theta_m(s)$ times smaller when m is fixed, $q\to\infty$ and $s=(\log q)^{1/(m-1)}$.

Proof. We prove this theorem by presenting a new self-correction algorithm $\mathfrak{A}$ for lifted multiplicity codes. Consider a vector $\mathbf{y}=(y_1,\dots,y_{q^m})=(y_{\mathbf{a}})|_{\mathbf{a}\in\mathbb{F}_q^m}$, which is a noisy version of the evaluation of the polynomial f. Say that we want to correct the value $f^{(<s)}(\mathbf{w}_0)$. The algorithm $\mathfrak{A}$ consists of three steps.

Step 1: Choose sets $Q_2,Q_3,\dots,Q_m$, $Q_i\subset\mathbb{F}_q$, $|Q_i|=s$, independently according to the uniform distribution over all subsets of size s. Form a set V of directions $\mathbf{v}=(1,v_2,\dots,v_m), v_i\in Q_i$.

Step 2: For every $\mathbf{v} \in V$ define a polynomial $g_{\mathbf{v}}(T) := f(\mathbf{w}_0 + T\mathbf{v})$. By the definition of lifted multiplicity codes we know that this polynomial agrees with some univariate polynomial of degree less than $qs - r$ on its first $s - 1$ derivatives. Apply the decoding algorithm for a univariate multiplicity code from [18], [43] to noisy evaluations of $g_{\mathbf{v}}(T)$ to obtain an estimation $\hat{g}_{\mathbf{v}}(T)$ of the correct polynomial $g_{\mathbf{v}}(T)$. Note that this decoding algorithm can correct up to $\lfloor (d_{min} - 1)/2 \rfloor$ errors, where $d_{min} := \lceil \frac{r+1}{s} \rceil$.

Step 3: Using Lemma 9 and polynomials $\hat{g}_{\mathbf{v}}(T)$, recover the value $f^{(<s)}(\mathbf{w}_0)$ to obtain $\hat{f}^{(<s)}(\mathbf{w}_0)$.

We now present an analysis of the algorithm. Call a direction $\mathbf{v}$ *good*, if the line $\mathbf{w}_0 + T\mathbf{v}$ contains at most $\lfloor (d_{min} - 1)/2 \rfloor$ errors. Note that if a direction $\mathbf{v}$ is good, then $\hat{g}_{\mathbf{v}}(T) \equiv_s g_{\mathbf{v}}(T)$. Thus, if all directions from V are good, the algorithm recovers the symbol correctly, i.e., $\hat{f}^{(<s)}(\mathbf{w}_0) = f^{(<s)}(\mathbf{w}_0)$. In the following we derive a bound on the probability that all directions from V are good.

Introduce an $(m-1)$-uniform $(m-1)$-partite hypergraph H, each part of which has size q. Index the elements within each part of the hypergraph with elements of $\mathbb{F}_q$. For every good direction $\mathbf{v} = (1, v_2, \ldots, v_m)$, draw a hyperedge $(v_2, \ldots, v_m)$ in H, where v_i is a vertex from the $(i-1)$th part. Then the probability of the successful recovery of $f^{(<s)}(\mathbf{w}_0)$ is lower bounded by the number of copies of $K_s^{(m-1)}$ in H divided by q^{m-1}.

The total number of good directions (or hyperedges in H) is at least

$$q^{m-1} - \frac{\alpha \Delta_{min} q^m}{\lfloor (d_{min} - 1)/2 \rfloor} = q^{m-1}(1 - 2\alpha + o(1)).$$

We show how we can find a large number of copies of $K_s^{(m-1)}$ in H. As long as the number of hyperedges in H is greater than $((m-1)q)^{m-1-1/s^{m-2}}$ we can find such a copy by Theorem 11. Then, we can spoil this copy by erasing one of its hyperedges and repeat the process for the obtained hypergraph. Obviously, all constructed copies of $K_s^{(m-1)}$ will be distinct. By this procedure, we can find at least

$$q^{m-1}(1 - 2\alpha + o(1)) - ((m-1)q)^{m-1-1/s^{m-2}} = q^{m-1}(1 - 2\alpha + o(1))$$

copies of $K_s^{(m-1)}$. Therefore, the probability of successful decoding is at least $1 - 2\alpha + o(1)$.

■

VIII. Conclusion

In this paper, we have investigated the rate, the distance, the availability and the self-correction properties of lifted Reed-Solomon codes and lifted multiplicity codes based on the evaluations of m-variate polynomials and discussed how to use them to construct batch codes, PIR codes, and LCCs. For some parameter regimes, such codes obtained from lifted RS and lifted multiplicity codes are shown to have a better rate/distance/availability/locality trade-off than other known constructions. In particular, our main results are:

1) We have improved the estimate on the rate of the m-dimensional lifts of RS codes when the field size is large. In particular, we have shown that for $r = O(1)$, the $[m, q - r, q]$ lifted RS code has rate $1 - \Theta(q^{\log \lambda_m - m})$ as $q \to \infty$.

2) We continue the study of lifted multiplicity codes initiated for the bi-variate case in [11] for any number of variables $m \geq 3$. Specifically, we show the rate of the $[m, s, sq - r, q]$ lifted multiplicity code to be $1 - O_m(s^{-1}(q/r)^{\log \lambda_m - m})$ and its relative distance to be $\Delta_{min} = \frac{r}{qs}(1 + o(1))$, by analyzing the code obtained from the span of good monomials. An interesting open problem is to extend this analysis to the code spanned by all good *polynomials*, i.e., the complete lifted multiplicity code.

3) We have proved that an $[m, s, sq - s, q]$ lifted multiplicity code is a k-PIR code of dimension $n = q^m(1 + o(1))$ with $k = (q/s)^{m-1}$. This improves the known upper bounds on the redundancy of PIR codes when k is sublinear in n and $k \geq \sqrt{n}$. For small enough s and any constant $\alpha < 1/4$, the $[m, s, sq - s, q]$ lifted multiplicity code is shown to be a $\big(qs^{m-1}, \alpha\Delta_{min}, 2\alpha\big)$-locally correctable code.

4) The locality property of lifted RS codes makes them attractive for constructing locally correctable codes and codes with the disjoint repair group property. We have shown that an $[m, q - r, q]$-lifted RS code is also a k-batch code with $k = rq^{m-2}$ and, by a generic transformation, we provide results on batch codes obtained from lifted multiplicity codes. This improves the known upper bounds on the redundancy of batch codes in some parameter regimes. On the other hand, there is no lower bound on the redundancy of batch and PIR codes other than that for $k \geq 3$ the redundancy of linear batch and PIR codes of length N is $\Omega(\sqrt{N})$ [24], [25]. Closing the (large) gap between the lower and upper bounds on the redundancy of both batch and PIR codes remains a major open problem.

REFERENCES

[1] L. Holzbaur, R. Polyanskaya, N. Polyanskii, and I. Vorobyev, "Lifted reed-solomon codes with application to batch codes," in *2020 IEEE Int. Symp. Inf. Theory (ISIT)*, 2020, pp. 634–639.

[2] L. Holzbaur, R. Polyanskaya, N. Polyanskii, I. Vorobyev, and E. Yaakobi, "On lifted multiplicity codes," in *2020 IEEE Information Theory Workshop (ITW)*. IEEE, 2021, pp. 1–5.

[3] C. Huang, M. Chen, and J. Li, "Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems," *ACM Trans. Storage*, vol. 9, no. 1, pp. 1–28, 2013.

[4] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, "On the locality of codeword symbols," *IEEE Trans. Inf. Theor.*, vol. 58, no. 11, p. 6925–6934, Nov. 2012.

[5] J. Katz and L. Trevisan, "On the efficiency of local decoding procedures for error-correcting codes," in *Proc. 32nd Annu. ACM Symp. Theory Comput. (STOC)*, 2000, pp. 80–86.

[6] S. Yekhanin *et al.*, "Locally decodable codes," *Found. Trends Theor. Comput. Sci.*, vol. 6, no. 3, pp. 139–255, 2012.

[7] T. Gur, G. Ramnarayan, and R. D. Rothblum, "Relaxed locally correctable codes," in *Proc. 9th Conf. Innov. Theor. Computer Sci. (ITCS)*, 2018, p. 27:1–27:11.

[8] E. Ben-Sasson, O. Goldreich, P. Harsha, M. Sudan, and S. Vadhan, "Robust PCPs of proximity, shorter PCPs, and applications to coding," *SIAM J. Comput.*, vol. 36, no. 4, pp. 889–974, 2006.

[9] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai, "Batch codes and their applications," in *Proc. 36th Annu. ACM Symp. Theory Comput. (STOC)*, 2004, pp. 262–271.

[10] A. Fazeli, A. Vardy, and E. Yaakobi, "PIR with low storage overhead: coding instead of replication," *arXiv preprint arXiv:1505.06241*, 2015.

[11] R. Li and M. Wootters, "Lifted multiplicity codes and the disjoint repair group property," in *Proc. Approx. Randomiz. Combinat. Optim. Algor. Techn. (APPROX/RANDOM)*, vol. 145, 2019, pp. 38:1–38:18.

[12] I. Reed, "A class of multiple-error-correcting codes and the decoding scheme," *Trans. IRE Prof. Group Inf. Theory*, vol. 4, no. 4, pp. 38–49, 1954.

[13] S. Arora and M. Sudan, "Improved low-degree testing and its applications," *Combinatorica*, vol. 23, no. 3, pp. 365–426, 2003.

[14] N. Alon, T. Kaufman, M. Krivelevich, S. Litsyn, and D. Ron, "Testing Reed-Muller codes," *IEEE Trans. Inf. Theory*, vol. 51, no. 11, pp. 4032–4039, 2005.

[15] R. Rubinfeld and M. Sudan, "Robust characterizations of polynomials with applications to program testing," *SIAM J. Comput.*, vol. 25, no. 2, pp. 252–271, 1996.

[16] A. Guo, S. Kopparty, and M. Sudan, "New affine-invariant codes from lifting," in *Proc. 4th Conf. Innov. Theor. Computer Sci. (ITCS)*, 2013, pp. 529–540.

[17] E. Ben-Sasson, G. Maatouk, A. Shpilka, and M. Sudan, "Symmetric LDPC codes are not necessarily locally testable," in *IEEE 26th Annu. Conf. Comput. Complex. (CCC)*, 2011, pp. 55–65.

[18] S. Kopparty, S. Saraf, and S. Yekhanin, "High-rate codes with sublinear-time decoding," *J. Assoc. Comput. Mach.*, vol. 61, no. 5, p. 28, 2014.

[19] L. Wu, "Revisiting the multiplicity codes: A new class of high-rate locally correctable codes," in *Proc. IEEE 53rd Annu. Allerton Conf. Commun. Contr. Comput. (Allerton)*, 2015, pp. 509–513.

[20] N. Polyanskii and I. Vorobyev, "Trivariate lifted codes with disjoint repair groups," in *Proc. IEEE XVI Int. Symp. Probl. Redund. Inf. Contr. Syst. (REDUNDANCY)*, 2019, pp. 64–68.

[21] Z. Dvir, S. Kopparty, S. Saraf, and M. Sudan, "Extensions to the method of multiplicities, with applications to kakeya sets and mergers," *SIAM J. Comput.*, vol. 42, no. 6, pp. 2305–2328, 2013.

[22] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[23] M. Vajha, V. Ramkumar, and P. Vijay Kumar, "Binary, shortened projective reed muller codes for coded private inf retrieval," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2017, pp. 2648–2652.

[24] S. Rao and A. Vardy, "Lower bound on the redundancy of PIR codes," *arXiv preprint arXiv:1605.01869*, 2016.

[25] M. Wootters, "Linear codes with disjoint repair groups," *unpublished mansucript, February*, 2016.

[26] H. Asi and E. Yaakobi, "Nearly optimal constructions of PIR and batch codes," *IEEE Trans. Inf. Theory*, vol. 65, no. 2, pp. 947–964, 2018.

[27] S. L. Frank-Fischer, V. Guruswami, and M. Wootters, "Locality via partially lifted codes," in *Proc. Approx. Randomiz. Combinat. Optim. Algor. Techn. (APPROX/RANDOM)*, vol. 81, 2017, pp. 43:1–43:17.

[28] S. Lin and D. J. Costello, *Error control coding: fundamentals and applications*. Upper Saddle River, NJ: Pearson/Prentice Hall, 2004.

[29] J. Hastings, A. Kanne, R. Li, and M. Wootters, "Wedge-lifted codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2021, pp. 2990–2995.

[30] F. Y. Chin, "A generalized asymptotic upper bound on fast polynomial evaluation and interpolation," *SIAM J. Comput.*, vol. 5, no. 4, pp. 682–690, 1976.

[31] V. Skachek, "Batch and PIR codes and their connections to locally repairable codes," in *Network Coding and Subspace Designs*. Springer, 2018, pp. 427–442.

[32] R. Polyanskaya, N. Polyanskii, and I. Vorobyev, "Binary batch codes with improved redundancy," *IEEE Transactions on Information Theory*, vol. 66, no. 12, pp. 7360–7370, 2020.

[33] A. S. Rawat, Z. Song, A. G. Dimakis, and A. Gál, "Batch codes through dense graphs without short cycles," *IEEE Trans. Inf. Theory*, vol. 62, no. 4, pp. 1592–1604, 2016.

[34] D. P. Woodruff, "A quadratic lower bound for three-query linear locally decodable codes over any field," *J. Computer Sci. Technol.*, vol. 27, no. 4, pp. 678–686, 2012.

[35] A. Vardy and E. Yaakobi, "Constructions of batch codes with near-optimal redundancy," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2016, pp. 1197–1201.

[36] R. Li and M. Wootters, "Improved batch code lower bounds," *arXiv preprint arXiv:2106.02163*, 2021.

[37] E. Ben-Sasson and M. Sudan, "Robust locally testable codes and products of codes," *Random Structures Algorithms*, vol. 28, no. 4, pp. 387–402, 2006.

[38] M. Viderman, "A combination of testability and decodability by tensor products," *Random Structures Algorithms*, vol. 46, no. 3, pp. 572–598, 2015.

[39] S. Kopparty, O. Meir, N. Ron-Zewi, and S. Saraf, "High-rate locally correctable and locally testable codes with sub-polynomial query complexity," *Journal of the ACM (JACM)*, vol. 64, no. 2, pp. 1–42, 2017.

[40] L. Trevisan, "Some applications of coding theory in computational complexity," in *Electron. Colloq. Comput. Complex. (ECCC)*, 2004.

[41] S. Kopparty and S. Saraf, "Local testing and decoding of high-rate error-correcting codes," in *Proc. Electron. Colloq. Comput. Complex. (ECCC)*, vol. 24, 2017, p. 126.

[42] P. Erdös, "On extremal problems of graphs and generalized graphs," *Israel J. Math.*, vol. 2, no. 3, pp. 183–190, 1964.

[43] M. Sudan, "Ideal error-correcting codes: Unifying algebraic and number-theoretic algorithms," in *Proc. Int. Symp. Applied Algebra Algebr. Algor. Error-Correcting Codes (AAECC)*. Springer, 2001, pp. 36–45.

[44] R. A. DeMillo and R. J. Lipton, "A probabilistic remark on algebraic program testing," *Inf. Process. Lett.*, vol. 7, no. 4, p. 193–195, 1977.

[45] R. Zippel, "Probabilistic algorithms for sparse polynomials," in *Proc. Int. Symp. Symb. Algebr. Manipul. (SYMSAC)*. Springer, 1979, pp. 216–226.

APPENDIX A
PROOF OF PROPOSITION 3

The proof is twofold, we need to show that

(Distinction) the evaluation of every monomial $\mathbf{X}^{\mathbf{d}}$ with $\deg_q(\mathbf{d}) \leq s-1$, which we refer to as a *type-s* monomial, gives a unique word

(Inclusion) these words are contained in the $[m, s, d, q]$ lifted multiplicity code as in Defintion 5 .

To show that the words are distinct, it is sufficient to prove that for an arbitrary non-trivial linear combination, written as $f(\mathbf{X})$, of type-s monomials, its evaluation is not equal to the all-zero codeword. Our proof is a straightforward generalization of [11, Lemma 14].

We prove the proposition by induction on m and s. More precisely, we deduce the statement for (m, s) from the cases for $(m-1, s)$ and $(m, s-1)$. The base case $m = 1$ is equivalent to [11, Lemma 11]. In the base case $s = 1$ the degree of each variable in f is at most $q-1$. Then the proposition follows from DeMillo–Lipton–Zippel Theorem [44], [45], which states that such polynomial can't have more than $q^m - (q-(q-1))^m = q^m - 1$ zeroes.

Now we prove the inductive step. Assume that $f(\mathbf{X})$ is a non-trivial linear combination of type-s monomials such that $f(\mathbf{X}) \equiv_s 0$. Consider the polynomial $g(X_1, \ldots, X_{m-1}) := f(X_1, \ldots, X_{m-1}, c)$ in $m-1$ variables, where $c \in \mathbb{F}_q$ is fixed. By the inductive hypothesis, we conclude that $g \equiv_s 0$. Hence, $(X_m - c)$ divides $f(\mathbf{X})$ for all $c \in \mathbb{F}_q$, so $(X_m^q - X_m)$ divides $f(\mathbf{X})$. Therefore, $f(\mathbf{X})$ can be represented as $f(\mathbf{X}) = (X_m^q - X_m)g(\mathbf{X})$.

It is easy to see that $g(\mathbf{X})$ is a linear span of type-$(s-1)$ monomials. Taking the $\mathbf{i}$th derivative of $f(\mathbf{X})$ for any $\mathbf{i} \in \mathbb{Z}_{\geq}^m$ with $i_m \geq 1$ we obtain

$$f^{(\mathbf{i})}(\mathbf{X}) = (X_m^q - X_m)g^{(\mathbf{i})}(\mathbf{X}) - g^{(\mathbf{j})}(\mathbf{X}),$$

where $\mathbf{j} = (i_1, \ldots, i_{m-1}, i_m - 1)$. The left-hand side is equal to zero for all $\mathbf{x} \in \mathbb{F}_q^m$ and $\mathbf{i} \in \mathbb{Z}_{\geq}^m$ with $\deg(\mathbf{i}) \leq s-1$. The right-hand side equals to $-g^{(\mathbf{j})}(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{F}_q^m$ and all $\mathbf{j} \in \mathbb{Z}_{\geq}^m$ with $\sum_{l=1}^{m-1} j_l < s-1$. By the induction hypothesis $g(\mathbf{X})$ is the zero polynomial, thus, $f(\mathbf{X})$ is the zero polynomial as well. This concludes the proof of the distinction property.

To show the inclusion, we prove that every $(d, s)^*$-good monomial $f(\mathbf{X}) = \mathbf{X}^{\mathbf{d}}$ over $\mathbb{F}_q$ satisfies the property that for any line $L \in \mathcal{L}_m$, the restriction $f|_L$ is equivalent up to order s to an univariate polynomial of degree less than d. Let a line L be parameterized as $(\mathbf{w} + \mathbf{v}T)|_{T \in \mathbb{F}_q}$ and $\mathbf{0}$ be the all-zero vector. Then, we have that

$$\begin{aligned} f|_L &= (\mathbf{w} + \mathbf{v}T)^{\mathbf{d}} \\ &= \sum_{\mathbf{0} \leq \mathbf{i} \leq \mathbf{d}} \prod_{j=1}^{m} v_j^{i_j} w_j^{d_j - i_j} \binom{d_j}{i_j} T^{i_j} \\ &\equiv_s \sum_{k=0}^{qs-1} c_k T^k := f^*(T), \end{aligned}$$

where c_k denotes the coefficients of the unique polynomial of degree $\leq qs-1$ that is equivalent to $f|_L$ (cf. Proposition 2). Recall that s and q are powers of 2. Hence, we have $f|_L(T) = f^*(T) \pmod{T^{qs} + T^s}$ by Proposition 2, so the coefficients $[T^s]f|_L$ that contribute to the coefficient c_k are exactly those for which $s = \deg(\mathbf{i}) \pmod{q}_s^* = k$, and we obtain

$$c_k := \sum_{\substack{\mathbf{0} \leq \mathbf{i} \leq \mathbf{d} \\ \deg(\mathbf{i}) \pmod{q}_s^* = k}} \prod_{j=1}^{m} v_j^{i_j} w_j^{d_j - i_j} \binom{d_j}{i_j}. \tag{4}$$

By Definition 6, for $k \geq d$, there is no $\mathbf{i} \in \mathbb{Z}_{qs}^m$ such that $\mathbf{i} \leq_2 \mathbf{d}$ and $\deg(\mathbf{i}) \pmod{q}_s^* = k$. Thus, for $k \geq d$ and every $\mathbf{i}$ used in the summation of (4), there exists some coordinate $j \in [m]$ such that $i_j \not\leq_2 d_j$. By Lucas's Theorem (e.g., see [11], [16]), for integers $d_j = (d_j^{(\ell-1)}, ..., d_j^{(0)})_2$ and $i_j = (i_j^{(\ell-1)}, ..., i_j^{(0)})_2$ it holds that

$$\binom{d_j}{i_j} = \prod_{\xi=0}^{\ell-1} \binom{d_j^{(\xi)}}{i_j^{(\xi)}} \mod 2.$$

It follows that if $i_j \not\leq_2 d_j$ the coefficient $\binom{d_j}{i_j} = 0$ in $\mathbb{F}_q$ (as q is a power of two) and therefore $c_k = 0$ for all $k \geq d$.

We have proved that the restriction of $\mathbf{X}^{\mathbf{d}}$ to any line is an univariate polynomial of degree at most $d-1$. Therefore, the $[m, s, d, q]$ lifted multiplicity code includes the codewords

$$\{(\mathbf{a}^{\mathbf{d}})|_{\mathbf{a} \in \mathbb{F}_q^m} \ : \ \mathbf{X}^{\mathbf{d}} \in \mathcal{F}_q(m, s, d)\} \ .$$

The inclusion of their linear combinations over $\mathbb{F}_q$ follows trivially from the proof. ■

APPENDIX B
LIFTED MULTIPLICITY CODE AND LIFTED MULTIPLICITY MONOMIAL CODE

We now give an example showing that lifted multiplicity codes are not necessarily spanned by the set of good monomials. Let $d = qs - 2$, $s = 2$, and $q > 2$. Denote by $M_1(\mathbf{X})$ and $M_2(\mathbf{X})$ the monomials

$$M_1(\mathbf{X}) := \mathbf{X}^{\mathbf{d}^{(1)}} = X_1^{qs-2} X_2$$
$$M_2(\mathbf{X}) := \mathbf{X}^{\mathbf{d}^{(2)}} = X_1^{(s-1)q-1} X_2^q \ ,$$

so $d_1^{(i)} = qs - 2$, $d_2^{(1)} = 1$, $d_1^{(2)} = (s-1)q - 1$, and $d_2^{(2)} = q$. Both monomials are type-s as

$$\deg_q(\mathbf{d}^{(1)}) = \deg_q(\mathbf{d}^{(2)}) = qs - 1 < 2(q-1) + (s-1)q$$

Further, both are $(d,s)^*$-bad, as the vectors $\mathbf{i}^{(1)} = \mathbf{d}^{(1)}$ and $\mathbf{i}^{(2)} = \mathbf{d}^{(2)}$ fulfill Definition 6 for each monomial, respectively. Also, their evaluation is not contained in an $[m, s, d, q]$ lifted multiplicity code, since for the line $(0, w_2) + (1, v_2)T \in \mathcal{L}_2$ we have

$$[T^{qs-1}] M_1(T, w_2 + v_2 T) = v_2$$
$$[T^{qs-1}] M_2(T, w_2 + v_2 T) = v_2^q \ .$$

However, the evaluation of their sum, i.e., the polynomial

$$P(\mathbf{X}) := M_1(\mathbf{X}) + M_2(\mathbf{X}) \ ,$$

is contained in the $[m, s, d, q]$ lifted multiplicity code as

$$\begin{aligned}[T^{qs-1}] P(w_1 + v_1 T, w_2 + v_2 T) &= [T^{qs-1}] M_1(w_1 + v_1 T, w_2 + v_2 T) + [T^{qs-1}] M_2(w_1 + v_1 T, w_2 + v_2 T) \\ &= v_1^{qs-2} v_2 + \underbrace{v_1^{(s-1)q-1} v_2^q}_{\overset{(\mathsf{a})}{=} v_1^{qs-2} v_2} = 0 \ ,\end{aligned}$$

where (a) holds because $v_1, v_2 \in \mathbb{F}_q$.

APPENDIX C
EXAMPLES OF RATE IMPROVEMENTS THROUGH LIFTING

To provide some intuition and show how lifting can improve the rate of lifted RS codes and lifted multiplicity codes, we provide examples for fixed sets of parameters.

A. RM codes vs. lifted RS codes

Let $f(X_1, X_2) = X_1^2 X_2^2$. Then the $[2, 3, 4]$ lifted RS code includes the codeword $\mathbf{c} = (f(a_1, a_2))|_{(a_1,a_2)\in\mathbb{F}_4^2}$ as for every line L, the degree of $f|_L$ is at most $2 < 3 = d$. Indeed, given a line L parameterized as $(w_1 + v_1 T, w_2 + v_2 T)|_{T\in\mathbb{F}_4}$ in $\mathbb{F}_4^2$, we have

$$\begin{aligned}f|_L = f(v_1 T + w_1, v_2 T + w_2) &= (v_1 T + w_1)^2 (v_2 T + w_2)^2 \\ &\overset{(i)}{=} (v_1^2 T^2 + w_1^2)(v_2^2 T^2 + w_2^2) \\ &\overset{(ii)}{=} (v_1^2 w_2^2 + v_2^2 w_1^2) T^2 + v_1^2 v_2^2 T + w_1^2 w_2^2,\end{aligned}$$

where in (i) we used the property $2v = 0$ for any $v \in \mathbb{F}_4$, and (ii) is implied by the fact that $T^4 = T$ in $\mathbb{F}_4[T]$. On the other hand, the 2-variate RM code of order 3 doesn't contain $\mathbf{c}$ as the degree of f is 4, which is larger than 3.

B. Multiplicity codes vs. lifted multiplicity codes

Let $m = s = 2$, $q = 4$, and $d = qs - 1 = 7$. Consider the monomial $M(\mathbf{X}) := X_1^2 X_2^6$. The degree of this monomial is $\deg(M(\mathbf{X})) = 8 > d$, so its evaluation is not contained in the $[2, 2, 7, 4]$ multiplicity code, as it only contains evaluations of degree $< d$ polynomials.

By Definition 5, the evaluation of $M(\mathbf{X})$ is contained in the $[2, 2, 7, 4]$ lifted multiplicity code if for every line $L \in \mathcal{L}_m$ there exists a polynomial $g(T) \in \mathcal{F}_q(d)$ such that the restriction of $M(\mathbf{X})$ to L is equivalent to $g(T)$. First, note that $M(\mathbf{X})$ is a type-s monomial, as $\deg_q(M(\mathbf{X})) = 1 \le s - 1$. Its evaluation in an arbitrary line $L \in \mathcal{L}_2$ is given by

$$\begin{aligned}M(\mathbf{X})|_L &= (w_1 + v_1 T)^2 (w_2 + v_2 T)^6 \\ &= (w_1^2 + v_1^2 T^2)(w_2^6 + w_2^4 v_2^2 T^2 + w_2^2 v_2^4 T^4 + v_2^6 T^6) \\ &= w_1^2 w_2^6 + (w_1^2 w_2^4 v_2^2 + v_1^2 w_2^6) T^2 + (w_1^2 w_2^2 v_2^4 + v_1^2 w_2^4 v_2^2) T^4 + (w_1^2 v_2^6 + v_1^2 w_2^2 v_2^4) T^6 + v_1^2 v_2^6 T^8 \ .\end{aligned}$$

By Proposition 2 and because s and q are powers of 2, we know that there exists an equivalent polynomial $M^*(T)$ of degree at most $qs-1=7$ such that $M(\mathbf{X})|_L \equiv_s M^*(T) \pmod{T^8+T^2}$. Here, we obtain this polynomial by substracting $v_1^2v_2^6(T^8+T^2)$ from $M(\mathbf{X})|_L$, which gives

$$M^*(T) = w_1^2w_2^6 + (w_1^2w_2^4v_2^2 + v_1^2w_2^6 + v_1^2v_2^6)T^2 + (w_1^2w_2^2v_2^4 + v_1^2w_2^4v_2^2)T^4 + (w_1^2v_2^6 + v_1^2w_2^2v_2^4)T^6 \ .$$

As the degree of this polynomial is $\deg(M^*(T)) < d = 7$ its evaluation is contained in the $[2, 2, 7, 4]$ lifted multiplicity code, thereby increasing its dimension compared to the $[2, 2, 7, 4]$ multiplicity code.

APPENDIX D
COMPARISON OF NEW BOUNDS TO KNOWN RESULTS

In Tables II we summarize the ranges of ε in which each bound on the required redundancy of n^ε-PIR and n^ε-batch codes of dimension n is best among the known results.

TABLE II: Non-binary PIR codes.

Given in	Reference	Based on	Best for ε in
Lemma 7, Item 2)	[26]	multiplicity codes	–
Lemma 7, Item 3)	[27]	partially lifted codes	–
Lemma 7, Item 4)	[11]	lifted mult. codes with $m=2$	$(0, \frac{1}{2}]$
Lemma 7, Item 5)	[16]	lifted RS codes	–
Lemma 7, Item 6)	[20]	lifted RS codes with $m=3$	$\{2/3\}$
Theorem 4	This work	lifted mult. codes	$(0, 1)$

TABLE III: Binary PIR codes.

Given in	Reference	Based on	Best for ε in
Lemma 8, Item 2)	[16]	lifted RS codes	$\{0.5\}$
Lemma 8, Item 3)	[26], [28]	array and one-step majority logic dec. codes	$[0, 0.273] \setminus \bigcup_{a\in\mathbb{N}}[\frac{\log(2-2^{-a})}{2a}, \frac{1}{2a}]$
Lemma 8, Item 4)	[26]	binary image of mult. codes	–
Lemma 8, Item 5)	[11]	binary image of lifted mult. codes with $m=2$	$[0.273, 0.5)$
Lemma 8, Item 6)	[20]	binary image of lifted RS codes with $m=3$	$\{2/3\}$
Lemma 8, Item 7)	[29]	wedge-lifted codes	$[0, 0.273] \bigcap \left\{\bigcup_{a\in\mathbb{N}}[\frac{\log(2-2^{-a})}{2a}, \frac{1}{2a}]\right\}$
Theorem 5	This work	binary image of lifted mult. codes	$[0.273, 1)$

TABLE IV: Non-binary batch codes.

Given in	Reference	Based on	Best for ε in
Lemma 10, Item 2)	[26], [32]	lift. mult. codes and mult. codes with $m=2$	$[0.25, 0.432]$
Lemma 10, Item 3)	[26]	PIR codes (mult. codes with $m \geq 3$)	–
Lemma 10, Item 4)	[32]	finite geometry design	$(0, 0.25]$
Theorem 7	This work	lifted RS codes	$[0.432, 0.582]$
Theorem 9	This work	PIR codes (lifted mult. codes)	$[0.582, 1)$

TABLE V: Binary batch codes.

Given in	Reference	Based on	Best for ε in
Lemma 11, Item 2)	[32]	binary image of lifted mult. codes with $m=2$	$[0.269, 0.41]$
Lemma 11, Item 3)	[26]	PIR codes (binary image of mult. codes with $m \geq 3$)	–
Lemma 11, Item 4)	[32]	finite geometry design	$(0, 0.269]$
Theorem 8	This work	binary image of lifted RS codes	$[0.41, 0.648]$
Theorem 10	This work	PIR codes (binary image of lifted mult. codes)	$[0.648, 1)$