Lifted Multiplicity Codes and the Disjoint Repair Group Property

Ray Li, *Graduate Student Member, IEEE*, and Mary Wootters, *Member, IEEE*

***Abstract*— Lifted Reed-Solomon Codes (Guo, Kopparty, Sudan 2013) were introduced in the context of locally correctable and testable codes. They are multivariate polynomials whose restriction to any line is a codeword of a Reed-Solomon code. We consider a generalization of their construction, which we call *lifted multiplicity codes*. These are multivariate polynomial codes whose restriction to any line is a codeword of a multiplicity code (Kopparty, Saraf, Yekhanin 2014). We show that lifted multiplicity codes have a better trade-off between redundancy and a notion of locality called the t-disjoint-repair-group property than previously known constructions. As a corollary, they also give better tradeoffs for PIR codes in the same parameter regimes. More precisely, we show that, for $t \leq \sqrt{N}$, lifted multiplicity codes with length N and redundancy $O(t^{0.585}\sqrt{N})$ have the property that any symbol of a codeword can be reconstructed in t different ways, each using a disjoint subset of the other coordinates. This gives the best known trade-off for this problem for any super-constant $t < \sqrt{N}$. We also give an alternative analysis of lifted Reed-Solomon codes using dual codes, which may be of independent interest.**

***Index Terms*— Error correction codes, lifted codes, multiplicity codes, locality, disjoint repair groups.**

I. Introduction

In THIS work we study *lifted multiplicity codes,* and show how they provide improved constructions of codes with the t-*disjoint repair group property* (t-DRGP), a notion of locality in error correcting codes.

An *error correcting code* of length N over an alphabet Σ is a set $\mathcal{C} \subseteq \Sigma^N$. There are several desirable properties in error correcting codes, and in this paper we study the trade-off between two of them. The first is the size of $\mathcal{C}$, which we would like to be as big as possible given N. The second desirable property is *locality.* Informally, a code $\mathcal{C}$ exhibits locality if, given (noisy) access to $c \in \mathcal{C}$, one can learn the i'th symbol c_i of c in sublinear time. As we discuss more below, locality arises in a number of areas, from distributed storage to complexity theory.

Two constructions of codes with locality are *lifted codes* [6] and *multiplicity codes* [14]; in fact, both of these constructions were among the first known high-rate Locally Correctable Codes. In this work, we consider a combination of the two ideas in *lifted multiplicity codes,* and we show that these codes exhibit locality beyond what's known for either lifted codes or for multiplicity codes.

More precisely, we study a particular notion of locality called the t-*disjoint-repair-group property* (t-DRGP). Informally, we say that $\mathcal{C}$ has the t-DRGP if any symbol c_i of $c \in \mathcal{C}$ can be obtained in t different ways, each of which involves a disjoint set of coordinates of c. Formally, we have the following definition.

Definition I.1: A code $\mathcal{C} \subseteq \Sigma^N$ has the t-*disjoint repair property* if for every $i \in [N]$, there is a collection of t disjoint subsets $S_1, \ldots, S_t \subseteq [N] \setminus \{i\}$, and functions $f_1, \ldots, f_t$ so that for all $c \in \mathcal{C}$ and for all $j \in [t]$, $f_j(c|_{S_j}) = c_i$. The sets $S_1, \ldots, S_t$ are called *repair groups.*

As discussed more in Section I-A below, the t-DRGP naturally interpolates between many different notions of locality. The t-DRGP is well-studied both when $t = O(1)$ is small (where it is related to Locally Repairable Codes and nearly equivalently to Private Information Retrieval Codes) and $t = \Omega(N)$ is large (where it is equivalent to Locally Correctable Codes). For this reason, it is natural to study the t-DRGP when t is intermediate; for example, when $t = N^a$ for $a \in (0, 1)$. In this case, it is possible for the size of the code $|\mathcal{C}|$ to be quite large: more precisely, it is possible for the *rate* $R = \frac{\log_{|\Sigma|} |\mathcal{C}|}{N}$ to approach 1 (notice that we always have $|\mathcal{C}| \leq |\Sigma|^N$, hence we always have $R \leq 1$). Thus, the goal is to understand exactly how quickly the rate can approach 1. That is, given t, how small can the *redundancy* $N - RN$ be?

Several works have tackled this question, and we illustrate previous results in Figure 1. Our main result is that lifted multiplicity codes improve on the best-known trade-offs for all super-constant $t \leq \sqrt{N}$.

a) Contributions: We summarize the main contributions of this work below.

1) For $t \leq \sqrt{N}$, we construct codes with the t-DRGP and redundancy at most
$$O\left(t^{\log_2(3)-1}\sqrt{N}\right) \approx O\left(t^{0.585}\sqrt{N}\right).$$
This gives the best known construction for all t with $t = \omega(1)$ and $t < \sqrt{N}$; the only previous result that held non-trivially for a range of t was redundancy

Manuscript received January 23, 2020; revised July 15, 2020; accepted October 21, 2020. Date of publication October 30, 2020; date of current version January 21, 2021. This work was supported in part by the National Science Foundation (NSF) under Grant DGE-1656518, Grant CCF-1657049, and Grant CCF-1844628; and in part by the Sloan Research Fellowship. This article was presented in part at RANDOM 2019. *(Corresponding author: Ray Li.)*

Ray Li is with the Department of Computer Science, Stanford University, Stanford, CA 94305 USA (e-mail: rayyli@cs.stanford.edu).

Mary Wootters is with the Department of Computer Science and the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: marykw@cs.stanford.edu).

Communicated by A. Mazumdar, Associate Editor for Coding Theory.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIT.2020.3034962.

Digital Object Identifier 10.1109/TIT.2020.3034962

0018-9448 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

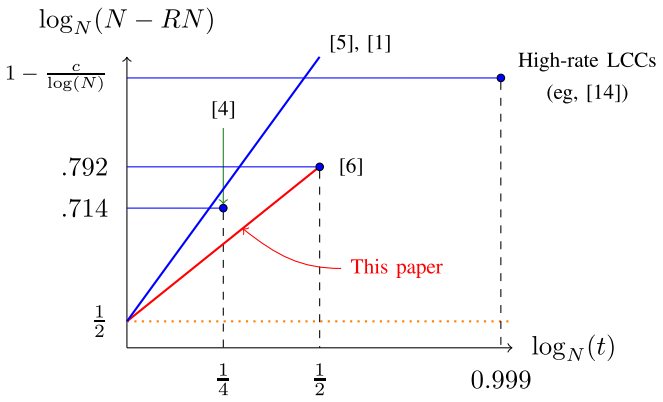

Fig. 1. The best trade-offs known between the number t of disjoint repair groups and the redundancy $N - RN$. Blue points and lines indicate upper bounds (possibility results), and the red line indicates our upper bound. The best lower bound (impossibility result) available is that we must have $\log_N((1-R)N) \geq 1/2$ for any $t \geq 2$, and this is shown as the dotted orange line.

$O(t\sqrt{N})$ [1], [2], [5] and our result also surpasses the specialized bound for $t = N^{1/4}$ of [4].
We note, however, that our construction has a large alphabet size, $N^{\Theta(N/t^2)}$. In contrast, the works [4]–[6] have alphabet size at most polynomial in N. However, we can follow the approach of [1] and make our code binary by replacing each symbol with an (uncoded) binary string. This yields *binary* codes with t-DRGP, that have the best known trade-offs between t and the redundancy when $N^{1/4} < t < N^{1/2}$ among all known codes with alphabet size $\text{poly}(N)$.

2) We give a new analysis of bivariate lifts of multiplicity codes. Both multiplicity codes and lifted codes have been studied before (even in the context of the t-DRGP), but to the best of our knowledge the only work to consider lifted multiplicity codes is [25]. That work studies m-variate lifts of multiplicity codes, where m is large; its goal is to obtain new constructions of high-rate locally correctable codes. In the context of our discussion, this corresponds to the t-DRGP when $t = N^{0.99}$. In contrast, for bivariate lifts, we are able to obtain more refined bounds which lead to improved results for the t-DRGP when $t \leq \sqrt{N}$.

b) Organization: In the remainder of the introduction, we survey related work and give an overview of our approach. In Section II, we give the formal definitions about polynomials and derivatives that we need. In Section III, we formally define lifted multiplicity codes. In Section IV, we prove that lifted multiplicity codes have high rate, and in Section V, we prove that they have the t-DRGP, which gives rise to our main theorem, Theorem I.2.

A. Background and Related Work

1) Disjoint Repair Groups: The t-DRGP and related notions have been studied both implicitly and explicitly across several communities. When $t = O(1)$ is small, several notions related to the t-DRGP have been studied, motivated primarily by distributed storage. These include Locally Repairable Codes (LRCs) with availability [17], [21], [22], [26], codes for Private Information Retrieval (PIR) [1], [2], [5] (all codes with the t-DRGP are t-PIR codes) and batch codes [1], [10], [18]; we refer the reader to [20] for a survey of these notions.[1]

To see why the t-DRGP might be relevant for distributed storage, consider a setting where some data is encoded as $c \in \mathcal{C}$, and then each c_i is sent to a separate server. If server i is later unavailable, we might want to reconstruct c_i without contacting too many other servers. This can be done if each symbol has one small repair group; this is the defining property of LRCs. Now suppose that several (say, $t-1$) servers are unavailable. If $\mathcal{C}$ has the t-DRGP then all $t-1$ unavailable symbols can be locally reconstructed: each node has at least t disjoint repair groups and at most $t-1$ of them have been compromised.

On the other hand, when $t = \Omega(N)$ is large, the t-DRGP has been studied in the context of Locally Decodable Codes and Locally Correctable Codes (LDCs/LCCs). In fact, the $\Omega(N)$-DRGP is equivalent to a constant-query LCC, and the notion has been used to prove impossibility results for such codes [15], [23].

Because of these motivations, there are several constructions of t-DRGP codes for a wide range of t; we illustrate the relevant ones in Figure 1. In the context of coded PIR, [1], [2], [5] give constructions of t-DRGP codes with redundancy $O(t\sqrt{N})$. This is known to be tight for $t = 2$ [19], [24], but no better lower bound is known.[2] When $t = \Omega(N)$ is very large, constructing codes with the t-DRGP is equivalent to constructing constant-query LCCs, and it is known that the rate of the code must tend to zero [23]. On the other hand, for any $\epsilon > 0$, when $t = O(N^{1-\epsilon})$ is just slightly smaller, then work on high-rate LCCs [6], [9], [11], [14] (see also [1]) imply that there are codes with rate 0.99 (or any constant less than 1) with the t-DRGP.[3]

When $t = \sqrt{N}$, there are a few constructions known that beat the $O(t\sqrt{N})$ bound mentioned above, including difference-set codes (see, e.g., [16]) and, relevant for us, lifted parity-check codes [6]. These constructions achieve redundancy $N^{\log_4(3)} \approx N^{0.79}$ when $t = \sqrt{N}$. In Appendix, we include a new proof of the fact that the lifted codes of [6] have this redundancy using a dual view of lifted codes.

When $t < \sqrt{N}$, there is only one construction known which beats the $O(t\sqrt{N})$ bound, due to [4]. For the special case of $t = N^{1/4}$, they give a construction based on "partially lifted codes" which has redundancy $O(N^{0.72}) = O(t^{0.88}\sqrt{N})$.

2) Lifting and Multiplicity Codes: Lifted multiplicity codes are based on lifted codes and multiplicity codes, both of which have a long history in the study of locality in error correcting codes.

a) Lifted Codes: Lifting was introduced by Guo, Kopparty and Sudan in [6]. The basic idea can be illustrated by Reed-Solomon (RS) codes. An RS code of degree d over $\mathbb{F}_q$

[1] In many (but not all) of these notions, we also care about the size of the repair groups but in this work we focus on the simpler problem of the t-DRGP.

[2] When the size s of the repair groups is bounded, it is known that the redundancy must be at least $\Omega(N \ln(t)/s)$ [22].

[3] In fact we may even take ϵ slightly sub-constant using the construction of [11].

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

is the code

$$\mathsf{RS}_{d,q} = \{(f(x_1), \ldots, f(x_q)) : f \in \mathbb{F}_q[X], \deg(f) < d\},$$

where $x_1, \ldots, x_q$ are the elements of $\mathbb{F}_q$. There is a natural multi-variate version of RS codes, known as Reed-Muller codes:

$$\mathsf{RM}_{d,q,m} = \left\{ \begin{array}{c} (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_{q^m})) : \\ f \in \mathbb{F}_q[X_1, \ldots, X_m], \deg(f) < d \end{array} \right\}$$

where $\mathbf{x}_1, \ldots, \mathbf{x}_{q^m}$ are the elements of $\mathbb{F}_q^m$. Reed-Muller codes have a very nice locality property, which is that the restriction of a RM codeword to a line in $\mathbb{F}_q^m$ yields an RS codeword. This fact has been taken advantage of extensively in applications like local decoding, local list-decoding and property testing. However, RM codes have a downside, which is that if $d < q$ (required for the above property to kick in), they have very low rate. With this inspiration, we could ask for the set $\mathcal{C}$ which contains evaluations of *all* m-variate polynomials which restrict to low-degree univariate polynomials on every line. Surprisingly, [6] showed that this set $\mathcal{C}$ can be much larger than the corresponding RM code! This code $\mathcal{C}$ is called a *lifted* Reed-Solomon code, and the main structural result of [6] is that $\mathcal{C}$ is the span of the monomials whose restrictions to lines are low-degree. This property is key when analyzing the rate of these codes. Moreover [6] showed that this is the case when we begin with *any* affine-invariant code, not just RS codes.

The original motivation for lifted codes was to construct LCCs, but [6] actually also give a code with the $\sqrt{N}$-DRGP, mentioned above; we give an alternate proof that this construction has the $\sqrt{N}$-DRGP in Appendix. A variant of lifting was also used in [4] to construct $N^{1/4}$-DRGP codes; however, the analysis of this construction is quite brittle and seems difficult to extend to non-trivial constructions for $t \neq N^{1/4}$.

b) Multiplicity Codes: Multiplicity codes were introduced by Kopparty, Saraf and Yekhanin [14] with the goal of constructing high-rate LCCs. The basic idea of multiplicity codes is to get around the low rate of RM codes discussed above in a different way, by appending derivative information to allow for higher-degree polynomials. That is, it is not useful to have an RS code with degree $d > q$, since $x^q = x$ for any $x \in \mathbb{F}_q$. However, if we replace the single evaluation $f(x)$ with a vector of evaluations $(f(x), f^{(1)}(x), \ldots, f^{(r-1)}(x))$, where $f^{(i)}$ denotes the i'th derivative, then it does make sense to take $d > q$. The m-variate multiplicity code $\mathsf{Mult}_{d,q,m,r}$ of degree d and order r over $\mathbb{F}_q$ is then defined similarly to $\mathsf{RM}_{d,q,m}$:

$$\mathsf{Mult}_{d,q,m,r} = \left\{ \begin{array}{c} (f^{(<r)}(\mathbf{x}_1), \ldots, f^{(<r)}(\mathbf{x}_{q^m})) : \\ f \in \mathbb{F}_q[X_1, \ldots, X_m], \deg(f) < d, \end{array} \right\}$$

where $f^{(<r)}(\mathbf{x}) \in \mathbb{F}_q^{\binom{m+r-1}{m}}$ is a vector containing all of the partial derivatives of f of order less than r, evaluated at $\mathbf{x}$. Since their introduction, multiplicity codes have found several uses beyond LCCs, including list-decoding [7], [12], and have even been used to explicitly construct codes with the t-DRGP [1].

c) Lifted Multiplicity Codes: To the best of our knowledge, the only work to study lifted multiplicity codes is the work of Wu [25]. The goal of that work is to obtain versions of multiplicity codes which are still high-rate LCCs but which require lower-order derivatives than the construction of [14]. The main result in [25] is that lifted multiplicity codes of rate $1 - \alpha$ are LCCs with locality N^{ϵ} (this corresponds roughly to having the t-DRGP with $t = O(N^{1-\epsilon})$). However, since the number of variables in the lift is large, it is hard to get a very precise handle on the codimension.

In comparison, in our work, we focus on the t-DRGP for $t \leq \sqrt{N}$, but where our goal is to get much tighter bound on the codimension of the code. We address the quantitative comparison between our bound on the rate and that obtainable by the techniques of [25] in Remark IV.5.

We note that the construction in [25] is similar to the construction presented here. Since this construction is somewhat non-trivial (for reasons discussed below), we include the details.

d) Why only bivariate lifts?: In contrast to [25], we study *bivariate* lifts of multiplicity codes. By focusing only on bivariate lifts (as was also done in [4]), we obtain a more precise handle on the codimension of lifted multiplicity codes, which gives results for the t-DRGP for $t \leq \sqrt{N}$. (See Remark IV.6 for more on why bivariate lifts make it much easier to analyze the codimension.) We believe that this wide range of t is interesting, and thus we think that bivariate lifts are worth focusing on.

We expect that lifted multiplicity codes can be analyzed over more variables. However, we expect that this will not improve the tradeoff between the redundancy and t (the number of repair groups) for the setting $t \leq \sqrt{N}$. Indeed, this tradeoff becomes worse for ordinary multiplicity codes [1]: for these codes, a larger number of variables yields better bounds only for larger values of t. In general, m-variable lifted multiplicity codes can have up to $q^{m-1} = N^{(m-1)/m}$ disjoint repair groups, so $\lceil 1/\varepsilon \rceil$ variables are needed for $N^{1-\varepsilon}$ repair groups. For $N^{(m-2)/(m-1)} \leq t \leq N^{(m-1)/m}$, we expect that the number of variables that gives the best rate for lifted multiplicity codes is m. We leave the analysis for more variables m this for future work (see Section VI).

B. Our Approach

We study lifted multiplicity codes to obtain improved constructions of codes with the t-DRGP. We focus on bivariate lifts in this paper in order to obtain codes with t-DRGP for $t \leq \sqrt{N}$. We expect that lifted multiplicity codes in more than two variables also give better codes for the t-DRGP when $t > \sqrt{N}$.

1) Definition of Lifted Multiplicity Codes: It is not immediately obvious how to apply lifting (and in particular, the nice characterization of it developed in [6] as the span of "good" monomials) to univariate multiplicity codes. We first note that the univariate multiplicity code $\mathsf{Mult}_{d,q,1,r} \subseteq \left(\mathbb{F}_q^r\right)^q$ does not fit the affine-invariant framework of [6], so their results do not immediately apply. Instead, we might try to define the bivariate lift of $\mathsf{Mult}_{d,q,1,r}$ as the set of vectors $(f^{(<r)}(\mathbf{x}_1), \ldots, f^{(<r)}(\mathbf{x}_{q^2}))$ for all polynomials f so that every restriction of f to a line agrees with some polynomial of degree less than d on its first $r-1$ derivatives; that is, the restriction of f is *equivalent up to order* r to a polynomial

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

of degree less than d. This works, but there are two non-trivial things to deal with.

1) First, in order to get a handle on the rate of the code, as in [6] we show that the set of valid polynomials f includes the span of a large set of "good" monomials. In contrast to [6], the good monomials in this work do not span the entire code. However, lower bounding the number of good monomials, which in turns gives a lower bound on the rate of the code, turns out to be enough for our results.
2) Second, we need to take some care about what monomials we allow. With lifted RS codes, one only allows monomials X^aY^b with individual degrees $a, b < q$; otherwise, we could have multiple monomials which correspond to the same codeword which leads to problems if we are counting monomials in order to understand the dimension of the code. As we show in Lemma III.5, it turns out that with multiplicity codes, we should only allow monomials X^aY^b with $\lfloor a/q \rfloor + \lfloor b/q \rfloor < r$; otherwise, we would have multiple monomials the correspond to the same codeword and this would create similar problems.

Dealing with these issues leads us to the final code and rate analysis, where we define the lifted multiplicity code to be all polynomials spanned by monomials X^aY^b with $\lfloor a/q \rfloor + \lfloor b/q \rfloor < r$, such that the restriction of the polynomial to a line is equivalent up to order r to some univariate polynomial of degree less than d. We then lower bound the number of evaluations of monomials in this code, giving a lower bound on the rate. We note that the work [25] considers a similar construction.

2) Lifted Multiplicity Codes Have the t-DRGP: In Corollary IV.3 we give a lower bound on the number of (q, r, d)-good monomials, and this leads to a lower bound on the dimension of the lifted multiplicity code; crucially, this can be quite a bit bigger than the dimension of the corresponding multivariate multiplicity code.

Finally, we observe that lifted multiplicity codes have the t-DRGP for a range of values of t. Similarly to previous constructions based on multivariate polynomial codes, the disjoint repair groups to recover the symbol $f^{(<r)}(\mathbf{x})$ are given by disjoint collections of lines through $\mathbf{x}$. More precisely, the values $f^{(<r)}(\mathbf{y})$ for the set of $\mathbf{y}$ that lie on r distinct lines through $\mathbf{x}$ can be used to recover $f^{(<r)}(\mathbf{x})$. Thus, the number of disjoint repair groups is $q/r = \sqrt{N}/r$. By adjusting r, we obtain the trade-off shown in Figure 1. Our main theorem is as follows.

Theorem I.2: For $q = 2^\ell$ and $r = 2^{\ell'}$ with $1 \le \ell' \le \ell$, there exists a code $\mathcal{C}$ over $\mathbb{F}_q^{\binom{r+1}{2}}$ with the following properties.

- The length of the code is q^2.
- The rate of the code is at least
$$1 - \frac{3r^{\log_2(8/3)}q^{\log_2(3)}}{\binom{r+1}{2}q^2},$$
so that the redundancy is at most
$$\frac{3r^{\log_2(8/3)}q^{\log_2(3)}}{\binom{r+1}{2}}.$$
- The code has the q/r-disjoint repair group property.

As a remark, our techniques can also recover any symbol from any one of its repair groups in polynomial time. For any $\gamma \in [0, 1]$, choosing $q = 2^\ell$ and $r = 2^{\ell'}$ with $\gamma \approx \ell'/\ell$ gives a code with length $N = q^2$ and redundancy at most
$$6N^{\log_4(3)-\gamma(1-\log_4(8/3))}$$
with the $N^{(1-\gamma)/2}$-DRGP. This is made formal in the following corollary.

Corollary I.3: For any $\epsilon \in (0, \frac{1}{2})$, there are infinitely many N so that, for $t = \lfloor N^\epsilon \rfloor$, there exists a code of length N which has the t-DRGP and redundancy at most $6\ t^{\log_2(3)-1}\sqrt{N}$.

We note that Theorem I.2 also yields results for constant t, not just for $t = N^\epsilon$ as presented in Corollary I.3. For example, by setting $r = q/2$ we obtain a code with the 2-DRGP and redundancy at most $9\sqrt{N}$. The constant 9 is not optimal here (the optimal constant for $t = 2$ is known to be $\sqrt{2}$ [19]), but to the best of our knowledge, Theorem I.2 does yield the best known bounds for any super-constant t.

The codes in Theorem I.2 and Corollary I.3 have the disadvantage of having a large alphabet size. Indeed, we have $r = q/t$, and so the alphabet size is $Q = q^{\binom{r+1}{2}} = N^{\Theta(N/t^2)}$, which is very large. It is an interesting question to obtain the results of Corollary I.3 with a code over a smaller alphabet (see open questions in Section VI). Among the existing work in Figure 1, [4]–[6] all have $\mathrm{poly}(N)$ or smaller sized alphabets.

For now, we observe as in [1] that, if $\mathcal{C}$ is a code with the t-DRGP, then replacing $\mathcal{C}$ with a binary code $\mathcal{C}'$, where each symbol in each codeword is replaced with $\log(Q)$ binary bits, yields a code that also has the t-DRGP. As a result, applying this to the code in Corollary I.3 yields a code with length $N_{bin} = N\log(Q) = N^{2-2\varepsilon}$ and redundancy $O(t^{\log_2(3/2)}\sqrt{N}\log(Q)) = O(N^{3/2+\varepsilon\log_2(3/8)}\log N)$.

Corollary I.4: For any $\epsilon \in (0, \frac{1}{2})$, there are infinitely many N so that, for $t = \lfloor N^{\frac{\epsilon}{2-2\varepsilon}} \rfloor$, there exists a *binary* code of length N which has the t-DRGP and redundancy at most $\tilde{O}(N^{\frac{3/2+\varepsilon\log_2(3/8)}{2-2\varepsilon}})$.

Among codes with alphabet size $\mathrm{poly}(N)$ or smaller, our binary codes give the best known tradeoff between t and redundancy when $N^{1/4} < t < N^{1/2}$ (at $t = N^{1/4}$ [4] gives a better redundancy).

II. Preliminaries

In this section, we introduce the background we need on polynomials and derivatives over finite fields. Throughout this paper, we assume that q is a power of 2. Let $\mathbb{F}_q$ denote the finite field of order q, and let $\mathbb{F}_q^*$ denote its multiplicative subgroup.

If a and b are nonnegative integers with binary representations $a = \overline{a_{\ell-1}\cdots a_0}$ and $b = \overline{b_{\ell-1}\cdots b_0}$, then we write $a \le_2 b$ if $a_i \le b_i$ for $i = 0, \ldots, \ell-1$. If a is an integer, let $(a \mod c)$ denote the element of $\{0, \ldots, c-1\}$ congruent to a mod c. We write $a \le_2^\ell b$ if $(a \mod 2^\ell) \le_2 (b \mod 2^\ell)$.

As in [6], we use Lucas's theorem.

Proposition II.1 (Lucas's Theorem): Let p be a prime and $a = \overline{a_{\ell-1}\cdots a_0}, b = \overline{b_{\ell-1}\cdots b_0}$ be written in base p. Then
$$\binom{a}{b} \equiv \prod_{i=0}^{\ell-1}\binom{a_i}{b_i} \mod p \tag{1}$$

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

In particular, if $p = 2$, then $\binom{a}{b} \equiv 1 \mod p$ if and only if $b \leq_2 a$.

A. Polynomials and Derivatives

For a vector $\mathbf{i} = (i_1, \ldots, i_m)$ of nonnegative integers, its *weight*, denoted $\text{wt}(\mathbf{i})$, equals $\sum_{k=1}^{m} i_k$. For a field $\mathbb{F}$, let $\mathbb{F}[X_1, \ldots, X_m] = \mathbb{F}[\mathbf{X}]$ be the ring of polynomials in the variables $X_1, \ldots, X_m$ with coefficients in $\mathbb{F}$. For a vector of nonnegative integers $\mathbf{i} = (i_1, \ldots, i_m)$ and a vector $\mathbf{X} = (X_1, \ldots, X_m)$ of variables, let $\mathbf{X}^{\mathbf{i}}$ denote the monomial $\prod_{j=1}^{m} X_j^{i_j} \in \mathbb{F}[\mathbf{X}]$, and for a vector $\mathbf{a} = (\alpha_1, \ldots, \alpha_m) \in \mathbb{F}^m$, let $\mathbf{a}^{\mathbf{i}}$ denote the value $\prod_{j=1}^{m} \alpha_j^{i_j}$, where $0^0 := 1$. For nonnegative vectors $\mathbf{i} = (i_1, \ldots, i_m)$ and $\mathbf{j} = (j_1, \ldots, j_m)$, we write $\mathbf{i} \leq \mathbf{j}$ if $i_k \leq j_k$ for all k. We also write $\binom{\mathbf{i}+\mathbf{j}}{\mathbf{i}}$ to denote $\prod_{k=1}^{m} \binom{i_k+j_k}{i_k}$. For nonnegative vector $\mathbf{i}$, we let $[\mathbf{X}^{\mathbf{i}}]P(\mathbf{X})$ denote the coefficient of $\mathbf{X}^{\mathbf{i}}$ in the polynomial $P(\mathbf{X})$.

We will use Hasse derivatives, a notion of derivatives over finite fields:

Definition II.2 (Hasse Derivatives): For $P(\mathbf{X}) \in \mathbb{F}[\mathbf{X}]$ and a nonnegative vector $\mathbf{i}$, the $\mathbf{i}$-th (Hasse) derivative of P, denoted $P^{(\mathbf{i})}(\mathbf{X})$ or $D^{(\mathbf{i})}P(\mathbf{X})$, is the coefficient of $\mathbf{Z}^{\mathbf{i}}$ in the polynomial $\tilde{P}(\mathbf{X}, \mathbf{Z}) := P(\mathbf{X} + \mathbf{Z}) \in \mathbb{F}[\mathbf{X}, \mathbf{Z}]$. Thus,

$$P(\mathbf{X} + \mathbf{Z}) = \sum_{\mathbf{i}} P^{(\mathbf{i})}(\mathbf{X})\mathbf{Z}^{i}. \tag{2}$$

For $\mathbf{x} \in \mathbb{F}_q^m$ and $P(X) \in \mathbb{F}_q[\mathbf{X}]$, we use the notation $P^{(<r)}(\mathbf{x}) \in \mathbb{F}_q^{\binom{m+r-1}{m}}$ to denote the vector containing $P^{(\mathbf{i})}(\mathbf{x})$ for all $\mathbf{i}$ so that $\text{wt}(\mathbf{i}) < r$. We record a few useful (well-known) properties of Hasse derivatives below (see [8]).

Proposition II.3 (Properties of Hasse Derivatives): Let $P(\mathbf{X}), Q(\mathbf{X}) \in \mathbb{F}[\mathbf{X}]$ and let $\mathbf{i}, \mathbf{j}$ be vectors of nonnegative integers. Then

1) $P^{(\mathbf{i})}(\mathbf{X}) + Q^{(\mathbf{i})}(\mathbf{X}) = (P + Q)^{(\mathbf{i})}(\mathbf{X})$.
2) $(P \cdot Q)^{(\mathbf{i})}(\mathbf{X}) = \sum_{0 \leq \mathbf{e} \leq \mathbf{i}} P^{(\mathbf{e})}(\mathbf{X}) \cdot Q^{(\mathbf{i}-\mathbf{e})}(\mathbf{X})$.
3) $(P^{(\mathbf{i})})^{(\mathbf{j})}(\mathbf{X}) = \binom{\mathbf{i}+\mathbf{j}}{\mathbf{i}} P^{(\mathbf{i}+\mathbf{j})}(\mathbf{X})$.

Using the above, we obtain the following useful derivative computation, and we provide a proof in Appendix for completeness.

Proposition II.4: Let $1 \leq r < q$ with q a power of 2, and let $P(X) = (X^q - X)^r$. Then,

$$P^{(i)}(X) = \begin{cases} \binom{r}{i}(X^q - X)^{r-i} & 0 \leq i \leq r \\ 0 & i > r \end{cases} \tag{3}$$

B. Polynomial Local Recovery

A key property exploited by earlier work on multiplicity codes [13], [14] is that $f^{(<r)}(\mathbf{x})$ can be recovered from $f^{(<q)}(\mathbf{y})$ for $\mathbf{y}$ that lie on a collection of lines through $\mathbf{x}$. More precisely, let $\mathcal{L}_m$ be the set of lines $L(T)$ of the form $\mathbf{a}T + \mathbf{b}$ with $\mathbf{a}, \mathbf{b} \in \mathbb{F}_q^m$. Given a multivariate polynomial $P(\mathbf{X}) \in \mathbb{F}_q[X_1, \ldots, X_m]$, if L is the line $\mathbf{a}T + \mathbf{b}$, let $P_L(T) \in \mathbb{F}_q[T]$ denote the univariate polynomial $P(\mathbf{a}T + \mathbf{b})$. Let $\mathcal{L}$ be the set of lines in $\mathbb{F}_q^2$ of the form $L(T) = (T, \alpha T + \beta)$ for $\alpha, \beta \in \mathbb{F}_q$.

For simplicity—and because it is enough for our application to the t-DRGP—we will consider only bivariate polynomials in this paper, although (see for example [13]) the same basic idea works for any m. We will further specialize to lines in $\mathcal{L}$—that is, lines of the form $L(T) = (T, \alpha T + \beta)$—because it will simplify some computations later in the paper. With these restrictions, we can specialize Equation (4) of [13] to obtain the following relationship between the derivatives of $P_L(T)$ and the derivatives of $P(X, Y)$.

Lemma II.5 (Follows From, e.g., [13], [14]): Suppose that $L_1, \ldots, L_r$ are r lines in $\mathcal{L}$ all passing through a point (γ, δ), with L_i being the line $(T, \alpha_i T + \beta_i)$. Then, for all polynomials $P(X, Y) \in \mathbb{F}_q[X, Y]$, the following matrix equality holds for all $i = 0, \ldots, r - 1$.

$$\begin{bmatrix} P_{L_1}^{(i)}(\gamma) \\ P_{L_2}^{(i)}(\gamma) \\ \vdots \\ P_{L_{i+1}}^{(i)}(\gamma) \end{bmatrix} = \begin{bmatrix} \alpha_1^0 & \alpha_1^1 & \cdots & \alpha_1^i \\ \alpha_2^0 & \alpha_2^1 & \cdots & \alpha_2^i \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{i+1}^0 & \alpha_{i+1}^1 & \cdots & \alpha_{i+1}^i \end{bmatrix} \begin{bmatrix} P^{(i,0)}(\gamma, \delta) \\ P^{(i-1,1)}(\gamma, \delta) \\ \vdots \\ P^{(0,i)}(\gamma, \delta) \end{bmatrix}. \tag{4}$$

When lines $L_1, \ldots, L_i$ are distinct, the middle matrix in (4) is a Vandermonde matrix, and Vandermonde matrices are invertible in polynomial time. Hence, we immediately have the following corollary.

Corollary II.6: Suppose that $L_1, \ldots, L_r$ are r distinct lines of the form $L_k(T) = (T, \alpha_k T + \beta_k)$ all passing through a point $(\gamma, \delta) \in \mathbb{F}_q^2$. For a polynomial $P(X, Y) \in \mathbb{F}_q[X, Y]$, given the polynomials $P_{L_1}(T), \ldots, P_{L_k}(T)$, the derivatives $P^{(\mathbf{i})}(\gamma, \delta)$ are uniquely determined and computable efficiently for all $\mathbf{i}$ such that $\text{wt}(\mathbf{i}) < r$.

III. Lifted Multiplicity Codes

In this section, we define lifted multiplicity codes. As noted in the introduction, we restrict our attention to bivariate codes because this is enough for our application to the t-DRGP. However, everything in this section extends to general m-variate codes. We define bivariate lifted multiplicity codes as the vectors $(f^{(<r)}(\mathbf{x}))_{\mathbf{x} \in \mathbb{F}_q^2}$ for polynomials $f(X)$ that live in the span of "good" monomials. In order to define these "good" monomials, we need a few more definitions.

A. Polynomial Equivalence

We first define a notion of polynomial equivalence.

Definition III.1: We say that two univariate polynomials $A(X), B(X) \in \mathbb{F}_q[X]$ are *equivalent up to order* r, written $A \equiv_r B$, if $A^{(i)}(\gamma) = B^{(i)}(\gamma)$ for all $i = 0, \ldots, r - 1$ and $\gamma \in \mathbb{F}_q$.

It is easy to see that the above definition does in fact give an equivalence relation. We now present two standard results regarding this equivalence relation. The first is a characterization of this equivalence.

Lemma III.2: For $A(X), B(X) \in \mathbb{F}_q[X]$ we have $A(X) \equiv_r B(X)$ if and only if $(X^q - X)^r | A(X) - B(X)$.

Proof: By considering the polynomial $A(X) - B(X)$, it suffices to prove $A(X)$ is equivalent to the zero polynomial up to order r if and only if $(X^q - X)^r | A(X)$. If $A(X) = (X^q - X)^r C(X)$ for some polynomial $C(X) \in \mathbb{F}_q[X]$, then, by part 2 of Proposition II.3 and Proposition II.4, for $0 \leq i < r$, we have $X^q - X | A^{(i)}(X)$, so $A^{(i)}(\gamma) = 0$ for all $0 \leq i < r$ and all $\gamma \in \mathbb{F}_q$, so $A(X) \equiv_r 0$.

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

Conversely, suppose that $A(X) \equiv_r 0$. By the definition of Hasse derivatives, we have $A(X) = A(\gamma + (X-\gamma)) = \sum_i A^{(i)}(\gamma)(X-\gamma)^i$. Since $A^{(i)}(\gamma) = 0$ for $i = 0, \ldots, r-1$, we have $(X-\gamma)^r | A(X)$. Thus is true for all γ, so $\prod_\gamma (X-\gamma)^r | A(X)$, so $(X^q - X)^r | A(X)$. □

Lemma III.2 gives the following corollary.

Lemma III.3: Let q be a power of 2 and $r \geq 1$. For every univariate polynomial $A(X)$, there exists a unique degree-at-most $rq-1$ polynomial $B(X)$ such that $A(X) \equiv_r B(X)$. Furthermore, if r is a power of 2, then for all a such that $\deg A - (qr - r) < a < qr$, we have $[X^a]A(X) = [X^a]B(X)$.

Proof: For existence of $B(X)$, note that, by Lemma III.2, we can take $B(X)$ to be the remainder when $A(X)$ is divided by $(X^q - X)^r$. For uniqueness of $B(X)$, suppose that $B_1(X)$ and $B_2(X)$ are equivalent to $A(X)$ up to order r and are of degree at most $rq-1$. By Lemma III.2, we have $(X^q - X)^r | B_1(X) - B_2(X)$. Additionally, $B_1(X) - B_2(X)$ has degree at most $rq-1$, so $B_1(X) - B_2(X) = 0$.

Now suppose r is a power of 2. Then $(X^q - X)^r = X^{rq} + X^r$. Above, to obtain $B(X)$ from $A(X)$, we need only to subtract terms of the form $X^{qr} + X^r, X^{qr+1} + X^{r+1}, \ldots, X^{\deg A} + X^{\deg A - qr + r}$. Thus, for a such that $\deg A - qr + r < a < qr$, the coefficients of X^a in $A(X)$ and $B(X)$ are equal. □

B. Type-r Polynomials

Define the order-r evaluation map $\text{eval}_{q,r} : \mathbb{F}_q[X,Y] \to \left(\mathbb{F}_q^{\binom{r+1}{2}}\right)^{q^2}$ by

$$\text{eval}_{q,r}(P) := (P^{(<r)}(\mathbf{x}))_{\mathbf{x} \in \mathbb{F}_q^2}, \tag{5}$$

We will want to restrict our attention to a subset of monomials $M(X,Y) = X^a Y^b$ whose order-r evaluations $\text{eval}_{q,r}(M)$ form a basis for the space $\{\text{eval}_{q,r}(P) : P \in \mathbb{F}_q[X,Y]\}$. To that end, we introduce the following definition.

Definition III.4 (Type-r Monomials): Call a monomial $X^a Y^b$ *type-r* if $\lfloor a/q \rfloor + \lfloor b/q \rfloor \leq r-1$. Let $\mathcal{F}_{q,r}$ be the family of polynomials $P \in \mathbb{F}_q[X,Y]$ that are spanned by type-r monomials.

It is easy to see that $\mathcal{F}_{q,r}$ is a dimension $\binom{r+1}{2}q^2$ vector space over $\mathbb{F}_q$. We now show that the type-r polynomials form a basis for bivariate polynomials, up to order r equivalence. We note that Lemma III.1 of [25] claims a similar statement, with a different argument.

Lemma III.5: The evaluation map $\text{eval}_{q,r} : \mathcal{F}_{q,r} \to \left(\mathbb{F}_q^{\binom{r+1}{2}}\right)^{q^2}$ is a bijection.

Proof of Lemma III.5: Since $\text{eval}_{q,r}$ is a linear map and $\mathcal{F}_{q,r}$ and $\mathbb{F}_q^{\binom{r+1}{2}q^2}$ have the same $\mathbb{F}_q$ dimension, it suffices to prove the map has trivial kernel. We prove by induction.

Base Case: $r = 1$**.** Suppose $P \in \mathcal{F}_{q,1}$ and $\text{eval}_{q,1}(P)$ is the 0-vector. Then $P(X,Y) = 0$ for all X, Y. For any $\delta \in \mathbb{F}_q$, the polynomial $P(X,\delta) \in \mathbb{F}_q[X]$ has degree at most $q-1$ but has q roots, so the polynomial must be 0. Hence, $(Y-\delta) | P(X,Y)$ for all δ, so $(Y^q - Y) | P(X,Y)$, which implies $P = 0$. This proves that $\text{eval}_{q,1}$ has trivial kernel.

Inductive step: Assume $r \geq 1$ and $\text{eval}_{q,r}$ has trivial kernel. We prove that $\text{eval}_{q,r+1}$ has trivial kernel.

Assume $P(X,Y)$ is a polynomial spanned by type-$(r+1)$ monomials with all **i**th derivatives equal to 0 for $\text{wt}(\mathbf{i}) < r+1$. Let $\delta \in \mathbb{F}_q$ and $B_\delta(X) := P(X,\delta)$. Then, for $0 \leq i < r$, we have $B_\delta^{(i)}(\gamma) = P^{(i,0)}(\gamma,\delta) = 0$ for all $\gamma \in \mathbb{F}_q$. Hence, for all $\gamma \in \mathbb{F}_q$, we have $(X-\gamma)^r | B_\delta(X)$. Hence, $(X^q - X)^r | B_\delta(X)$. Since $\deg B_\delta(X) \leq \deg_X P(X,Y) < qr$ for all δ, we have $B_\delta(X) = 0$. Thus, $P(X,\delta)$ is the 0 polynomial for all δ, so $(Y-\delta) | P(X,Y)$ for all δ, so $(Y^q - Y) | P(X,Y)$. Hence, we may write $P(X,Y) = (Y^q - Y)Q(X,Y)$ for some polynomial $Q(X,Y) \in \mathbb{F}_q[X,Y]$.

As polynomial P is type-$(r+1)$, polynomial Q is type-r: if Q had a nonzero coefficient for $X^a Y^b$ with $\lfloor a/q \rfloor + \lfloor b/q \rfloor > r-1$, then the coefficient $X^a Y^{b+q}$ is nonzero in P, which is a contradiction. For all i, j with $i \geq 0, j \geq 1$ and $i + j \leq r$, we have

$$P^{(i,j)}(X,Y) \;=\; (Y^q - Y)Q^{(i,j)}(X,Y) - Q^{(i,j-1)}(X,Y). \tag{6}$$

Here we applied part 2 of Proposition II.3 and the $r = 1$ case of Proposition II.4. At every X and Y, the left side is 0 by assumption on P and the right side $Q^{(i,j-1)}(X,Y)$. We conclude that $Q^{(i',j')}$ evaluates to 0 everywhere for every nonnegative i' and j' satisfying $i' + j' \leq r-1$. Since Q is type-r, we have $Q = 0$ by the induction hypothesis, so $P = 0$. This completes the induction, completing the proof. □

C. Definition of Lifted Multiplicity Codes

Finally we are ready to define lifted multiplicity codes, which we define as the set of evaluations $\text{eval}_{q,r}(P)$ of polynomials whose restrictions to lines[4] are equivalent, up to order r, to a low degree polynomial:

Definition III.6 (Lifted Multiplicity Codes, First Definition): The (q,r,d) *(bivariate) lifted multiplicity code* is a code $\mathcal{C}$ over alphabet $\Sigma = \mathbb{F}_q^{\binom{r+1}{2}}$ of length q^2 given by

$$\mathcal{C} = \left\{ \text{eval}_{q,r}(P) : \begin{array}{c} P \in \mathbb{F}_q[X,Y] \text{ and, for any } L(T) \in \mathcal{L}, \\ P(L(T)) \equiv_r Q(T) \text{ for some} \\ Q \in \mathbb{F}_q[T] \text{ of degree less than } d. \end{array} \right\}$$

Definition III.6 is natural but difficult to get a handle on directly. Following the approach of previous work [4], [6], we show that lifted multiplicity code contains the set of vectors $\text{eval}_{q,r}(P)$ for P that lie in the span of a set of "good" monomials, which makes it easier to bound the rate. Informally, a monomial is (q,r,d)-good if its restriction along every line is equivalent, up to order r, to a polynomial of degree less than d.

Definition III.7 ((q,r,d)-Good Monomials): Call a monomial $M_{a,b}(X,Y) = X^a Y^b \in \mathbb{F}_q[X,Y]$ (q,r,d)-*good* (or simply good, when r and d are understood) if it is type-r and for every line $(T, \alpha T + \beta) \in \mathcal{L}$, the univariate polynomial $M_{a,b}(T, \alpha T + \beta)$ is equivalent, up to order r, to polynomial of degree less than d, and call it (q,r,d)-*bad* otherwise.

[4]To simplify calculations, we consider restrictions to lines of the form $L(T) = (T, \alpha T + \beta)$. That is, we do not include lines of the form $L(T) = (\alpha, T)$.

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

By definition all good monomials lie in our lifted multiplicity code, so to lower bound the rate of the code it suffices to lower bound the number of good monomials.

Lemma III.8: Let $\mathcal{C}$ be the bivariate (q, r, d) lifted multiplicity code. Then, for every (q, r, d)-good monomial $M(X, Y)$, $\text{eval}_{q,r}(M) \in \mathcal{C}$, and the rate of $\mathcal{C}$ is at least $\frac{\#(q,r,d)\text{-good monomials}}{\binom{r+1}{2}q^2}$.

Proof: The first part follows from the definition of good monomial. For the second part, $\mathcal{C}$ is linear and the $\mathbb{F}_q$-span of all good monomials have pairwise distinct evaluations by Lemma III.5, so $|\mathcal{C}| \geq q^{(\#(q,r,d)\text{-good monomials})}$. As $\mathcal{C}$ is a length q^2 code over an alphabet of size $|\Sigma| = q^{\binom{r+1}{2}}$, the rate is at least $\frac{\log|\mathcal{C}|}{q^2 \log|\Sigma|} = \frac{\#(q,r,d)\text{-good monomials}}{\binom{r+1}{2}q^2}$. □

Remark III.9: A previous version of this paper incorrectly asserted that every codeword of the lifted multiplicity code is spanned by good monomials. As observed by Nikita Polianskii, this is in fact not true. For example, when $r = 2$ and $d = 2q - 1$, the monomials $X^{2q-2}Y$ and $X^{q-1}Y^q$ are not (q, r, d)-good as verified by the line (T, T), but their sum $X^{2q-2}Y + X^{q-1}Y^q$ is in the (q, r, d)-lifted multiplicity code: the restriction of the sum to a line $(T, \alpha T + \beta) \in \mathcal{L}$ has a T^{2q-1} coefficient of $\alpha + \alpha^q = 0$ and hence has degree strictly less than $d = 2q - 1$.

IV. The Rate of Lifted Multiplicity Codes

In this section, we bound the rate (and hence, the redundancy) of lifted multiplicity codes. Our final result on the rate is Corollary IV.3 below, which implies that for r, q and d of an appropriate form, the lifted multiplicity code over order r and degree d over $\mathbb{F}_q$ has rate at least

$$1 - \frac{6}{r}\left(r - \frac{d}{q}\right)^{\log_2(4/3)}.$$

In the next section, we choose $d = qr - r$, which will yield a code of rate $1 - \frac{6}{r}\left(\frac{r}{q}\right)^{\log_2(4/3)}$ and will give us Theorem I.2.

Before we prove this result, we briefly compare our approach to more straightforward ones, and discuss why we are able to do better.

First, we discuss what might be a first strategy building on the analysis of [6] for lifted Reed-Solomon codes. Similarly to that work, we want to show there are few bad monomials. We can show (after checking some conditions) that a monomial is bad if, restricted to some line, in the resulting univariate polynomial, one of the coefficients of $T^{qr-s}, T^{qr-s+1}, \ldots, T^{qr-1}$ is nonzero. This corresponds to the analysis of lifted Reed-Solomon codes when $r = s = 1$. For each $s' = 1, \ldots, s$, similar to the analysis of the lifted Reed-Solomon code, we can bound the number of monomials that could cause the coefficient of $T^{rq-s'}$ to be nonzero by $rq^{\log_2(3)}$. Using the union bound and summing these bounds gives a bound $rsq^{\log_2(3)}$ on the number of bad monomials for the lifted multiplicity code. However, when $r = s$ (the setting we will consider), this gives a rate of $1 - q^{\log_2(3/4)}$. Thus, this yields a code with the same redundancy of $N^{\log_4(3)}$ as the lifted Reed-Solomon code, and we have made no improvement.

In order to do better, the key to our analysis is to observe that monomials that are bad for some s' are likely to be bad for another s'', so the union bound is wasteful. Instead, using some tricks with binary arithmetic (captured in Lemma IV.1), we are able to analyze together all the monomials that make any of the coefficients of $T^{rq-s}, \ldots, T^{rq-1}$ nonzero, giving a better bound.

Second, we compare our approach to the analysis of [25], which also studies lifted multiplicity codes, but focuses on a different parameter regime (one where t is much larger). As described more in Remark IV.5, the approach of [25] does not yield anything better in the parameter regime that we consider ($t \leq \sqrt{N}$) than does the approach described above (or indeed even any better than standard (not lifted) multiplicity codes when $r \gg q^{1/23}$). The reason that we are able to do better than the straightforward argument above while the approach of [25] does not is that [25] uses a stricter requirement for a monomial to be good in [25, Lemma III.3] than we do in our Lemma IV.2. Thus, the approach of [25] counts a smaller number of good monomials and ends up with a weaker bound on the rate.

Now, we prove our result. We begin with a lemma that will be useful.

Lemma IV.1: Let $s = 2^{\ell_s}$ and $q = 2^\ell$ with $\ell_s \leq \ell$. The number of $a_1, b_1 \in \{0, 1, \ldots, q-1\}$ such that at least one of the following is true

$$\begin{aligned} q - 1 - a_1 &\leq_2^\ell b_1 \\ q - 2 - a_1 &\leq_2^\ell b_1 \\ &\vdots \\ q - s - a_1 &\leq_2^\ell b_1 \end{aligned} \tag{7}$$

is at most $2 \cdot 3^\ell \cdot (4/3)^{\ell_s} = 2 \cdot 3^\ell \cdot s^{\log_2(4/3)}$.

Proof: Suppose we write the numbers $(q - 1 - a_1 \mod q), (q - 2 - a_1 \mod q), \ldots, (q - s - a_1 \mod q)$ in binary with ℓ digits (possibly with leading zeros). As these number span 2^{ℓ_s} consecutive integers mod q, when written in this binary form, their most significant $\ell - \ell_s$ coordinates take on at most 2 values. Let $a_2 = \lfloor \frac{(q-1-a_1 \mod q)}{2^{\ell_s}} \rfloor$ and $b_2 = \lfloor \frac{b_1}{2^{\ell_s}} \rfloor$ so that $a_2, b_2 \in \{0, \ldots, 2^{\ell-\ell_s} - 1\}$, and a_2 and b_2 are the most significant $\ell - \ell_s$ coordinates of $(q - 1 - a_1 \mod q)$ and b_1, respectively, when written in ℓ-digit binary. Then if one of the equations of (7) is true, then we must have either $a_2 \leq_2 b_2$ or $a_2 - 1 \leq_2 b_2$. This gives at most $2 \cdot 3^{\ell-\ell_s}$ choices for the pair (a_2, b_2). Given a_2 and b_2, there are 2^{ℓ_s} choices for each of a_1 and b_1, for a total of at most $2 \cdot 3^{\ell-\ell_s} \cdot 4^{\ell_s}$ solutions to (7). □

Lemma IV.2: Let $r = 2^{\ell_r}$, $s = 2^{\ell_s}$ and $q = 2^\ell$ with $\ell_r, \ell_s \in \{1, \ldots, \ell - 1\}$. The number of $(q, r, rq - s)$-good monomials is at least $\binom{r+1}{2}4^\ell - 3rs^{\log_2(4/3)} \cdot 3^\ell$.

Proof: The number of type-r monomials is $\binom{r+1}{2}q^2 = \binom{r+1}{2}4^\ell$. A monomial $M_{a,b}$ is $(q, r, rq - s)$-good if, for every $\alpha, \beta \in \mathbb{F}_q$, we have

$$M_{a,b,\alpha,\beta}(T) \;\coloneqq\; T^a(\alpha T + \beta)^b = \sum_{i=0}^{b} \alpha^i \beta^{b-i} T^{a+i} \binom{b}{i}. \tag{8}$$

can be represented as a polynomial of degree less than $rq - s$. Next, we apply Lemma III.3, which says that there is a unique polynomial $B(T)$ so that $\deg(B) \leq rq - 1$ so that $B(T) \equiv_r M_{a,b,\alpha,\beta}(T)$, and further that all of the coefficients

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

$[T^c]B(T)$ for $\deg(M_{a,b,\alpha,\beta}) - (qr-r) < c < qr$ are equal to the corresponding coefficient of $B(T)$. As $M_{a,b}$ is type r, we have $\lfloor a/q \rfloor + \lfloor b/q \rfloor < r$, so the degree of the polynomial $M_{a,b,\alpha,\beta}$ is at most $a+b \le (r+1)q-2$, and

$$((r+1)q-2) - (qr-r) = r+q-2 < qr-s$$

for any allowed choice of q, r, s, so $[T^c]B(T) = [T^c]$ $M_{a,b,\alpha,\beta}(T)$ for all c so that

$$qr - s \le c \le qr.$$

Thus, to show that $B(T)$ has degree less than $qr-s$, it suffices to show that the coefficients of $T^{qr-s}, T^{qr-s+1}, \ldots, T^{qr-1}$ in $M_{a,b,\alpha,\beta}$ are all zero.

Write $a = a_0 q + a_1$ and $b = b_0 q + b_1$ where $a_0 + b_0 \le r-1$ and $0 \le a_1, b_1 \le q-1$. Note that if $a_0 + b_0 < r-1$, then for $s' = 1, \ldots, s$ coefficient $[T^{rq-s'}]M_{a,b,\alpha,\beta}$ is always zero except possibly when $a_0 + b_0 = r-2$ and $a_1 + b_1 \ge 2q - s$. This can happen for at most $\frac{rs^2}{2}$ pairs (a,b). Hence, for $a_0 + b_0 < r-1$, there are $\le \frac{rs^2}{2}$ bad monomials (a,b).

Now assume $a_0 + b_0 = r-1$. For $s' = 1, \ldots, s$, the coefficient of $T^{rq-s'}$ in $T^a(\alpha T + \beta)^b$ is 0 if $rq - s' < a$ or $a + b < rq - s'$. Otherwise, the coefficient is

$$\alpha^{rq-s'-a}\beta^{b-rq+s'+a}\binom{b}{rq-s'-a} = \alpha^{rq-s'-a}\beta^{b-rq+s'+a}\binom{b_0q+b_1}{b_0q+q-s'-a_1}. \quad (9)$$

By Proposition II.1, the binomial coefficient is nonzero (mod 2) if and only if $b_0 q + q - s' - a_1 \le_2 b_0 q + b_1$, which, as q is a power of 2, happens only if $q - s' - a_1 \le_2^\ell b_1$. Hence, if $a_0 + b_0 = r-1$, the monomial $M_{a,b}$ is $(r, rq-s)$-bad only if some $s' = 1, \ldots, s$ satisfies $q - s' - a_1 \le_2^\ell b_1$. Hence, by Lemma IV.1, for a fixed a_0, b_0 with $a_0 + b_0 = r-1$, there are at most $2s^{\log_2(4/3)}3^\ell$ bad monomials $M_{a,b}$, so there are at most $r \cdot s^{\log_2(4/3)}3^\ell$ bad monomials $M_{a,b}$ over all a_0, b_0 with $a_0 + b_0 = r-1$. As we showed, there are at most $\frac{rs^2}{2}$ bad monomials when $a_0 + b_0 < r-1$. Hence, there are at least $\binom{r+1}{2}4^\ell - 2rs^{\log_2(4/3)}3^\ell - \frac{rs^2}{2} \ge \binom{r+1}{2}q^2 - 3rs^{\log_2(4/3)}q^{\log_2(3)}$ good monomials, as desired. □

Lemma IV.2 and Lemma III.8 together imply Corollary IV.3, which in turn implies the informal result stated at the beginning of the section.

Corollary IV.3: Let $r = 2^{\ell_r}$, $s = 2^{\ell_s}$ and $q = 2^\ell$ with $\ell_r, \ell_s \in \{1, \ldots, \ell-1\}$. A $(q, r, rq-s)$ lifted multiplicity code has rate at least $1 - 6r^{-1}s^{\log_2(4/3)}q^{\log_2(3/4)}$.

Remark IV.4: We apply Corollary IV.3 for $r = s \le q$, giving that a lifted multiplicity code of rate at least $1 - 6r^{\log_2(2/3)}q^{\log_2(3/4)}$. By comparison [14], a 2-variate multiplicity code of order r evaluations of degree at most $rq-r$ polynomials over $\mathbb{F}_q$ has rate $\frac{\binom{rq-r+2}{2}}{\binom{r+1}{2}q^2} \le 1 - \Omega(\frac{1}{r})$, which is smaller than the rate of lifted multiplicity codes for $r \ll q$.

Remark IV.5 (Quantitative Comparison to [25]): The work [25] also studies lifted multiplicity codes, but focuses on a different parameter regime than we focus on here (where t is large, rather than $t \le \sqrt{N}$). Perhaps because they focus on a different parameter regime, the approach of [25] does not yield any nontrivial results in our parameter regime, and consequently our analysis of lifted multiplicity codes is much stronger.

For example, for degree $d = rq - r$ codes, [25] bounds [5] the rate of the code below by $1 - \Theta(\frac{r}{q})^{1/(32\ln 2)}$. This is a weaker bound than the straightforward bound of $1 - q^{\log_2(3/4)}$ sketched at the beginning of this section, and significantly weaker than our bound in Corollary IV.3 of $1 - 6r^{\log_2(2/3)}q^{\log_2(3/4)}$ for all r and q. Moreover, for $r \gg q^{1/23}$, the bound of [25] is even weaker than the lower bound on the rate of (non-lifted) multiplicity codes, which is $1 - \Omega(1/r)$.

Remark IV.6 (The Value of Bivariate Lifts): In addition to likely giving better bounds than m-variate lifts (see Why only bivariate lifts? in Section I-A), another reason that we study only bivariate lifts in this paper is that it makes the computations much more tractable. In the proof of Lemma IV.2, we study $M_{a,b,\alpha,\beta}(T) = (T^a)(\alpha T + \beta)^b$, and expand out the terms to apply Lucas's theorem. If we were to consider, say, trivariate lifts, we would have to expand expressions of the form $(T^a)(\alpha T + \beta)^b(\gamma T + \delta)^c$, and it would become more complicated to keep track of the coefficients on various powers. Analyzing m-variate lifts would become more complicated still. In particular, it seems harder to get as tight a bound on the codimension of the code for m-variate lifts for $m > 2$ as we are able to get for $m = 2$. Given that we are already able to obtain good codes for bivariate lifts, we restrict our attention to this simpler case.

V. Disjoint Repair Groups of Lifted Multiplicity Codes

Finally, we prove Theorem I.2, which we repeat below.

Theorem (Theorem I.2, Restated): Let $r = 2^{\ell_r}$ and $q = 2^\ell$ with $\ell_r < \ell$ and $\mathcal{C}$ be the $(q, r, rq-r)$ lifted multiplicity code.

- The length of the code is q^2.
- The rate of the code is at least $1 - 6r^{\log_2(2/3)}q^{\log_2(3/4)}$.
- The code has the q/r-disjoint repair group property.

Proof: The first item follows from the definition of $\mathcal{C}$, and the second item is by Corollary IV.3. To see the third item, we observe that, given a point $(\gamma, \delta) \in \mathbb{F}_q^2$, lines $L_1, \ldots, L_r$ passing through (γ, δ), and $P^{(<r)}(\mathbf{y})$ at all points $\mathbf{y}$ on the lines $L_1, \ldots, L_r$ except (γ, δ) itself, we can (efficiently) recover $P^{(<r)}(\gamma, \delta)$. This guarantees the q/r-disjoint repair group property, because we can group the q lines of $\mathcal{L}$ of the form $L(T) = (T, \alpha T + \beta)$ passing through (γ, δ) arbitrarily into groups of r, giving q/r disjoint repair groups. For any line L_k, the polynomial $P_{L_k}(T)$ has degree at most $rq - r - 1$, as P is $(q, r, qr-r)$-good. By taking linear combinations of directional derivatives (Lemma II.5), we can efficiently compute $P_{L_k}^{(i)}(\gamma')$ for every $i = 0, \ldots, r-1$, every $k = 1, \ldots, r$, and every $\gamma' \ne \gamma$. We can compute $P_{L_k}(T)$ using a generalization of polynomial interpolation. This can be done in $O(D \log D)$ time, where $D < rq$ is the degree of

[5] The details are as follows: using some notation from [25], for a rate $1-\alpha$ code in our parameter regime, $n = 2$ variables, a prime $p = 2$, a parameter $b = \lceil \log_p n \rceil + 1 = 2$, $q = p^\ell$, $r = p^{\ell_r}$ (they use m for r, s for ℓ, and t for ℓ_r), $\alpha_t = \frac{\alpha}{1 - \frac{N_n(p^{\ell_r}-2)}{N_n(p^{\ell_r})}} = \frac{\alpha}{1 - \frac{N_2(r/4)}{N_2(r)}} \approx \frac{\alpha}{1-1/16} = \Theta(\alpha)$ (here $N_2(r) := \binom{r+1}{2}$, and we assume $r \gg 1$), $c = bp^{bn}\ln\frac{1}{\alpha_t} + \ell_r = 32\ln\frac{1}{\alpha_t} + \ell_r$, $d = (1-p^{-c})rq$. In our setting, we choose $d = rq - r$, which requires $p^c = q$, so $c = \ell$, so $\ell - \ell_r = 32\ln\frac{1}{\alpha_t} = 32\ln 2 \cdot \log\frac{1}{\alpha} - O(1)$. Thus, $\alpha = \Theta(2^{(\ell_r - \ell)/(32\ln 2)}) = (\frac{r}{q})^{1/(32\ln 2)}$.

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

the polynomial (see e.g. [3]) Hence, by Corollary II.6, from $P_{L_1}(T), \ldots, P_{L_k}(T)$, we can efficiently compute $P^{(i,j)}(\gamma, \delta)$ for all i, j with $0 \leq i + j \leq r - 1$. □

VI. Conclusion

We conclude with some open questions.

1) We have shown that lifted multiplicity codes with redundancy $O(t^{0.585}\sqrt{N})$ have the t-DRGP for a range of $t \leq \sqrt{N}$. However, we do not know of any general lower bounds when $t \in (1, \sqrt{N})$ beyond the lower bound for $t = 2$, which implies that the redundancy must be at least $\Omega(\sqrt{N})$ for any t. When $t \geq \sqrt{N}$, there is a stronger redundancy lower bound of $\Omega(t)$, which holds simply because a code with the t-DRGP must have Hamming distance at least t. Thus, it is an open question whether or not our bound is tight or whether one can do better.
2) Lifted multiplicity codes display better locality for the t-DRGP problem for $t \leq \sqrt{N}$; it is a natural question to ask whether they can be used for larger t, and in particular whether they could lead to improved constructions of locally correctable codes. In particular, it would be interesting if lifted multiplicity codes could qualitatively out-perform (un-lifted) multiplicity codes as high-rate LCCs, for example by maintaining the high rate while achieving sub-polynomial query complexity.[6] We note that for the LCC problem, one typically does not care about pinning down the rate, so long as it is close to 1, instead focusing on the query complexity. In contrast, in this work, we have focused on pinning down the rate much more precisely.
3) Related to the above, it would be natural to understand the rate and locality of lifted multiplicity codes over more than two variables.
4) The alphabet size of lifted multiplicity codes is $q^{\binom{r+1}{2}}$, which, if the multiplicity is $r = q^\alpha$ for a constant $\alpha > 0$, is exponential type in the code length q^2. In practical applications, a smaller alphabet size is desirable. It would be interesting to achieve the results of Corollary I.3 with a code whose length grows independently of the alphabet size.
5) In this paper, we studied the t-DRGP locality property, which requires that each symbol has many disjoint repair groups. Another common notion of locality is an Locally Recoverable Code (LRC) with locality d, which requires that each symbol has one repair group of size at most d. These two notions are combined in the notion of an LRC with locality d and availability t (see, e.g. [17], [21], [26]), combines these two notions. This requires that each symbol have t disjoint repair groups, each of size at most d. The techniques in this paper yield codes with locality rq and availability q/r, where r is the multiplicity. It would be interesting to construct codes with a better trade-off between locality and availability, possibly using lifting and/or multiplicity techniques.

[6] As noted in the introduction, the work [25] showed the lifted multiplicity codes are good LCCs with lower-order derivatives than were required by the (un-lifted) multiplicity codes of [14], but it does not show how to improve the query complexity to sub-polynomial.

Appendix

Proof of Proposition II.4: By part 2 of Proposition II.3,

$$P^{(i)}(X) = \sum_{j_1 + \cdots + j_r = i} \prod_{k=1}^{r} D^{(j_k)}(X^q - X). \tag{10}$$

We have $D^{(1)}(X^q - X) = 1$ (the field has characteristic 2). For $2 \leq i < q$, the ith derivative of $X^q - X$ is $\binom{q}{i} X^{q-i}$, which is 0, as $\binom{q}{i}$ is even by Proposition II.1. The summand above is nonzero if and only if $j_1, j_2, \ldots, j_r \leq 1$. When $i \leq r$, this happens when i of the j_k's are 1 and $r - i$ are 0, which happens for $\binom{r}{i}$ choices of $(j_1, \ldots, j_r)$. This gives $P^{(i)}(X) = \binom{r}{i}(X^q - X)^{r-i}$ for $0 \leq i \leq r$. When $i > r$, some j_k is at least 2, in which case $P^{(r)}(X) = 0$ for $r < i < q$. □

Proof of Lemma II.5: Let $\mathbf{a}_k$ denote the vector $(1, \alpha_k)$, and let $\mathbf{b}_k$ denote the vector $(0, \beta_k)$. By assumption, we have that $\mathbf{a}_k \gamma + \mathbf{b}_k = (\gamma, \delta)$. By the definition of Hasse derivatives, we have, for all $k = 1, \ldots, r$

$$\begin{aligned} P_{L_k}(T + Z) &= P(\mathbf{a}_k T + \mathbf{b}_k + \mathbf{a}_k Z) \\ &= \sum_{\mathbf{i} \in \mathbb{N}^2} P^{(\mathbf{i})}(\mathbf{a}_k T + \mathbf{b}_k) \cdot (\mathbf{a}_k Z)^{\mathbf{i}} \\ &= \sum_{\mathbf{i} \in \mathbb{N}^2} P^{(\mathbf{i})}(\mathbf{a}_k T + \mathbf{b}_k) \cdot \mathbf{a}_k^{\mathbf{i}} Z^{\mathrm{wt}(\mathbf{i})} \\ P_{L_k}(T + Z) &= \sum_{i \geq 0} P_{L_k}^{(i)}(T) Z^i \end{aligned} \tag{11}$$

Hence, for all $i \geq 0$ and $k = 1, \ldots, r$, we have

$$P_{L_k}^{(i)}(T) = \sum_{\mathbf{i}: \mathrm{wt}(\mathbf{i}) = i} P^{(\mathbf{i})}(\mathbf{a}_k T + \mathbf{b}_k) \mathbf{a}_k^{\mathbf{i}} \tag{12}$$

By plugging in $T = \gamma$, we have for all $i \geq 0$ and $k = 1, \ldots, r$,

$$P_{L_k}^{(i)}(\gamma) = \sum_{\mathbf{i}: \mathrm{wt}(\mathbf{i}) = i} P^{(\mathbf{i})}(\gamma, \delta) \mathbf{a}_k^{\mathbf{i}}. \tag{13}$$

Rewriting this in matrix form gives the desired result. □

It was shown in [6] that bivariate lifted parity-check codes over $\mathbb{F}_q$, where $q = 2^\ell$, have co-dimension 3^ℓ. Here, we give an alternative proof using dual codes. The techniques in this proof are not directly related to the techniques that we used in the main body of the paper, but we found this alternative proof illuminating so we include it.

Let $q = 2^\ell$. Recall $\mathcal{L}$ is the set of lines expressible as $L(T) = (T, \alpha T + \beta)$ where $\alpha, \beta \in \mathbb{F}_q$. One way to think about codes with locality is by considering their dual code. If the code is a subset of $\mathbb{F}_q^{q \times q}$, then the dual code corresponds to lines of repair groups. Given a line $L(T)$ in $\mathcal{L}$, define the corresponding dual codeword:

$$(c_L^\perp)_{ij} \coloneqq \begin{cases} 1 & (i,j) = L(t) \text{ for some } t \in \mathbb{F}_q \\ 0 & \text{o/w} \end{cases} \tag{14}$$

Let

$$V_{\mathcal{L}} \coloneqq \operatorname{span}\left\{ c_L^\perp : L \in \mathcal{L} \right\}. \tag{15}$$

Note that $V_{\mathcal{L}}$ is spanned by 4^ℓ elements, so the trivial bound on the dimension is 4^ℓ. We give the following improved bound, matching the analysis of [6].

Lemma A.1: The subspace $V_{\mathcal{L}}$ has dimension at most 3^ℓ.

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.

Proof: A codeword $c_L^{\perp}$ is the evaluation of the following polynomial on $\mathbb{F}_q^{q\times q}$:

$$P_L(X,Y) \;:=\; \prod_{\beta\neq\beta_L} (\alpha_L X + \beta - Y). \tag{16}$$

If $(X,Y) \notin L$, then the polynomial evaluates to 0 as $Y - \alpha_L X \neq \beta_L$, and otherwise it evaluates to

$$\prod_{\beta\neq\beta_L} (\beta - \beta_L) = \prod_{\beta\in\mathbb{F}_q^*} \beta = 1. \tag{17}$$

For $a+b \geq q$, the coefficient of X^aY^b in $P_L(X,Y)$ is 0. For $a+b \leq q$, the coefficient of X^aY^b in $P_L(X,Y)$ is

$$\binom{a+b}{a}\alpha_L^a(-1)^b \sum_{\substack{\beta_1,\ldots,\beta_{q-1-a-b}\in\mathbb{F}_q \\ \text{distinct},\neq\beta_L}} \prod_{j=1}^{q-1-a-b} \beta_j. \tag{18}$$

This is because we first chose $a+b$ terms that contain X or Y, then choose which terms are X and which terms are Y, and this gives us a many α_L's and b many -1's, and we sum over the choices of the β terms that we choose. Hence, the only a,b such that $[X^aY^b]P_L(X,Y) \neq 0$ for any L are the pairs (a,b) such that $a+b \leq q-1$ and $\binom{a+b}{a} \equiv 1 \mod 2$. There are at most 3^ℓ pairs by Proposition II.1. It follows that the polynomials $P_L(X,Y)$ are spanned by 3^ℓ monomials X^aY^b with $\binom{a+b}{a} \equiv 1 \mod 2$. Hence, the vector space $V_{\mathcal{L}}$ is spanned by 3^ℓ dual codewords in $\mathbb{F}_q^{q\times q}$ and thus has dimension at most 3^ℓ. □

ACKNOWLEDGMENT

The authors would like to thank Eitan Yaakobi for helpful conversations. They would also like to thank Julien Lavauzelle for pointing out Ref. [25], for pointing out an error in an earlier version of this article and for suggesting Open Question 4. They also thank Nikita Polianskii for pointing out an error in an earlier version of this article. A previous version claimed that a lifted code is exactly the span of all good monomials, but in fact the span of all good monomials only forms a subset of the lifted code (see Remark III.9). This does not change their main result, as their lower bound on the number of good monomials still gives the same lower bound on the rate of the lifted code. They finally thank anonymous reviewers for helpful comments on an earlier draft of this article.

REFERENCES

[1] H. Asi and E. Yaakobi, "Nearly optimal constructions of PIR and batch codes," *IEEE Trans. Inf. Theory*, vol. 65, no. 2, pp. 947–964, Feb. 2019.
[2] S. R. Blackburn and T. Etzion, "PIR array codes with optimal PIR rates," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 2658–2662.
[3] 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.
[4] S. L. Frank-Fischer, V. Guruswami, and M. Wootters, "Locality via partially lifted codes," in *Proc. Approximation, Randomization, Combinat. Optim., Algorithms Techn. (APPROX/RANDOM)*, vol. 81. Wadern, Germany: Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017, pp. 43:1–43:17.
[5] A. Fazeli, A. Vardy, and E. Yaakobi, "Codes for distributed PIR with low storage overhead," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2015, pp. 2852–2856.
[6] A. Guo, S. Kopparty, and M. Sudan, "New affine-invariant codes from lifting," in *Proc. 4th Conf. Innov. Theor. Comput. Sci. (ITCS)*, Berkeley, CA, USA, Jan. 2013, pp. 529–540.
[7] V. Guruswami and C. Wang, "Linear-algebraic list decoding for variants of Reed–Solomon codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 6, pp. 3257–3268, Jun. 2013.
[8] W. P. J. Hirschfeld, G. Korchmáros, and F. Torres, *Algebraic Curves Over a Finite Field* (Princeton Series in Applied Mathematics). Princeton, NJ, USA: Princeton Univ. Press, 2008.
[9] B. Hemenway, R. Ostrovsky, and M. Wootters, "Local correctability of expander codes," *Inf. Comput.*, vol. 243, pp. 178–190, Aug. 2015.
[10] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai, "Batch codes and their applications," in *Proc. 26th Annu. ACM Symp. Theory Comput. (STOC)*, 2004, pp. 262–271.
[11] S. Kopparty, O. Meir, N. Ron-Zewi, and S. Saraf, "High-rate locally-correctable and locally-testable codes with sub-polynomial query complexity," in *Proc. 48th Annu. ACM SIGACT Symp. Theory Comput. (STOC)*, 2016, pp. 202–215.
[12] S. Kopparty, "List-decoding multiplicity codes," *Theory Comput.*, vol. 11, no. 1, pp. 149–182, 2015.
[13] S. Kopparty, "Some remarks on multiplicity codes," *Contemp. Math.*, vol. 625, pp. 155–176, 2014.
[14] S. Kopparty, S. Saraf, and S. Yekhanin, "High-rate codes with sublinear-time decoding," *J. ACM*, vol. 61, no. 5, pp. 1–20, Sep. 2014.
[15] 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.
[16] S. Lin and D. J. Costello, *Error Control Coding*. London, U.K.: Pearson Education India, 2001.
[17] A. S. Rawat, D. S. Papailiopoulos, A. G. Dimakis, and S. Vishwanath, "Locality and availability in distributed storage," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2014, pp. 681–685.
[18] A. S. Rawat, Z. Song, A. G. Dimakis, and A. Gal, "Batch codes through dense graphs without short cycles," *IEEE Trans. Inf. Theory*, vol. 62, no. 4, pp. 1592–1604, Apr. 2016.
[19] S. Rao and A. Vardy, "Lower bound on the redundancy of PIR codes," 2016, *arXiv:1605.01869*. [Online]. Available: http://arxiv.org/abs/1605.01869
[20] V. Skachek, "Batch and PIR codes and their connections to locally repairable codes," in *Network Coding and Subspace Designs*. Cham, Switzerland: Springer, 2018, pp. 427–442.
[21] I. Tamo and A. Barg, "Bounds on locally recoverable codes with multiple recovering sets," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2014, pp. 691–695.
[22] I. Tamo, A. Barg, and A. Frolov, "Bounds on the parameters of locally recoverable codes," *IEEE Trans. Inf. Theory*, vol. 62, no. 6, pp. 3070–3083, Jun. 2016.
[23] D. P. Woodruff, "A quadratic lower bound for three-query linear locally decodable codes over any field," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*. Berlin, Germany: Springer, 2010, pp. 766–779.
[24] M. Wootters. (2016). *Linear Codes With Disjoint Repair Groups*. [Online]. Available: https://sites.google.com/site/marywootters/disjoint_repair_groups.pdf
[25] L. Wu, "Revisiting the multiplicity codes: A new class of high-rate locally correctable codes," in *Proc. 53rd Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2015, pp. 509–513.
[26] A. Wang and Z. Zhang, "Repair locality with multiple erasure tolerance," *IEEE Trans. Inf. Theory*, vol. 60, no. 11, pp. 6979–6987, Nov. 2014.

Ray Li (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees in mathematical sciences from Carnegie Mellon University in 2017. He is currently pursuing the Ph.D. degree with Stanford University. His research interests include coding theory, combinatorics, and fine-grained complexity.

Mary Wootters (Member, IEEE) received the B.A. degree in mathematics and computer science from Swarthmore College in 2008, and the Ph.D. degree in mathematics from the University of Michigan in 2014. She is currently an Assistant Professor of computer science and electrical engineering with Stanford University. She was an NSF Postdoctoral Fellow at Carnegie Mellon University from 2014 to 2016. She works in theoretical computer science, applied mathematics, and information theory. Her research interests include error correcting codes and randomized algorithms for dealing with high-dimensional data. She was a recipient of the NSF CAREER Award and was named a Sloan Research Fellow in 2019.

Authorized licensed use limited to: Stanford University. Downloaded on September 16,2021 at 20:56:35 UTC from IEEE Xplore. Restrictions apply.