

Improved List-Decodability and List-Recoverability of Reed–Solomon Codes via Tree Packings ^{*}

Zeyu Guo[†] Ray Li[‡] Chong Shangguan[§] Itzhak Tamo[¶] Mary Wootters^{||}

Abstract

This paper shows that there exist Reed–Solomon (RS) codes, over large finite fields, that are combinatorially list-decodable well beyond the Johnson radius, in fact almost achieving list-decoding capacity. In particular, we show that for any $\varepsilon \in (0, 1]$ there exist RS codes with rate $\Omega\left(\frac{\varepsilon}{\log(1/\varepsilon)+1}\right)$ that are list-decodable from radius of $1 - \varepsilon$. We generalize this result to list-recovery, showing that there exist $(1 - \varepsilon, \ell, O(\ell/\varepsilon))$ -list-recoverable RS codes with rate $\Omega\left(\frac{\varepsilon}{\sqrt{\ell}(\log(1/\varepsilon)+1)}\right)$. Along the way we use our techniques to give a new proof of a result of Blackburn on optimal linear perfect hash matrices, and strengthen it to obtain a construction of strongly perfect hash matrices.

To derive the results in this paper we show a surprising connection of the above problems to graph theory, and in particular to the tree packing theorem of Nash-Williams and Tutte. We also state a new conjecture that generalizes the tree-packing theorem to hypergraphs, and show that if this conjecture holds, then there would exist RS codes that are *optimally* (non-asymptotically) list-decodable.

^{*}Keywords: Reed–Solomon codes, Nash-Williams–Tutte Theorem, Johnson radius, list decoding, list recovery, perfect hash matrix; MSC2010 classification: 05C05, 05D05, 11T71

[†]Department of Computer Science, University of Haifa, Haifa 3498838, Israel. Email: zguotcs@gmail.com

[‡]Department of Computer Science, Stanford University. Email: rayyli@cs.stanford.edu

[§]Research Center for Mathematics and Interdisciplinary Sciences, Shandong University, Qingdao 266237, China. Email: theoreming@163.com. Part of the work was done while the author was a postdoc at Tel Aviv University and was supported by the Israel Science Foundation (ISF grant number 1030/15).

[¶]Department of Electrical Engineering - Systems, Tel Aviv University, Tel Aviv 6997801, Israel. Email: zactamo@gmail.com.

^{||}Departments of Computer Science and Electrical Engineering, Stanford University. Email: marykw@stanford.edu

Contents

1	Introduction	1
1.1	Contributions	3
1.2	Related Work	5
1.3	Technical Overview	7
1.4	Future Directions and Open Questions	11
2	Preliminaries	12
2.1	Cycle Spaces	12
2.2	t -Wise Intersection Matrices	14
3	Connection to Tree Packing and an Intermediate Result	15
4	Application to Perfect Hashing	19
4.1	Proof of Theorem 1.3	19
4.2	Proof of Proposition 4.1	21
5	Near-Optimal List-Recovery of RS Codes: Proof of the Main Theorem	22
5.1	Overview of the Proof	22
5.2	A Combinatorial Lemma	23
5.3	Proof of Theorem 5.1	24
5.4	Proof of Lemma 5.3	26
6	Towards Conjecture 1.4: A Hypergraph Nash-Williams–Tutte Conjecture	30
6.1	A Hypergraph Nash-Williams–Tutte Conjecture	31
6.2	Proof of Theorem 6.2	33
A	Examples of Intersection Matrices	38

1 Introduction

Reed–Solomon (RS) codes are a classical family of error correcting codes, ubiquitous in both theory and practice. To define an RS code, let \mathbb{F}_q be the finite field of size q , and let $1 \leq k < n \leq q$. Fix n distinct evaluation points $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbb{F}_q$. The $[n, k]$ -Reed–Solomon code over \mathbb{F}_q with evaluation points $(\alpha_1, \dots, \alpha_n)$ is defined as the set

$$\{(f(\alpha_1), \dots, f(\alpha_n)) : f \in \mathbb{F}_q[x], \deg(f) < k\}.$$

RS codes attain the optimal trade-off between *rate* and *distance*. The rate of a code $\mathcal{C} \subset \mathbb{F}_q^n$ is defined as $R = \log_q |\mathcal{C}|/n$. The rate is a number between 0 and 1, and the closer to 1 the better. The (relative) distance of a code $\mathcal{C} \subset \mathbb{F}_q^n$ is defined to be $\delta(\mathcal{C}) = \min_{c \neq c' \in \mathcal{C}} d(c, c')$, where $d(c, c') = |\{i \in [n] : c_i \neq c'_i\}|/n$ is relative Hamming distance. Again, the relative distance is a number between 0 and 1, and the closer to 1 the better. An $[n, k]$ -RS code has rate k/n and distance $(n - k + 1)/n$, which is the best-possible trade-off, according to the Singleton bound [Sin64].

Because RS codes attain this optimal trade-off (and also because they admit efficient algorithms), they have been well-studied since their introduction in the 1960’s [RS60]. However, perhaps surprisingly, there is still much about them that we do not know. One notable example is their (combinatorial)¹ *list-decodability* and more generally their *list-recoverability*. We discuss list-decodability first, and discuss list-recoverability after that.

List-Decodability of RS Codes. List-decodability can be seen as a generalization of distance. For $\rho \in (0, 1)$ and $L \geq 1$, we say that a code $\mathcal{C} \subset \mathbb{F}_q^n$ is (ρ, L) -list-decodable if for any $y \in \mathbb{F}_q^n$,

$$|\{c \in \mathcal{C} : d(c, y) \leq \rho\}| \leq L.$$

In particular, $(\rho, 1)$ -list-decodability is the same as having distance greater than 2ρ . List-decodability was introduced by Elias and Wozencraft in the 1950’s [Eli57, Woz58]. By now it is an important primitive in both coding theory and theoretical computer science more broadly. In general, larger *list sizes* (the parameter L) allow for a larger *list-decoding radius* (the parameter ρ). In this work, we will be interested in the case when $\rho = 1 - \varepsilon$ is large.

The list-decodability of Reed–Solomon codes is of interest for several reasons. First, both list-decodability and Reed–Solomon codes are central notions in coding theory, and the authors believe that question is interesting in its own right. Moreover, the list-decodability of Reed–Solomon codes has found applications in complexity theory and pseudorandomness [CPS99, STV01, LP20].

Until recently, the best bounds available on the list-decodability of RS codes were bounds that hold generically for any code. The *Johnson bound* states that any code with minimum relative distance δ is $(1 - \sqrt{1 - \delta}, qn^2\delta)$ -list-decodable over an alphabet of size q ([Joh62], see also [GRS19, Theorem 7.3.3]). This implies that, for any $\varepsilon \in (0, 1)$, there are RS codes that are list-decodable up to radius $1 - \varepsilon$ (with polynomial list sizes) that have rate $\Omega(\varepsilon^2)$. The celebrated Guruswami–Sudan algorithm [GS99] gives an efficient algorithm to list-decode RS codes up to the Johnson bound, but it breaks down at this point. Meanwhile, the *list-decoding capacity theorem* implies that no code (and in particular, no RS code) that is list-decodable up to radius $1 - \varepsilon$ can have rate bounded above ε , unless the list sizes are exponential.

There have been several works over the past decade aimed at closing the gap between the Johnson bound (rate $\Theta(\varepsilon^2)$) and the list-decoding capacity theorem (rate $\Theta(\varepsilon)$). On the negative side, it is known that *some* RS codes (that is, some way of choosing the evaluation points $\alpha_1, \dots, \alpha_n$), are

¹Throughout this paper, we will study *combinatorial* (rather than *algorithmic*) list-decodability.

not list-decodable substantially beyond the Johnson bound [BKR10]. On the positive side, Rudra and Wootters [RW14] showed that a random choice of evaluation points will, with high probability, yield a code that is list-decodable up to radius $1 - \varepsilon$ with rate $O\left(\frac{\varepsilon}{\log^5(1/\varepsilon)\log q}\right)$. Unfortunately, while the dependence on ε in the rate is nearly optimal (the “correct” dependence should be linear in ε , according to the list-decoding capacity theorem), the $\log q$ term in the denominator means that the rate necessarily goes to zero as n grows, as we must have $q \geq n$ for RS codes. Working in a different parameter regime, Shangguan and Tamo showed that over a large alphabet, there exist RS codes of rate larger than $1/9$ that can also be list-decoded beyond the Johnson bound (and in fact, optimally) [ST20a]. However, this result only holds for small list sizes ($L = 2, 3$), and in particular, for such small list sizes one cannot hope to list-decode up to a radius $1 - \varepsilon$ that approaches 1. Thus, there was still a substantial gap between capacity and the best known trade-offs for list-decoding RS codes.

List-Recoverability of RS Codes. The gap between capacity and the best known trade-offs for RS codes is even more pronounced for *list recovery*, a generalization of list decoding. We say that a code $\mathcal{C} \subset \mathbb{F}_q^n$ is (ρ, ℓ, L) -list-recoverable if for any $S_1, S_2, \dots, S_n \subset \mathbb{F}_q$ with $|S_i| = \ell$,

$$|\{c \in \mathcal{C} : d(c, S_1 \times S_2 \times \dots \times S_n) \leq \rho\}| \leq L.$$

Here, we extend the definition of Hamming distance to sets by denoting

$$d(c, S_1 \times \dots \times S_n) = \frac{1}{n} |\{i \in [n] : c_i \notin S_i\}|.$$

The parameter ℓ is called the *input list size*. List-decoding is the special case of list-recovery for $\ell = 1$. List-recovery first arose in the context of list-decoding (for example, the Guruswami–Sudan algorithm mentioned above is in fact a list-recovery algorithm), but has since found applications beyond that, for example in pseudorandomness [GUV09] and algorithm design [DW20].

Both the Johnson bound and the list-decoding capacity theorem have analogs for list-recovery. The list-recovery Johnson bound [GS01] implies that there are RS codes of rate $\Omega(\varepsilon^2/\ell)$ that are list-recoverable up to radius $1 - \varepsilon$ with input list size ℓ and polynomial output list size. However, the list-recovery capacity theorem implies that there are codes of rate $\Omega(\varepsilon)$ (with *no* dependence on ℓ) that achieve the same guarantee, provided that the alphabet size q is sufficiently large.

Thus the gap for list-recovery (between rate $\Theta(\varepsilon^2/\ell)$ and $\Theta(\varepsilon)$) is even larger than that for list-decoding, and in particular the dependence on ℓ becomes important. To the best of our knowledge, before our work there were *no* results known for RS codes that established list-recovery up to arbitrarily large radius $1 - \varepsilon$ with a better dependence on ℓ than $1/\ell$.

Motivating question. Given this state of affairs, our motivating question is whether or not RS codes can be list-decoded or list-recovered up to radius $1 - \varepsilon$ with rates $\Omega(\varepsilon)$ (in particular, with a linear dependence on ε and no dependence on the alphabet size q or the input list size ℓ). As outlined below, we nearly resolve this question for list-decoding and make substantial progress for list-recovery.

Subsequent work. After this paper first appeared, and inspired by the techniques in this paper and in [ST20a], Ferber, Kwan, and Saueremann showed that there exist $(1 - \varepsilon, O(1/\varepsilon))$ -list-decodable RS codes with rate $\Omega(\varepsilon)$ over a field size polynomial in the block length, improving our result for list-decoding [FKS20]. In a very recent work, Goldberg, Shangguan, and Tamo further improved the rate of [FKS20] by showing the existence of $(1 - \varepsilon, O(1/\varepsilon))$ -list-decodable RS codes with rate approaching $\frac{\varepsilon}{2 - \varepsilon}$ [GST21]. See Section 1.2 for more details.

1.1 Contributions

Our main result establishes the list-recoverability (and in particular, the list-decodability), of Reed–Solomon codes up to radius $1 - \varepsilon$, representing a significant improvement over previous work. Our techniques build on the approach of [ST20a]; the main new technical contribution is a novel connection between list-decoding RS codes and the Nash-Williams–Tutte theorem in graph theory, which may be of independent interest. We outline our contributions below.

Existence of RS codes that are near-optimally list-decodable. Our main theorem for list-decoding is as follows.

Theorem 1.1 (RS codes with near-optimal list-decoding). *There is a constant $c \geq 1$ so that the following statement holds. For any $\varepsilon \in (0, 1]$ and any sufficiently large n , there exist RS codes of rate $R \geq \frac{\varepsilon}{c(\log(1/\varepsilon)+1)}$ over a large enough finite field (as a function of n and ε), that are $(1 - \varepsilon, c/\varepsilon)$ -list-decodable.*

As discussed above, Theorem 1.1 is stronger than the result of Rudra and Wootters [RW14], in that the result of [RW14] requires that the rate tend to zero as n grows, while ours holds for constant-rate codes. On the other hand, our result requires the field size q to be quite large (see Table 1), which [RW14] did not require.

Our result also differs from the result of Shangguan and Tamo [ST20a] discussed above. Because that work focuses on small list sizes, it does not apply to list-decoding radii approaching 1. In contrast, we are able to list-decode up to radius $1 - \varepsilon$. We note that [ST20a] is able to show that RS codes are exactly optimal, while we are off by logarithmic factors. Both our work and that of [ST20a] require large field sizes.

Generalization to list-recovery. Theorem 1.1 follows from a more general result about list-recovery. Our main result is the following (see Theorem 5.1 for a more detailed version).

Theorem 1.2 (RS codes with list-recovery beyond the Johnson bound). *There is a constant $c \geq 1$ such that the following statement holds. For any $\varepsilon \in (0, 1]$, any positive integer ℓ , and any sufficiently large n , there exist RS codes with rate $R \geq \frac{\varepsilon}{c\sqrt{\ell}(\log(1/\varepsilon)+1)}$ over a large enough (as a function of n , ε , and ℓ) finite field, that are $(1 - \varepsilon, \ell, c\ell/\varepsilon)$ -list-recoverable.*

Theorem 1.2 establishes list-recoverability for RS codes well beyond the Johnson bound, and in particular breaks the $1/\ell$ barrier. To the best of our knowledge, this is the first result to do so for radius arbitrarily close to 1, although we note that work of Lund and Potukuchi achieved a similar rate for small error radius [LP20]. We discuss related work below in Section 1.2 and summarize quantitative results in Table 1.

Applications to perfect hashing. Our techniques also have an application to the construction of *strongly perfect hash matrices*, as detailed below. Given a matrix and a set S of its columns, a row is said to *separate* S if, restricted to this row, these columns have distinct values. For a positive integer t , a matrix is said to be a *t -perfect hash matrix* if any set of t distinct columns of the matrix is separated by at least one row. Perfect hash matrices were introduced by Mehlhorn [Meh84] in 1984 for database management, and since then they have found various applications in cryptography [Bla03], circuit design [NW95], and the design of deterministic analogs of probabilistic algorithms [AN96].

Let $\text{PHF}(n, m, q, t)$ denote a q -ary t -perfect hash matrix with n rows and m columns. Given m, q, t , determining the minimal n such that there exists a $\text{PHF}(n, m, q, t)$ is one of the major open

	Radius ρ	List size L	Rate R	Field size q
List-Decoding:				
Capacity	$1 - \varepsilon$	-	$\leq \varepsilon$	-
Johnson bound	$1 - \varepsilon$	$\text{poly}(n)$	$C\varepsilon^2$	$q \geq n$
[RW14]	$1 - \varepsilon$	C/ε	$\frac{C\varepsilon}{\log^3(1/\varepsilon)\log(q)}$	$q \geq Cn \log^C(n/\varepsilon)/\varepsilon$
[ST20a]	$\frac{L}{L+1}(1 - R)$	$L = 2, 3$	R	$q = 2^{Cn}$
This work (Thm. 1.1)	$1 - \varepsilon$	C/ε	$\frac{C\varepsilon}{\log(1/\varepsilon)}$	$q = \left(\frac{1}{\varepsilon}\right)^{Cn}$
List-Recovery:				
Capacity	$1 - \varepsilon$	-	$\leq \varepsilon$	-
Johnson bound	$1 - \varepsilon$	$\text{poly}(n)$	$\frac{C\varepsilon^2}{\ell}$	$q \geq n$
[LP20]	$\rho \leq 1 - 1/\sqrt{2}$	$C\ell$	$\frac{C}{\sqrt{\ell \cdot \log q}}$	$q \geq Cn\sqrt{\ell} \cdot \log n$
This work (Thm. 1.2)	$1 - \varepsilon$	$\frac{C\ell}{\varepsilon}$	$\frac{C\varepsilon}{\sqrt{\ell \cdot \log(1/\varepsilon)}}$	$q = \left(\frac{\ell}{\varepsilon}\right)^{Cn}$

Table 1: Prior work on list-decoding and list-recovery of RS codes. Above, C refers to an absolute constant. The ‘‘Capacity’’ results refer to the list-decoding and list-recovery capacity theorems, respectively, and are impossibility results. Above, we assume that $q \geq n$ and that $n \rightarrow \infty$ is growing relative to $1/\varepsilon$ and ℓ , and that n is sufficiently large.

questions in this field, and has received considerable attention (see, e.g., [BW98, Bla00, SG16]). For any integers $t \geq 2$, $k \geq 2$, and sufficiently large prime power q , using tools from linear algebra Blackburn [BW98] constructed a PHF($k(t-1), q^k, q, t$), which remains the best-known construction for such parameters so far.

Constructing perfect hash matrices is related to list-recovery and list-decoding. Indeed, if the columns of our matrix are codewords, then the matrix is a t -perfect hash matrix if and only if the code is $(0, t-1, t-1)$ -list-recoverable. On the way to proving our main result on list-recovery, we prove a theorem (Theorem 3.1, which we will state later), that gives very precise bounds, but only in a restricted setting. While this setting is too restrictive to immediately yield results on list-recovery in general, it turns out to be enough to say something interesting about perfect t -hash matrices. In particular, we are able to recover Blackburn’s result, and extend it to a generalization of perfect hashing where every set of t columns needs to be separated not just by one row but by many rows.

Theorem 1.3. *Given integers $1 \leq k < n$ and $t \geq 3$, for a sufficiently large prime power q , there exists an $n \times q^k$ matrix, defined on the alphabet \mathbb{F}_q , such that any set of t columns is separated by at least $n - k(t-1) + 1$ rows.*

We call a matrix with the property given by Theorem 1.3 a *strongly t -perfect hash matrix*; this can be viewed as an ‘‘error-resilient’’ version of perfect hash matrices. Strongly perfect hash matrices were first introduced by the third and fourth authors of this paper for $t = 3$, with a slightly different definition [ST20b]. Indeed, Lemma 25 of [ST20b] implies the $t = 3$ case of Theorem 1.3, but it breaks down at that point. We overcome this barrier, and construct strongly t -perfect hash matrices for all integers $t \geq 3$. The main ingredient in our proof is a surprising connection from strongly perfect hashing to graph theory (see Section 4 for the details). Perfect hash matrices with

a similar property (i.e., every set of t columns needs to be separated by more than one row) have also been studied in [Dou19], but not to our knowledge in this parameter regime.

Theorem 1.3 recovers Blackburn’s result by taking $n = k(t - 1)$, and it establishes the new result for strongly perfect hash matrices. Based on a result of Blackburn [BW98], we also show (Proposition 4.1) that the hash matrix in Theorem 1.3 is optimal for strongly perfect hashing, among all *linear* hash matrices. Both Theorem 1.3 and Proposition 4.1 are proved in Section 4.

A new connection to the Nash-Williams–Tutte theorem, and a new hypergraph Nash-Williams–Tutte conjecture. In order to derive our results, we build on the framework of [ST20a]. That work developed a framework to view the list-decodability of Reed–Solomon codes in terms of the singularity of *intersection matrices* (which we define in Section 2). The main new technical contribution of our work is to connect the singularity of these matrices to tree-packings in particular graphs. This connection allows us to use the Nash-Williams–Tutte theorem from graph theory to obtain our results. The Nash-Williams–Tutte theorem gives sufficient conditions for the existence of a large *tree packing* (that is, a collection of pairwise edge-disjoint spanning trees) in a graph.

We think that this connection is a contribution in its own right, and it is our hope that it will lead to further improvements to our results on Reed–Solomon codes. In particular, we hope that it will help establish the following conjecture of [ST20a]:

Conjecture 1.4 (Conjecture 1.5 of [ST20a]). *For any $\varepsilon > 0$ and integers $1 \leq k < n$ with $\varepsilon n \in \mathbb{Z}$, there exist RS codes with rate $R = \frac{k}{n}$ over a large enough (as a function of n and ε) finite field, that are list-decodable from radius $1 - R - \varepsilon$ and list size at most $\lceil \frac{1-R-\varepsilon}{\varepsilon} \rceil$.*

Conjecture 1.4 is stronger than our Theorem 1.1 about list-decoding. In particular, our theorem is near-optimal, but it is interesting mostly in the low-rate/high-noise parameter regime. In contrast, Conjecture 1.4 conjectures that there exist *exactly* optimal RS codes, in any parameter regime.

To encourage others to use our new connection and make progress on Conjecture 1.4, we propose a method of attack in Section 6. This outline exploits our connection to the Nash-Williams–Tutte theorem, and proceeds via a conjectured generalization of the Nash-Williams–Tutte theorem to hypergraphs: we show that establishing this hypergraph conjecture (which is stated as Conjecture 6.1 in Section 6) would in fact establish Conjecture 1.4. In Section 6, we give evidence for our hypergraph Nash-Williams–Tutte conjecture, Conjecture 6.1, observing that the “easy direction” of the conjecture follows from the Nash-Williams–Tutte theorem, and also that a quantitative relaxation of the conjecture follows from existing work [CS07, CCV09]. As further evidence of the viability of this approach, this quantitative relaxation implies a second proof of our main list-decoding result, Theorem 1.1, and we also sketch this proof in Section 6.²

1.2 Related Work

We briefly review related work. See Table 1 for a quantitative comparison to prior work.

List-decoding of RS codes. Ever since the Guruswami–Sudan algorithm [GS99], which efficiently list-decodes RS codes up to the Johnson bound, it has been open to understand the extent to which RS codes are list-decodable *beyond* the Johnson bound, and in particular if there are RS codes that are list-decodable all the way up to the list-decoding capacity theorem, matching the

²This second proof does not immediately establish list-recoverability, which is why we focus on our first proof.

performance of completely random codes. There have been negative results that show that *some* RS codes are not list-decodable to capacity [BKR10], and others that show that even if they were, in some parameter regimes we are unlikely to find an efficient list-decoding algorithm [CW07]. The work of Rudra and Wootters, mentioned above, showed that for any code with suitably good distance, a random puncturing of that code was likely to be near-optimally list-decodable; this implies that an RS code with random evaluation points is likely to be list-decodable. Unfortunately, as discussed above, this result requires a constant alphabet size q in order to yield a constant-rate code, while RS codes necessarily have $q \geq n$.

Recently, Shangguan and Tamo [ST20a] studied the list-decodability of RS codes in a different parameter regime, namely when the list size L is very small, either 2 or 3. They were able to get extremely precise bounds on the rate (showing that there are RS codes that are exactly optimal), but unfortunately for such small list sizes, it is impossible for any code to be list-decodable up to radius $1 - \varepsilon$ for small ε , which is our parameter regime of interest. Unlike the approach of [RW14], which applies to random puncturings of any code, the work of [ST20a] targeted RS codes specifically and developed an approach via studying *intersection matrices*. The reason that their approach stopped at $L = 3$ was the difficulty of analyzing these intersection matrices. We build on their approach and use techniques from graph theory—in particular, the Nash-Williams–Tutte theorem—to analyze the relevant intersection matrices beyond what [ST20a] were able to do. We discuss our approach more below in Section 1.3.

Subsequent work on list-decoding of RS codes. After our work first appeared, and inspired by our approach, Ferber, Kwan, and Sauermaun [FKS20] gave a beautiful proof establishing the existence of RS codes with rate $\Omega(\varepsilon)$ that are list-decodable from radius $1 - \varepsilon$ with list size $O(1/\varepsilon)$, over a polynomially (in the code’s length) large finite field.³ Compared with our result on the list-decodability of RS codes, their result removes the logarithmic factor in $1/\varepsilon$, and allows for smaller alphabet sizes; additionally, their proof is much shorter. In further follow-up work, Goldberg, Shangguan, and Tamo [GST21] further improved the rate from $\Omega(\varepsilon)$ to a rate approaching $\frac{\varepsilon}{2-\varepsilon}$.

However, we believe that there are still some advantages to our approach (beyond inspiring that of [FKS20] and [GST21]). First, the result of [FKS20] does not apply to list-recovery, and while [GST21] does apply to list-recovery, they do not surpass the $1/\ell$ barrier in the rate. Second, neither [FKS20] nor [GST21] fully resolve Conjecture 1.4 about optimal list-decodability of RS codes. We believe that the framework and tools developed in this paper together with Conjecture 6.1 provide a plausible attack method to resolve Conjecture 1.4.

List-recovery of RS codes. While the Guruswami–Sudan algorithm is in fact a list-recovery algorithm, much less was known about the list-recovery of RS codes beyond the Johnson bound than was known about list-decoding. (There is a natural extension of the Johnson bound for list-recovery, see [GS01]; for RS codes, it implies that an RS code of rate about ε^2/ℓ is list-recoverable up to radius $1 - \varepsilon$ with input list sizes ℓ and polynomial output list size). As with list-decoding, it is known that *some* RS codes are not list-recoverable beyond the Johnson bound [GR06]. However, much less was known on the positive front. In particular, neither of the works [RW14, ST20a] discussed above work for list-recovery. In a recent work, Lund and Potuchuki [LP20] have proved an analogous statement to that of [RW14]: any code of decent distance, when randomly punctured to an appropriate length, yields with high probability a good list-recoverable code. This implies

³In fact, they show something more general: if one begins with any code of sufficiently large distance over a sufficiently large alphabet, and randomly punctures it to rate $\Omega(\varepsilon)$, the resulting code is with high probability $(1 - \varepsilon, O(1/\varepsilon))$ list-decodable.

the existence of RS codes that are list-recoverable beyond the Johnson bound. However, in [LP20] there is again a dependence on $\log(q)$ in the rate bound, meaning that for RS codes, the rate must be sub-constant. Further, the work of [LP20] only applies up to radius $\rho = 1 - 1/\sqrt{2}$, and in particular does not apply to radii $\rho = 1 - \varepsilon$, as we study in this work. Our results also work in the constant- ρ setting of [LP20], and in that regime we show that RS codes of rate $\Omega(1/\sqrt{\ell})$ are $(\rho, \ell, O(\ell))$ list-recoverable, which improves over the result of [LP20] by a factor of $\log q$ in the rate. However, we do require the field size to be much larger than that is required by [LP20] (see Table 1).

Subsequent work on list-recovery of RS codes. The recent work of Goldberg, Shangquan, and Tamo [GST21] mentioned above builds on [FKS20], and shows that there are RS codes of rate approaching $\frac{\varepsilon}{1+\ell-\varepsilon}$ that are $(1 - \varepsilon, \ell, L_{\varepsilon, \ell})$ -list-recoverable, for a constant $L_{\varepsilon, \ell}$ that depends only on ε and ℓ . Compared to our work, while [GST21] improves the dependence on ε in the rate by a factor of $\log(1/\varepsilon)$, it has a worse dependence on ℓ , and in particular does not break the $1/\ell$ barrier that is present in the Johnson bound.

List-decoding and list-recovery of RS-like codes. There are constructions—for example, of *folded RS codes* and *univariate multiplicity codes* [GR08, GW13, Kop15, KRZSW18]—of codes that are based on RS codes and that are known to achieve list-decoding (and list-recovery) capacity, with efficient algorithms. Our goal in this work is to study Reed–Solomon codes themselves.

Perfect hash matrices and strongly perfect hash matrices. Perfect hash matrices have been studied extensively since the 1980s. There are two parameter regimes that are studied. The first is when the alphabet size q is constant and the number of rows tends to infinity [Nil94, FK84, KM88, Kör86, XY19]. The second is when the number of rows is viewed as a constant, while q may tend to infinity [BW98, Bla00, SG16]. In both cases the strength t of a perfect hash matrix is a constant. Our work studies the second case; as mentioned above, Blackburn [Bla00] gave an optimal construction for linear hash matrices in this parameter regime, and as a special case we obtain a second proof of Blackburn’s result.

The study of strongly perfect hash matrices is relatively new [ST20b]. The thesis [Dou19] collected some recent results on a closely related topic. However, the parameters considered there are quite different from those in our paper, and to the best of our knowledge, our construction is the best known in the parameter regime we consider. Another related notion called *balanced hashing* was introduced in [AG07, AG09], where, with our notation, any set of t columns of a matrix needs to be separated by at least a_1 and at most a_2 rows, for some integers $a_1 \leq a_2$. Note that in our setting, we want every set of t columns to be separated by as many rows as possible, while in the setting of balanced hashing it cannot exceed the threshold a_2 ; thus, the two settings are incomparable.

1.3 Technical Overview

Intersection matrices. Our approach is centered around *intersection matrices*, introduced in [ST20a]. Intersection matrices and their nonsingularity are defined formally below in Definition 2.2, but we give a brief informal introduction here. A t -wise intersection matrix, M , is defined by a collection of sets $I_1, I_2, \dots, I_t \subseteq [n]$, and has entries that are monomials in $\mathbb{F}_q[x_1, x_2, \dots, x_n]$. It was shown in [ST20a] that if there is a counter-example to the list-decodability of a Reed–Solomon code with evaluation points $(\alpha_1, \dots, \alpha_n)$ —that is, if there exist polynomials f_1, f_2, \dots, f_{L+1} that

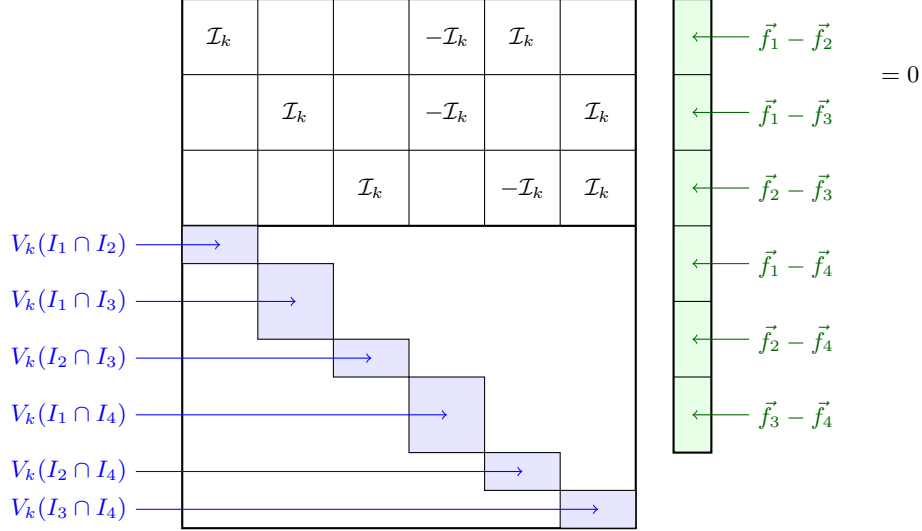


Figure 1: Let $f_1, f_2, f_3, f_4 \in \mathbb{F}_q[x]$ have degree $k - 1$ and suppose that $I_j = \{i : f_j(\alpha_i) = g(\alpha_i)\}$. (In particular, f_i and f_j agree on $I_i \cap I_j$). Then the matrix-vector product depicted above is zero, where the vector \vec{f}_i refers to the k coefficients of the polynomial f_i , and the i -th coordinate of this vector is the coefficient of x^{i-1} in f . Here, $V_k(I_i \cap I_j) \in \mathbb{F}_q^{|I_i \cap I_j| \times k}$ denotes the Vandermonde matrix with (i, j) entry equal to α_i^{j-1} for $1 \leq i \leq |I_i \cap I_j|$ and $j \in [k]$. The notation \mathcal{I}_k denotes the $k \times k$ identity matrix.

all agree with some other polynomial $g : \mathbb{F}_q \rightarrow \mathbb{F}_q$ at many points α_i —then there is a $(L + 1)$ -wise intersection matrix that becomes singular when α_i is plugged in for x_i for all $i \in [n]$.

The set-up (both the definition of an intersection matrix and the connection to list-decoding) is most easily explained by an example. Suppose that we are interested in list-decoding for $L = 3$, and suppose that we are interested in a RS code with evaluation points $\alpha_1, \alpha_2, \dots, \alpha_n$. Let f_1, f_2, f_3, f_4 and g be a counter-example to list-decoding, as above, and for $1 \leq j \leq 4$, let $I_j = \{i \in [n] : f_j(\alpha_i) = g(\alpha_i)\}$. Now consider the product shown in Figure 1 (see the caption for notation).

An inspection of Figure 1 shows that the matrix-vector product depicted is zero. Indeed, the top part is zero for any choice of the f_i , and the bottom part is zero since f_i and f_j are assumed to agree on $\{\alpha_s : s \in I_i \cap I_j\}$. The matrix shown is the 4-wise intersection matrix for the sets I_1, I_2, I_3, I_4 , evaluated at $\alpha_1, \dots, \alpha_n$. If the f_i 's agree too much with the function g (i.e., if they are a counter-example to list-decodability for some given radius), then the sets $I_i \cap I_j$ are going to be larger, and this matrix will have more rows. In particular, the more the f_i 's agree with g , the harder it is for this matrix to be singular. Intuitively, this sets us up for a proof by contradiction: if f_1, f_2, f_3, f_4 agree too much with g , then this matrix is nonsingular (at least for a non-pathological choice of α_i 's); but Figure 1 displays a kernel vector!

A t -wise intersection matrix (for sets I_1, \dots, I_t) generalizes a 4-wise intersection matrix shown in Figure 1. The bottom part looks exactly the same—a block-diagonal matrix with Vandermonde blocks—and the top part is an appropriate generalization that causes the analogous $k \cdot \binom{t}{2}$ -long vector corresponding to the f_i 's to vanish.

A conjecture about t -wise intersection matrices. With the motivation in Figure 1, the strategy of [ST20a] was to study t -wise intersection matrices M for $t = L + 1$, and to show that for every appropriate choice of I_1, \dots, I_t , the polynomial $\det(M) \in \mathbb{F}_q[x_1, x_2, \dots, x_n]$ is not identically zero. The list-decodability of RS codes would then follow from the DeMillo–Lipton–Schwartz–Zippel lemma along with a counting argument. In particular, they made the following conjecture,

and showed that it implies Conjecture 1.4 about list-decoding. Below, the *weight* of a family of subsets I_1, \dots, I_t of $[n]$ is defined to be

$$\text{wt}(I_1, \dots, I_t) = \sum_{i=1}^t |I_i| - \left| \bigcup_{i=1}^t I_i \right|, \quad (1)$$

and for a set J of indices, we use the shorthand $\text{wt}(I_J) := \text{wt}(I_j : j \in J)$.

Conjecture 1.5 (Conjecture 5.7 of [ST20a]). *Let $t \geq 3$ be an integer and $I_1, \dots, I_t \subseteq [n]$ be subsets satisfying*

- (i) $\text{wt}(I_J) \leq (|J| - 1)k$ for all nonempty $J \subseteq [t]$,
- (ii) Equality holds for $J = [t]$, i.e., $\text{wt}(I_{[t]}) = (t - 1)k$.

Then the t -wise intersection matrix $M_{k, (I_1, \dots, I_t)}$ is nonsingular over any finite field.

The conditions (i) and (ii) above turn out to be the right way of quantifying “the f_i ’s agree enough with g .” That is, if the f_i ’s agree too much with g (in the sense of going beyond Conjecture 1.4 about list-decoding), then it is possible to find sets I_j so that (i) and (ii) hold.

Unfortunately, the work of [ST20a] was only able to establish Conjecture 1.5 for $t = 3, 4$ (corresponding to $L = 2, 3$), and it seemed challenging to extend their techniques directly to much larger values of L .

Establishing the conjecture under an additional assumption, and using that to establish our main results. In this work, we use a novel connection to the Nash-Williams–Tutte theorem, which establishes the existence of pairwise edge-disjoint spanning trees in a graph, to extend the results of [ST20a] to larger L , at the cost of an additional assumption. More precisely, we are able to show in Theorem 3.1 (stated and proved in Section 3) that Conjecture 1.5 holds, *provided* that the sets I_j do not have any nontrivial three-wise intersections $I_i \cap I_j \cap I_\ell \neq \emptyset$.

The connection to the Nash-Williams–Tutte theorem is explained in Section 3. Briefly, we consider each term in the expression

$$\det(M) = \sum_{\sigma \in S_n} (-1)^{\text{sgn}(\sigma)} \prod_{i=1}^n M_{i, \sigma(i)}.$$

We show that $\prod_{i=1}^n M_{i, \sigma(i)}$ is a nonzero monomial in x_1, \dots, x_n if and only if σ picks out a tree packing of a graph⁴ that is determined by the sets I_1, \dots, I_t . It turns out that the requirements of (i) and (ii) in Conjecture 1.5 translate exactly into the requirements needed to apply the Nash-Williams–Tutte theorem to this graph. Thus, if (i) and (ii) hold, then there exists a tree packing in this graph and hence a nonzero term in $\det(M)$.

If the sets $I_i \cap I_j$ and $I_{i'} \cap I_{j'}$ that appear in the lower part of the t -wise intersection matrix do not intersect (that is, if there are no three-wise intersections among the sets I_j), then the reasoning above is enough to establish the conclusion of Conjecture 1.5, because all of the terms that appear in the expansion of the determinant are distinct monomials, and they cannot cancel. This is why Theorem 3.1 has this assumption.

While Theorem 3.1 is not strong enough to immediately establish results for list-decoding or list-recovery (indeed, there is no reason that there should not be three-wise intersections for the

⁴Throughout this paper, a tree packing of a graph G means a collection of pairwise edge-disjoint spanning trees of G .

polynomials f_i discussed above), it *is* enough for our application to perfect hash matrices, which we work out in Section 4.

In order to apply Theorem 3.1 to list-decoding, we back off from Conjecture 1.5 a bit. First, we allow a factor of $\Theta(\log t)$ slack on the right-hand sides of (i) and (ii). Second, rather than showing that the t -wise intersection matrix $M_{k,(I_1,\dots,I_t)}$ is nonsingular, we show that there exists a t' -wise intersection matrix that is nonsingular for some $t' < t$. Following the connection of [ST20a] illustrated in Figure 1, this turns out to be enough to establish our main theorem on list-decoding/recovery.

We choose this smaller intersection matrix in Lemma 5.3 by carefully choosing a random subset J of $[t]$. By greedily removing elements from the sets $\{I_j : j \in J\}$, we can obtain subsets $I'_j \subset I_j$ with empty three-wise intersections $I'_j \cap I'_{j'} \cap I'_{j''} = \emptyset$. Furthermore, by the careful random choice of J , and since we allowed a $\Theta(\log t)$ slack in the initial weight bounds, we can show this step does not delete too many elements. This is the key step of Lemma 5.3. Using some of the sets $\{I_j : j \in J\}$, we can find a smaller intersection matrix obeying the setup of Conjecture 1.5 with the additional guarantee that all three-wise intersections are empty. We provide a more detailed summary of the proof in Section 5.1.

Another avenue to list-decoding: a hypergraph Nash-Williams–Tutte conjecture. Extending our connection of list-decoding RS codes to the Nash-Williams–Tutte theorem, we show that a suitable *hypergraph* generalization of the Nash-Williams–Tutte theorem would imply Conjecture 1.5 about the nonsingularity of intersection matrices, without any need for an additional assumption about three-wise intersections of the sets I_j .

We conjecture that such a generalization is true, and we state it in Section 6 as Conjecture 6.1. It requires a bit of notation to set up, so we do that in Section 6 rather than here; however, the reader interested in the hypergraph conjecture can at this point jump straight to Section 6 without missing anything.

We show that if our hypergraph conjecture were true, it would imply Conjecture 1.5, on the nonsingularity of intersection matrices (Theorem 6.2). This in turn would imply Conjecture 1.4, establishing the existence of RS codes with optimal list-decodability. This suggests a plan of attack towards Conjecture 1.4.

While we are unable to establish this challenging conjecture in full, we give some evidence for it. First, we show that the “easy part” of the conjecture follows from the Nash-Williams–Tutte theorem. Second, we observe that a quantitative relaxation of the conjecture follows from known results on Steiner tree packings [CS07] and disjoint bases of polymatroids [CCV09]. This relaxation can be combined with the connection of hypergraph packings and intersection matrices established in Theorem 6.2, and the connection between intersection matrices and list decoding RS codes, to give a second proof of Theorem 1.1, that there are *near*-optimally list-decodable RS codes.

In addition to implying the optimal list-decodability of RS codes, Conjecture 6.1 may be of independent interest. A hypergraph generalization of Nash-Williams–Tutte is known for *partition-connected* hypergraphs [FKK03] (see Section 6 for definition), a well studied notion. However, for a different notion called *weak-partition-connectivity*, less seems to be known, and Conjecture 6.1 poses a Nash-Williams–Tutte generalization for weakly-partition-connected hypergraphs.

Organization. A graphical overview of our results can be found in Figure 2. We begin in Section 2 with the needed notation and definitions, including the definition of t -wise intersection matrices.

In Sections 3, 4, and 5, we prove Theorem 1.1 and Theorem 1.2 using our proof of Conjecture 1.5 under the additional assumption of no three-wise intersections. More precisely, in Section 3, we

show how to use the Nash-Williams–Tutte theorem from graph theory to prove Theorem 3.1, which establishes Conjecture 1.5 under the assumption of no three-wise intersections. In Section 4 we use Theorem 3.1 to prove Theorem 1.3 about perfect hash matrices. A reader interested in the list-recovery result can skip Section 1.3 and continue directly to Section 5. In Section 5 we prove Theorem 1.2 on the list-recoverability of RS codes. As list-recovery is more general than list-decoding, this gives our first proof of Theorem 1.1 on list-decoding.

In Sections 6, we conjecture a hypergraph generalization of the Nash-Williams–Tutte theorem and prove (Theorem 6.2) that it implies optimal list-decodability of RS codes. We also give some evidence for it by observing an “easy direction” follows from the ordinary Nash-Williams–Tutte theorem, by highlighting a known relaxation, and sketching how this relaxation gives us a second proof of Theorem 1.1.

1.4 Future Directions and Open Questions

In this work, we have shown the existence of near-optimally list-decodable RS codes in the large-radius parameter regime. To do this, we have established a connection between the intersection matrix approach of [ST20a] and tree packings. Along the way, we also developed applications to the construction of strongly perfect hash matrices, and we have introduced a new hypergraph version of the Nash-Williams–Tutte theorem. We highlight a few questions that remain open.

Can RS codes *exactly* achieve list-decoding capacity? In spite of the results and tools developed in this paper, we were not able to prove Conjecture 1.4. We hope that the avenue of attack discussed in Section 6 will be able to finish the job. We note that the analogous question regarding the limits of list-recoverability of RS codes also remains open.

Efficient list-decoding of RS codes? We remark that, using a simple idea from [ST20a] one can convert each of the existence results of RS codes reported in this paper into an explicit code construction, although over a much larger field size. Hence, given such an explicit code construction, is it possible to decode it efficiently up to its guaranteed list-decoding radius? A similar question can be asked for list-recoverability. We note that [CW07], which shows that decoding RS codes much beyond the Johnson bound is likely hard in certain parameter regimes, does not apply to our parameter regime when the field size is large.

Generalizing the Nash-Williams–Tutte theorem to hypergraphs. In an attempt to resolve Conjecture 1.4, we present Conjecture 6.1, a new graph-theoretic conjecture, which can be viewed as a generalization of the Nash-Williams–Tutte theorem to hypergraphs. In addition to being interesting on its own, resolving this conjecture would imply the existence of optimally list-decodable RS codes.

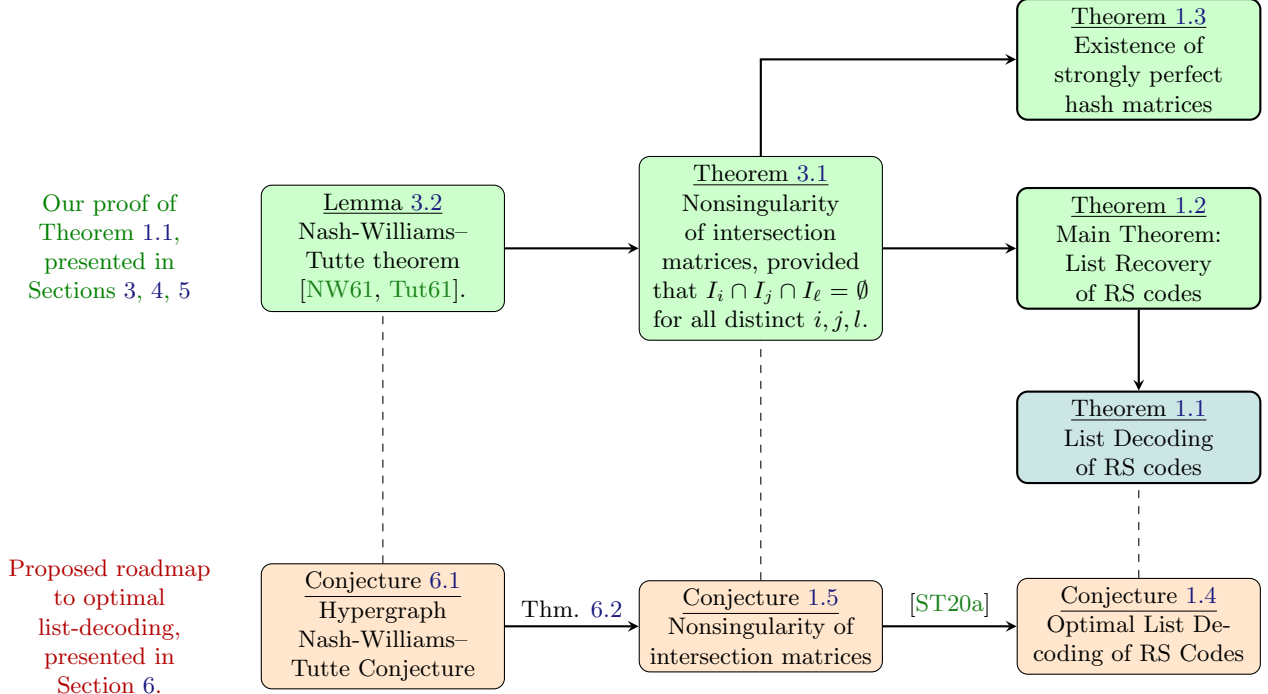


Figure 2: A diagram of the results and conjectures presented in this work. Solid arrows represent logical implications. Dashed lines indicate how the proposed roadmap to optimal list decoding parallels our proof of Theorem 1.1.

2 Preliminaries

The main goal of this section is to present the definition of t -wise intersection matrices over an arbitrary field \mathbb{F} .

Let $\mathbb{N}^+ = \{1, 2, \dots\}$ and $[n] = \{1, 2, \dots, n\}$ for $n \in \mathbb{N}^+$. Denote by $\log x$ the base-2 logarithm of x . For a finite set X and an integer $1 \leq k \leq |X|$, let $\binom{X}{k} = \{A \subseteq X : |A| = k\}$ be the family of all k -subsets of X . For an integer $t \geq 3$, we define the following lexicographic order on $\binom{[t]}{2}$. For distinct $S_1, S_2 \in \binom{[t]}{2}$, $S_1 < S_2$ if and only if $\max(S_1) < \max(S_2)$ or $\max(S_1) = \max(S_2)$ and $\min(S_1) < \min(S_2)$. For a partition \mathcal{P} of X , let $|\mathcal{P}|$ denote the number of parts of \mathcal{P} . In the remaining part of this paper, assume that n, k are integers satisfying $1 \leq k < n$.

We view a polynomial $f \in \mathbb{F}_q[x]$ of degree at most $k - 1$ as a vector of length k defined by its k coefficients, where for $1 \leq i \leq k$, the i -th coordinate of this vector is the coefficient of x^{i-1} in f . By abuse of notation that vector is also denoted by f .

2.1 Cycle Spaces

We need the notion of the *cycle space* of a graph, which is typically defined over the boolean field \mathbb{F}_2 (see, e.g., [Die17]). Here we define it over an arbitrary field \mathbb{F} . An equivalent definition can be found in [BBN93], where it is called the “circuit-subspace”.

Let K_t be the undirected complete graph with the vertex set $[t]$. Denote by $\{i, j\}$ the edge connecting vertices i and j . Let K_t^o be the oriented graph obtained by replacing $\{i, j\}$ with the directed edge (i, j) for all $1 \leq i < j \leq t$. For a graph G with vertex set $[t]$, an *oriented cycle* in G

is a set of directed edges of the form

$$C = \{(i_0, i_1), (i_1, i_2), \dots, (i_{m-1}, i_m)\}$$

where $m \geq 3$, i_0, \dots, i_{m-1} are distinct, $i_m = i_0$ and $\{i_{j-1}, i_j\}$ is an edge of G for all $j = 1, \dots, m$.

Suppose C is a union of edge-disjoint oriented cycles in G . Then C is uniquely represented by a vector $u^C = (u_{\{i,j\}}^C : \{i,j\} \in \binom{[t]}{2}) \in \mathbb{F}^{\binom{[t]}{2}}$, defined for $1 \leq i < j \leq t$ by

$$u_{\{i,j\}}^C = \begin{cases} 1 & (i,j) \in C, \\ -1 & (j,i) \in C, \\ 0 & \text{else.} \end{cases}$$

Hence, the sign of a nonzero coordinate $u_{\{i,j\}}^C$ indicates whether the orientation of $\{i,j\}$ in C complies with its orientation in K_t^o . We further assume that the coordinates of u^C are ordered by the aforementioned lexicographic order on $\binom{[t]}{2}$.

Denote by $C(G) \subseteq \mathbb{F}^{\binom{[t]}{2}}$ the subspace spanned by the set of vectors

$$\{u^C : C \text{ is an oriented cycle in } G\}$$

over \mathbb{F} . We call $C(G)$ the *cycle space* of G over \mathbb{F} . We are particularly interested in the cycle space $C(K_t)$ of K_t . For distinct $i, j, \ell \in [t]$, denote by $\Delta_{ij\ell}$ the oriented cycle $\{(i,j), (j,\ell), (\ell,i)\}$ and call it an *oriented triangle*. We have the following lemma, generalizing [Die17, Theorem 1.9.5].

Lemma 2.1. *The vector space $C(K_t) \subseteq \mathbb{F}^{\binom{[t]}{2}}$ has dimension $\binom{t-1}{2}$, and the set*

$$\mathcal{B}_t = \{u^{\Delta_{ij\ell}} : 1 \leq i < j \leq t-1\}$$

is a basis of $C(K_t)$.

Proof. The vectors in \mathcal{B}_t are linearly independent since $u_{\{i,j\}}^{\Delta_{ij\ell}} = 1$ and $u_{\{i',j'\}}^{\Delta_{ij\ell}} = 0$ for $1 \leq i < j \leq t-1$ and $1 \leq i' < j' \leq t-1$ with $\{i,j\} \neq \{i',j'\}$. Let W be the span of \mathcal{B}_t over \mathbb{F} . Consider an arbitrary oriented cycle C in K_t . We claim that $u^C \in W$, and this would imply that \mathcal{B}_t is a basis of $C(K_t)$ and that the dimension of $C(K_t)$ is $|\mathcal{B}_t| = \binom{t-1}{2}$.

Denote by e_C the smallest $\{i,j\} \in \binom{[t]}{2}$ in the lexicographic order such that $(i,j) \in C$ or $(j,i) \in C$. Next, we will prove the claim by a reverse induction on the lexicographic order of e_C . Note that $t \notin e_C$ since $|C| \geq 3$, which implies that the claim is vacuously true when $e_C = \{t-1, t\}$ (which never occurs). Now assume that the claim holds for all oriented cycles C' with $e_{C'} > e_C$. Let $\{i,j\} = e_C$, where $i < j$. We may assume that $(i,j) \in C$ by flipping the orientation of C if necessary, which corresponds to negating u^C .

Let s be the number of directed edges that C and $\Delta_{ij\ell}$ share. If $s = 3$ then it is clear by definition that $C = \Delta_{ij\ell}$, and we are done. Otherwise, $1 \leq s \leq 2$ and it is easy to verify that $u^C - u^{\Delta_{ij\ell}} = u^{C'}$ for a set C' that is either an oriented cycle in G or a disjoint union of two oriented cycles C_1, C_2 in G passing through t . The latter case occurs when C passes through t and $(t,i), (j,t) \notin C$. In either case, the smallest edge (under the lexicographic order) of C' is greater than the edge $e_C = \{i,j\}$. Hence, by the induction hypothesis and the fact that $u^{C'} = u^{C_1} + u^{C_2}$ when C' is the disjoint union of C_1 and C_2 , we have $u^{C'} \in W$. So $u^C = u^{C'} + u^{\Delta_{ij\ell}} \in W$, completing the proof of the claim. \square

The basis \mathcal{B}_t is also viewed as a $\binom{t-1}{2} \times \binom{[t]}{2}$ matrix over \mathbb{F} whose columns are labeled by the edges $\{i,j\}$ of K_t , according to the lexicographic order defined above. Moreover, the rows of \mathcal{B}_t

represent $u^{\Delta_{ijt}}$ for $1 \leq i < j \leq t-1$, and are labeled by $\{i, j\} \in \binom{[t-1]}{2}$, also according to the lexicographic order. For example, $\mathcal{B}_3 = (1, -1, 1)$ and

$$\mathcal{B}_4 = \begin{pmatrix} 1 & -1 & 1 & & & \\ & 1 & -1 & & 1 & \\ & & 1 & -1 & 1 & \end{pmatrix},$$

where the 6 columns are labeled and ordered lexicographically by $\{1, 2\} < \{1, 3\} < \{2, 3\} < \{1, 4\} < \{2, 4\} < \{3, 4\}$. Observe for example that the ± 1 entries in the first row correspond to the oriented triangle $\Delta_{124} = \{(1, 2), (2, 4), (4, 1)\}$, where we have -1 on the column labeled by the edge $\{1, 4\}$, since the directed edge $(4, 1)$ in Δ_{124} has the opposite orientation from the orientation of the edge in K_t^o .

We remark that the above definition of \mathcal{B}_t , is given with respect to the fixed orientation of the edges of K_t^o , as with the definition of u^C for any oriented cycle C . One may define \mathcal{B}_t with respect to other orientations of edges, which corresponds to changing the signs in some columns. These definitions are all equivalent and the analysis in this paper holds for any orientation up to change of signs.

Moreover, when the characteristic of \mathbb{F} is two, we recover the definition of \mathcal{B}_t in [ST20a] using the fact that $1 = -1$. While working in the case $\text{char}(\mathbb{F}) = 2$ has the advantage that there is no need to distinguish the signs, the theory holds more generally over any field.

2.2 t -Wise Intersection Matrices

We proceed to define t -wise intersection matrices, but we begin with a few preliminary definitions. Given n variables or field elements x_1, \dots, x_n , define the $n \times k$ Vandermonde matrix

$$V_k(x_1, \dots, x_n) = \begin{pmatrix} 1 & x_1 & \cdots & x_1^{k-1} \\ & & \ddots & \\ & & & \\ 1 & x_n & \cdots & x_n^{k-1} \end{pmatrix}. \quad (2)$$

When the x_i 's are understood from the context, for $I \subseteq [n]$, we use the abbreviation $V_k(I) := V_k(x_i : i \in I)$ to denote the restriction of $V_k(x_1, \dots, x_n)$ to the rows with indices in I .

Let \mathcal{I}_k denote the identity matrix of order k . Next, we give the definition of t -wise intersection matrices.

Definition 2.2 (t -wise intersection matrices). *For a positive integer k and $t \geq 3$ subsets $I_1, \dots, I_t \subseteq [n]$, the t -wise intersection matrix $M_{k, (I_1, \dots, I_t)}$ is the $\binom{t-1}{2}k + \sum_{1 \leq i < j \leq t} |I_i \cap I_j| \times \binom{t}{2}k$ variable matrix with entries in $\mathbb{F}[x_1, \dots, x_n]$, defined as*

$$\left(\frac{\mathcal{B}_t \otimes \mathcal{I}_k}{\text{diag}(V_k(I_i \cap I_j) : \{i, j\} \in \binom{[t]}{2})} \right),$$

where \otimes is tensor product of matrices and

- $\mathcal{B}_t \otimes \mathcal{I}_k$ is a $\binom{t-1}{2}k \times \binom{t}{2}k$ matrix with entries in $\{0, \pm 1\}$,

- $\text{diag}(V_k(I_i \cap I_j) : \{i, j\} \in \binom{[t]}{2})$ is a block diagonal matrix with blocks $V_k(I_i \cap I_j)$, ordered by the lexicographic order on $\{i, j\} \in \binom{[t]}{2}$. Note that this matrix has order $(\sum_{1 \leq i < j \leq t} |I_i \cap I_j|) \times \binom{t}{2} k$. If $I_i \cap I_j = \emptyset$ for some i, j , then $V_k(I_i \cap I_j)$ is of order $0 \times k$ and the $\{i, j\} \in \binom{[t]}{2}$ block of k columns is a $\sum_{1 \leq i < j \leq t} |I_i \cap I_j| \times k$ zero matrix.

The reader is referred to Appendix A (see Example A.1) for an example of a 4-wise intersection matrix. We note that when $t = 2$, \mathcal{B}_t is an empty matrix and $M_{k, (I_1, I_2)}$ is simply a Vandermonde matrix.

For a vector $\alpha \in \mathbb{F}^n$, the evaluation of $M_{k, (I_1, \dots, I_t)}$ at the vector α is denoted by $M_{k, (I_1, \dots, I_t)}(\alpha)$, where each variable x_i is assigned the value α_i . Given subsets $I_1, \dots, I_t \subseteq [n]$, we call the variable matrix $M_{k, (I_1, \dots, I_t)}$ *nonsingular* if it contains at least one $\binom{t}{2} k \times \binom{t}{2} k$ submatrix whose determinant is a nonzero polynomial in $\mathbb{F}[x_1, \dots, x_n]$.

The paper [ST20a] connects the nonsingularity of intersection matrices to the list-decodability of RS codes. We will use this connection to prove our main result, Theorem 1.2.

However, we will first prove that *certain* intersection matrices are nonsingular. This will both allow us to cleanly illustrate the connection to disjoint tree packings of graphs, and it will also yield Theorem 1.3 about perfect hash matrices. We will do this in Theorem 3.1 in the next section.

3 Connection to Tree Packing and an Intermediate Result

In this section we prove the following theorem. We recall from (1) the definition of the *weight* of a collection of sets:

$$\text{wt}(I_1, \dots, I_t) = \sum_{i=1}^t |I_i| - \left| \bigcup_{i=1}^t I_i \right|,$$

Theorem 3.1. *Let $t \geq 2$ be an integer and $I_1, \dots, I_t \subseteq [n]$ be subsets satisfying (i) $I_i \cap I_j \cap I_l = \emptyset$ for all $1 \leq i < j < l \leq t$; (ii) $\text{wt}(I_J) \leq (|J| - 1)k$ for all nonempty $J \subseteq [t]$; (iii) $\text{wt}(I_{[t]}) = (t - 1)k$. Then the t -wise intersection matrix $M_{k, (I_1, \dots, I_t)}$ is nonsingular over any field.*

As discussed above, this theorem stops short of Conjecture 1.5, due to the assumption that $I_i \cap I_j \cap I_l = \emptyset$. In the language of list-decoding Reed–Solomon codes, this only gives us a statement about lists of potential codewords that have no three-wise intersections. However, we will build on this statement to prove our main theorem about list-recovery (Theorem 1.2), and moreover this is already enough to prove our result on the existence of strongly perfect hash matrices (Theorem 1.3).

The main tool of proving Theorem 3.1 is the following classical result in graph theory.

Lemma 3.2 (Nash-Williams [NW61], Tutte [Tut61], see also Theorem 2.4.1 of [Die17]). *A multi-graph contains k edge-disjoint spanning trees if and only if for every partition \mathcal{P} of its vertex set it has at least $(|\mathcal{P}| - 1)k$ cross-edges. Here an edge is called a cross-edge for \mathcal{P} if its two endpoints are in different members of \mathcal{P} .*

In order to apply the Nash-Williams–Tutte theorem, we will construct a graph G from the sets I_1, I_2, \dots, I_t . We first note that the assumptions on I_1, \dots, I_t from Theorem 3.1 imply some nice properties that will later allow us to apply Lemma 3.2.

Claim 3.3. *Suppose that I_1, \dots, I_t are subsets satisfying the assumptions of Theorem 3.1. Then the matrix $M_{k, (I_1, \dots, I_t)}$ is a square matrix of order $\binom{t}{2}k$. Further, for any $J \subseteq [t]$ with $|J| \geq 2$,*

$$\text{wt}(I_J) = \sum_{\{i,j\} \in \binom{J}{2}} |I_i \cap I_j|. \quad (3)$$

Proof. By (1) (the definition of weight) and the inclusion-exclusion principle

$$\text{wt}(I_{[t]}) = \sum_{i=1}^t |I_i| - \left| \bigcup_{i=1}^t I_i \right| = \sum_{j=2}^t \sum_{J \in \binom{[t]}{j}} (-1)^{|J|} \left| \bigcap_{i \in J} I_i \right|. \quad (4)$$

Therefore, by assumption (i) of Theorem 3.1 we have $\text{wt}(I_{[t]}) = \sum_{1 \leq i < j \leq t} |I_i \cap I_j|$. Then, by assumption (iii) the matrix $M_{k, (I_1, \dots, I_t)}$ is in fact a square matrix of order $\binom{t}{2}k$. Similarly, (3) holds for any $J \subseteq [t]$ with $|J| \geq 2$. \square

To prove Theorem 3.1, let us construct a multigraph G defined on a set V of t vertices, say $V = \{v_1, \dots, v_t\}$. For $1 \leq i < j \leq t$, connect vertices v_i, v_j by $|I_i \cap I_j|$ multiple edges.

Applying Lemma 3.2 to G leads to the following claim.

Claim 3.4. *Let G be as above. Then G contains k edge-disjoint spanning trees.*

Proof. Let $\mathcal{P} = \{V_1, \dots, V_s\}$ be an arbitrary partition of V . Then it is clear that $\sum_{i=1}^s |V_i| = t$. According to Lemma 3.2, to prove the claim it suffices to show that G has at least $(s-1)k$ cross-edges with respect to \mathcal{P} . By (3) and assumption (iii) of Theorem 3.1 it is easy to see that G contains $\sum_{1 \leq i < j \leq t} |I_i \cap I_j| = (t-1)k$ edges. Moreover, by (3) and assumption (ii) of Theorem 3.1 one can infer that for each $i \in [s]$, the induced subgraph of G on the vertex set V_i has at most $\text{wt}(I_j : j \in V_i) \leq (|V_i| - 1)k$ edges. It follows that the number of cross-edges of G (with respect to \mathcal{P}) is at least

$$(t-1)k - \sum_{i=1}^s (|V_i| - 1)k = \left(t - 1 - \sum_{i=1}^s |V_i| + s \right) k = (s-1)k,$$

as needed, thereby completing the proof of the claim. \square

Below, we will relate a tree packing of this graph G to the determinant of the intersection matrix $M_{k, (I_1, \dots, I_t)}$. In order to do this, we first record a property of the matrix \mathcal{B}_t . Recall that the columns of \mathcal{B}_t are indexed by $\binom{[t]}{2}$.

Claim 3.5. *Removing a set of columns from \mathcal{B}_t will not reduce its row rank if and only if the columns are labeled by an acyclic subgraph of K_t .*

Proof. First we prove the if direction. Assume to the contrary that we can remove from \mathcal{B}_t some columns labeled by an acyclic subgraph H of K_t and reduce the row rank. Let \mathcal{B}'_t be the submatrix of \mathcal{B}_t after the removal of the columns labeled by H . The rows of \mathcal{B}'_t are linearly dependent by assumption. Hence, there exists a nonzero vector $u \in \mathbb{F}^{\binom{t-1}{2}}$ such that $u \cdot \mathcal{B}'_t = 0$. As $u \neq 0$ and the rows of \mathcal{B}_t are linearly independent, we have $u \cdot \mathcal{B}_t \neq 0$. Let $S \subseteq \binom{[t]}{2}$ be the support of $u \cdot \mathcal{B}_t$, where the support of a vector of length n is the subset of $[n]$ that records the indices of its nonzero coordinates. As $u \cdot \mathcal{B}_t \neq 0$ and $u \cdot \mathcal{B}'_t = 0$, we have $\emptyset \neq S \subseteq H$.

Consider the $\binom{t}{2} \times t$ matrix $D = (D_{\{i,j\},s})$ which is defined by

$$D_{\{i,j\},s} = \begin{cases} 1 & s = j, \\ -1 & s = i, \\ 0 & \text{otherwise,} \end{cases}$$

where $1 \leq i < j \leq t$ and $s \in [t]$. Note that the rows and columns of D are labeled by $\{i, j\} \in \binom{[t]}{2}$ and $s \in [t]$ respectively. It is easy to verify that $\mathcal{B}_t \cdot D = 0$, which implies that $u \cdot \mathcal{B}_t \cdot D = 0$. Denote $u \cdot \mathcal{B}_t$ by $w = (w_{\{i,j\}}) \in \mathbb{F}^{\binom{[t]}{2}}$, whose support is S . As $\emptyset \neq S \subseteq H$ and H is acyclic, we can find $s_0 \in [t]$ whose degree in S is one, i.e., there exists a unique edge $\{i_0, j_0\} \in S$ such that $s_0 \in \{i_0, j_0\}$. Then, the s_0 -th entry of $w \cdot D$ is

$$\sum_{\{i,j\} \in \binom{[t]}{2}} w_{\{i,j\}} D_{\{i,j\},s_0} = w_{\{i_0,j_0\}} D_{\{i_0,j_0\},s_0} = \pm w_{\{i_0,j_0\}} \neq 0,$$

which is a contradiction as $w \cdot D = u \cdot \mathcal{B}_t \cdot D = 0$.

Now we prove the only if direction. It suffices to prove that removing from \mathcal{B}_t a set of columns labeled by a cycle C of K_t will reduce its row rank by at least 1. Let us orient the edges of C to make it an oriented cycle, which by abuse of notation is also denoted by C . Since the rows of \mathcal{B}_t form a basis of $C(K_t)$, there is a nonzero vector $u \in \mathbb{F}^{\binom{[t]}{2}}$ such that $u \cdot \mathcal{B}_t = u^C$. Let \mathcal{B}'_t be the submatrix of \mathcal{B}_t after the removal of the columns labeled by C . Then it is not hard to check that $u \cdot \mathcal{B}'_t = 0$, which implies that the rows of \mathcal{B}'_t are linearly dependent, as needed. \square

Next we present the proof of Theorem 3.1. Recall from Claim 3.3 that under the assumptions of Theorem 3.1, the t -wise intersection matrix

$$M_{k,(I_1,\dots,I_t)} = \left(\frac{\mathcal{B}_t \otimes \mathcal{I}_k}{\text{diag}\left(V_k(I_i \cap I_j) : \{i,j\} \in \binom{[t]}{2}\right)} \right),$$

is a square matrix of order $\binom{t}{2}k$, and is defined by exactly $(t-1)k$ variables x_s , $s \in S$, where $S \subseteq [n]$ is some subset of size $(t-1)k$. In order to prove that $M_{k,(I_1,\dots,I_t)}$ is nonsingular, we proceed to show the nonsingularity of the following matrix, obtained by permuting the columns and rows of $M_{k,(I_1,\dots,I_t)}$:

$$M'_{k,(I_1,\dots,I_t)} := \left(\frac{\mathcal{I}_k \otimes \mathcal{B}_t}{(\mathcal{C}_i : 0 \leq i \leq k-1)} \right),$$

where $\mathcal{C}_i = \text{diag}\left(V_k^{(i)}(I_j \cap I_{j'}) : \{j, j'\} \in \binom{[t]}{2}\right)$ and $V_k^{(i)}(I_j \cap I_{j'})$ is the $(i+1)$ -th column of $V_k(I_j \cap I_{j'})$. Above, $(\mathcal{C}_i : 0 \leq i \leq k-1)$ is a $(t-1)k \times \binom{t}{2}k$ variable matrix, which consists of the matrices \mathcal{C}_i stacked next to each other. See Figure 3 for an illustration, and Example A.2 in Appendix A for a concrete example.

Proof of Theorem 3.1. If $t = 2$, then $M_{k,(I_1,I_2)}$ is a $k \times k$ Vandermonde matrix, which is nonsingular, so assume $t \geq 3$. For the rest of the proof, we will consider the matrix $M' = M'_{k,(I_1,\dots,I_t)}$ discussed above, and show that it is nonsingular.

Let the graph G be as in the discussion above; recall that for distinct $i, j \in [t]$, two vertices v_i, v_j in G are connected by $|I_i \cap I_j|$ edges. By Claim 3.3, M' is a square matrix with $\binom{t}{2}k$ rows

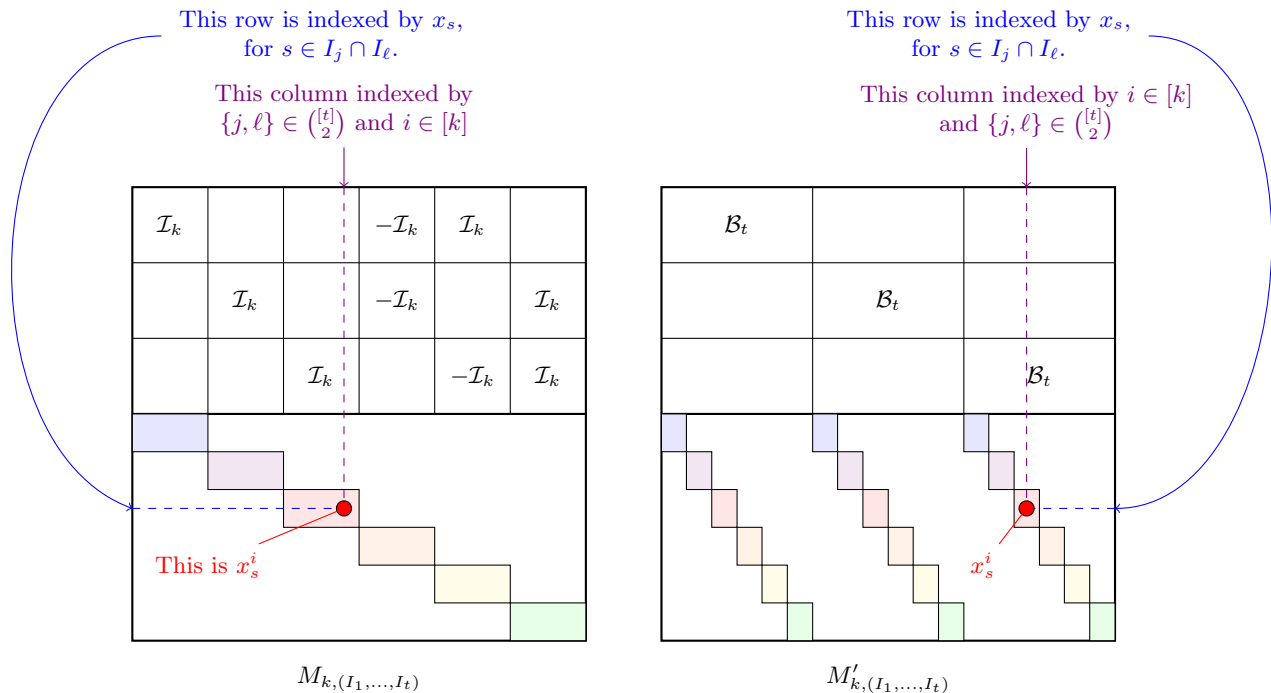


Figure 3: Re-ordering the rows/columns of an intersection matrix. (In this cartoon, $t = 4$ and $k = 3$).

and columns, and $k(t - 1)$ “variable” rows at the bottom. Let $S \subseteq [n]$ be the subset that records the indices of variables x_s that appear in M' .

For $1 \leq i < j \leq t$, fix an arbitrary one-to-one correspondence between the $|I_i \cap I_j|$ edges connecting v_i, v_j and the $|I_i \cap I_j|$ variables $x_s \in S$ so that $s \in I_i \cap I_j$. Since any three distinct subsets I_i, I_j, I_ℓ have empty intersection, this yields a one-to-one correspondence

$$\phi : E(G) \longrightarrow \{x_s : s \in S\},$$

between the $(t - k)k$ edges of G and the $(t - 1)k$ variables with indices in S .

By Claim 3.4, the edges of G can be partitioned into k edge-disjoint spanning trees T_i , and $G = \bigcup_{i=0}^{k-1} T_i$.

Observe that for each $0 \leq i \leq k - 1$, \mathcal{C}_i has entries that are either zero or of the form x_s^i for some $x_s \in S$. We will show how to use the tree decomposition of G to choose nonzero entries in each \mathcal{C}_i so that (a) every row in the bottom part of M' is chosen exactly once, and (b) when the columns chosen are removed from M' , the resulting submatrix of \mathcal{B}_t is nonsingular. This will mean that the product of these non-zero entries appears in the determinant expansion of M' .

For each i , we pick $t - 1$ non-zero elements from each \mathcal{C}_i : we choose x_s^i for $x_s \in \{\phi(e) : e \in T_i\}$. That is, we consider all of the variables x_s corresponding to edges that appear in T_i . Let $m_i(x)$ denote the product of these entries:

$$m_i(x) = \prod_{x_s \in \{\phi(e) : e \in T_i\}} x_s^i.$$

Let $m(x) = \prod_{i=0}^{k-1} m_i(x)$. Since ϕ is a bijection, $m(x)$ is a product of $(t - 1)k$ distinct entries chosen from the submatrix $(\mathcal{C}_i : 0 \leq i \leq k - 1)$, and crucially, no two of them appear in the same row or column.

To conclude the proof, it is enough to show that $m(x)$ appears as a nonvanishing term in the determinant expansion of $M'_{k,(I_1,\dots,I_t)}$. Indeed, removing from $M'_{k,(I_1,\dots,I_t)}$ the $(t-1)k$ rows and columns that correspond to $m(x)$, the resulting submatrix is a block diagonal matrix

$$\text{diag}\left(\mathcal{B}'_t(i) : 0 \leq i \leq k-1\right),$$

where for each i , $\mathcal{B}'_t(i)$ is a square submatrix of \mathcal{B}_t of order $\binom{t-1}{2}$. By construction, each $\mathcal{B}'_t(i)$ is obtained by removing from \mathcal{B}_t a set of $t-1$ columns labeled by the spanning tree T_i . By Claim 3.5, this implies that $\mathcal{B}'_t(i)$ is nonsingular. Moreover, as each of the sets $I_i \cap I_j$ are disjoint due to the assumptions of the theorem, the monomial $m(x)$ appears only once in the determinant expansion of $M'_{k,(I_1,\dots,I_t)}$. Consequently, the “coefficient” of $m(x)$ in the determinant expansion of $M'_{k,(I_1,\dots,I_t)}$ is nonvanishing, completing the proof of the theorem. \square

4 Application to Perfect Hashing

In this section, we apply Theorem 3.1 to perfect hashing, and we prove Theorem 1.3.

Theorem (Theorem 1.3, restated). *Given integers $1 \leq k < n$ and $t \geq 3$, for a sufficiently large prime power q , there exists an $n \times q^k$ matrix, defined on the alphabet \mathbb{F}_q , such that any set of t columns is separated by at least $n - k(t-1) + 1$ rows.*

We will also show that Theorem 1.3 is optimal, at least within the class of *linear* hash matrices. Generalizing a definition of [BW98] (with a slightly different terminology), we say that an $n \times q^k$ matrix M is called *linear* if it is defined over the field \mathbb{F}_q and has the form $M = PQ$, where P is an $n \times k$ coefficient matrix and Q is the $k \times q^k$ matrix whose columns are formed by the q^k distinct vectors of \mathbb{F}_q^k .

With this terminology, we will prove the following proposition, which generalizes a result of [BW98] (see Theorem 4 of [BW98]).

Proposition 4.1. *If a linear $n \times q^k$ matrix separates any set of t columns by at least r rows, then $r \leq n - k(t-1) + 1$.*

Proposition 4.1 implies that the bound in Theorem 1.3 is tight, at least for linear constructions.

4.1 Proof of Theorem 1.3

Fix \mathbb{F} to be the finite field \mathbb{F}_q , and let us begin with an overview of the proof. Recall that an evaluation vector of \mathbb{F}_q^n is a vector whose coordinates are all distinct. It is well-known that an $[n, k]$ -RS code over \mathbb{F}_q^n is of size q^k , and that any two distinct codewords agree on at most $k-1$ coordinates. For our purpose, we view an $[n, k]$ -RS code as an $n \times q^k$ matrix whose columns are the codewords of the code. More precisely, the columns are all the vectors

$$\left\{ (f(\alpha_1), \dots, f(\alpha_n))^T, f \in \mathbb{F}_q[x], \deg(f) < k \right\}$$

with some arbitrary ordering, and $\alpha = (\alpha_1, \dots, \alpha_n)$ is the evaluation vector that defines the code. We say that the evaluation vector α *defines* the $n \times q^k$ matrix.

Fix an integer $t \geq 3$. An evaluation vector $\alpha \in \mathbb{F}_q^n$ is called “bad” if it does not define a strongly t -perfect hashing matrix. The main idea in the proof of Theorem 1.3 is to show that the number of bad evaluation vectors is at most $O_{n,t}(q^{n-1})$, whereas there are $\frac{q!}{(q-n)!} = \Theta_n(q^n)$ distinct evaluation

vectors. Therefore, for sufficiently large q there must exist an evaluation vector that is not bad, i.e., it defines an $n \times q^k$ strongly t -perfect hash matrix.

The main tool used in proving the upper bound on the number of bad evaluation vectors is the following well-known result.

Lemma 4.2 (DeMillo-Lipton-Schwartz-Zippel lemma, see, e.g., [Juk11] Lemma 16.3). *A nonzero polynomial $f \in \mathbb{F}_q[x_1, \dots, x_n]$ of degree d has at most dq^{n-1} zeros in \mathbb{F}_q^n .*

We need two more lemmas before presenting the proof of Theorem 1.3.

Lemma 4.3. *Given integers $1 \leq k < n$, $t \geq 3$, if an evaluation vector $\alpha \in \mathbb{F}_q^n$ does not define an $n \times q^k$ strongly t -perfect hashing matrix, then there exists an integer $s \in \{3, \dots, t\}$ and subsets $I_1, \dots, I_s \subseteq [n]$ such that*

- (i) $I_i \cap I_j \cap I_l = \emptyset$ for all $1 \leq i < j < l \leq s$;
- (ii) $\text{wt}(I_J) \leq (|J| - 1)k$ for any nonempty subset $J \subseteq [s]$;
- (iii) $\text{wt}(I_{[s]}) = (s - 1)k$;
- (iv) *the s -wise intersection matrix $M_{k, (I_1, \dots, I_s)}$ over \mathbb{F}_q is a nonsingular square matrix of order $k \binom{s}{2}$, whose determinant is a nonzero polynomial in $\mathbb{F}_q[x_1, \dots, x_n]$ of degree less than $k^2 t$.*

Proof. If an evaluation vector $\alpha \in \mathbb{F}_q^n$ is bad, then the $n \times q^k$ matrix it defines contains t distinct columns defined by polynomials f_1, \dots, f_t , which are separated by at most $n - k(t - 1)$ rows. Equivalently, there are at least $k(t - 1)$ rows that do not separate these t columns.

Next, we iteratively construct the sets $I_1, \dots, I_t \subseteq [n]$. We set all of them to be the empty set, and then for each row i that does not separate the t columns, we add i to arbitrary two sets I_j, I_l for which $f_j(\alpha_i) = f_l(\alpha_i)$. It is easy to verify that the sets I_j satisfy the following properties

- (a) $I_i \cap I_j \cap I_l = \emptyset$ for all $1 \leq i < j < l \leq t$;
- (b) $|I_i \cap I_j| \leq k - 1$ for distinct $i, j \in [t]$;
- (c) $\text{wt}(I_{[t]}) = \sum_{1 \leq i < j \leq t} |I_i \cap I_j| \geq k(t - 1)$.

Indeed, (a) follows from the definition of I_1, \dots, I_t , (b) follows from the property of RS codes, and (c) follows from (4) and (a).

Let s be the smallest positive integer for which there exist a subset $S \subseteq [t]$ of size s with $\text{wt}(I_S) \geq k(s - 1) > 0$. By (c) s is well-defined. Furthermore, as $\text{wt}(I_S) = 0$ for any $|S| = 1$ and $\text{wt}(I_S) < k$ for any $|S| = 2$, we have $3 \leq s \leq t$. Assume without loss of generality that $S = [s]$.

By construction and the minimality of s , the sets I_1, \dots, I_s satisfy properties (i) and (ii). We proceed to verify that also (iii) holds. Note that properties (i) and (ii) continue to hold if one removes an element from one of the sets I_j , and by doing so, the weight $\text{wt}(I_{[s]})$ can reduce by at most one. Hence, by iteratively removing elements from the sets I_j , one can construct sets, which we also denote by I_1, \dots, I_s , that satisfy property (iii), while retaining properties (i) and (ii).

Since the subsets $I_1, \dots, I_s \subseteq [n]$ satisfy the three assumptions of Theorem 3.1, it holds that $M_{k, (I_1, \dots, I_s)}$ is a nonsingular matrix. The claims on the order of the matrix and the degree of the polynomial are easy to verify, thereby completing the proof of (i)-(iv) \square

Lemma 4.4. *Let α be an evaluation vector that does not define an $n \times q^k$ strongly t -perfect hashing matrix, and let $M_{k, (I_1, \dots, I_s)}$ be the s -wise intersection matrix for $3 \leq s \leq t$ given by Lemma 4.3. Then, the matrix $M_{k, (I_1, \dots, I_s)}(\alpha)$, which is the evaluation of $M_{k, (I_1, \dots, I_s)}$ at α , does not have full rank.*

Proof. To prove the lemma it suffices to show that the matrix $M_{k,(I_1,\dots,I_s)}(\alpha)$ has a nontrivial kernel. Towards this end, let f_1, \dots, f_s be the s distinct polynomials that correspond to the set $S = [s]$ found in the proof of Lemma 4.3.

We view a polynomial of degree at most $k - 1$ also as a vector of length k defined by its k coefficients, where for $1 \leq i \leq k$, the i -th coordinate of the vector is the coefficient of the monomial x^{i-1} in that polynomial. Let $f_{ij} = f_i - f_j \in \mathbb{F}_q^k$ for $1 \leq i < j \leq s$, and let $f = (f_{ij} : 1 \leq i < j \leq s) \in \mathbb{F}_q^{\binom{s}{2}k}$, which is the concatenation of the vectors f_{ij} according to the lexicographic order on $\binom{[s]}{2}$ defined in Section 2.

We claim that $M_{k,(I_1,\dots,I_s)}(\alpha) \cdot f^T = 0$. Note that for any $1 \leq i < j \leq s$, we have

$$f_{ij} + f_{js} - f_{is} = 0. \quad (5)$$

Recall that the row vectors of \mathcal{B}_s correspond to the oriented triangles Δ_{ijs} . Then it follows from (5) and the definition of \mathcal{B}_s that

$$(\mathcal{B}_s \otimes \mathcal{I}_k) \cdot f^T = 0. \quad (6)$$

Moreover, observe that by definition, for any $l \in I_i \cap I_j$ we have $f_i(\alpha_l) = f_j(\alpha_l)$, which implies that $0 = f_i(\alpha_l) - f_j(\alpha_l) = (f_i - f_j)(\alpha_l) = f_{ij}(\alpha_l)$. Therefore $V_k(I_i \cap I_j) \cdot f_{ij}^T = 0$, which implies that

$$\text{diag}\left(V_k(I_i \cap I_j) : \{i, j\} \in \binom{[s]}{2}\right) \cdot f^T = 0. \quad (7)$$

Combining (6) and (7) we conclude that $M_{k,(I_1,\dots,I_s)}(\alpha) \cdot f^T = 0$, completing the proof of the claim. \square

We are now in a position to prove Theorem 1.3.

Proof of Theorem 1.3. We give an upper bound on the number of bad evaluation vectors that do not define a strongly t -perfect hash matrix.

Let \mathcal{M} be the set of s -wise intersection matrix $M_{k,(I_1,\dots,I_s)}$ that satisfy conditions (i)-(iv) of Lemma 4.3 for any $s \in \{3, \dots, t\}$. It is clear that any s -wise intersection matrix $M_{k,(I_1,\dots,I_s)}$ is completely determined by the subsets I_1, \dots, I_s , therefore the size of \mathcal{M} is at most 2^{nt} .

By Lemma 4.3, for any bad evaluation vector $\alpha \in \mathbb{F}_q^n$ there exists a matrix $M \in \mathcal{M}$ whose determinant is a nonzero polynomial in $\mathbb{F}_q[x_1, \dots, x_n]$ of degree less than k^2t . However, by Lemma 4.4 $\det(M)(\alpha) = 0$. Therefore, the set of bad evaluation vectors is contained in the union of the zero sets of the polynomials $\det(M), M \in \mathcal{M}$, which by Lemma 4.2, is of size at most

$$|\mathcal{M}| \cdot (k^2t)q^{n-1} \leq (2^{nt}k^2t)q^{n-1} = O_{n,t}(q^{n-1}).$$

The result follows by observing that the number of evaluation vectors, i.e., the number of vectors in \mathbb{F}_q^n with pairwise distinct coordinates is $\frac{q!}{(q-n)!} = \Theta_n(q^n)$. Hence, for sufficiently large q there exist many evaluation vectors that are not bad, and the result follows. \square

4.2 Proof of Proposition 4.1

Next, we prove Proposition 4.1, which implies that Theorem 1.3 is tight, at least for linear hash matrices.

Proof of Proposition 4.1. It is clear that any $n \times q^k$ matrix M given by an $[n, k]$ -RS code is linear, as we may write $M = V_k(\alpha_1, \dots, \alpha_n) \cdot Q$, where $\alpha = (\alpha_1, \dots, \alpha_n)$ is the evaluation vector, $V_k(\alpha_1, \dots, \alpha_n)$ is the associated $n \times k$ Vandermonde matrix as defined in (2), and Q is the $k \times q^k$ matrix whose columns are formed by the q^k distinct vectors of \mathbb{F}_q^k . The statement on the optimality follows from Theorem 4 of [BW98] which claims that any linear $N \times q^k$ matrix that has the property that any set of t columns is separated by at least one row, satisfies $N \geq k(t-1)$.

By this result, the restriction to the first $k(t-1) - 1$ rows of an $n \times q^k$ linear matrix, contains a set of t columns that are not separated at all. Hence, this set of columns is separated by at most $n - k(t-1) + 1$ rows of the $n \times q^k$ linear matrix, and the result follows. \square

5 Near-Optimal List-Recovery of RS Codes: Proof of the Main Theorem

In this section we prove our main theorem on the list-recovery of RS codes, Theorem 1.2. We will in fact prove the following theorem, which implies Theorem 1.2.

Theorem 5.1. *Let $k, n, L, \ell \in \mathbb{N}^+$, $\varepsilon \in (0, 1]$, and $\delta > 0$ be such that $L \geq (1 + \delta)\ell/\varepsilon - 1$ and*

$$k/n \leq \frac{\varepsilon}{c\sqrt{\ell}\left(\frac{1+\delta}{\delta}\right)\left(\log\left(\frac{1}{\varepsilon}\right) + \log\left(\frac{1+\delta}{\delta}\right) + 1\right)},$$

where $c > 0$ is the constant in Lemma 5.3. Consider the RS code

$$\mathcal{C} = \{(f(\alpha_1), \dots, f(\alpha_n)) : f(x) \in \mathbb{F}_q[x], \deg(f) < k\}$$

where $q \geq 2^{c'(L+n \log L)}$ for a large enough constant $c' > 0$ and $\alpha_1, \dots, \alpha_n$ are chosen uniformly and independently from \mathbb{F}_q at random. Then with high probability, the code \mathcal{C} has rate $R = k/n$ and is list-recoverable up to relative distance $1 - \varepsilon$ with input list size ℓ and output list size L . In particular, by choosing δ to be any positive constant, we could achieve $L = O(\ell/\varepsilon)$ and $R = \Omega\left(\frac{\varepsilon}{\sqrt{\ell}(\log(1/\varepsilon)+1)}\right)$.

We begin with an overview of the proof.

5.1 Overview of the Proof

We give an overview of our proof of Theorem 5.1. For simplicity, let us first assume the input list size ℓ equals one, i.e., we restrict to the case of list decoding. In this case, Theorem 5.1 states that there exist RS codes of rate $\Omega\left(\frac{\varepsilon}{\log(1/\varepsilon)+1}\right)$ that are list-decodable from radius $1 - \varepsilon$ with list size $O(1/\varepsilon)$.

As discussed previously, Conjecture 1.5 about the nonsingularity of intersection matrices would be enough to establish Theorem 5.1, and indeed an even stronger result. While we do not know if Conjecture 1.5 holds in general, Theorem 3.1 states that it holds under an extra condition that $I_i \cap I_{i'} \cap I_{i''} = \emptyset$ for distinct $i, i', i'' \in [t]$. Our proof of Theorem 5.1 is based on this theorem.

As Theorem 3.1 requires the above extra condition, which does not hold in general, we cannot simply follow the proof in [ST20a] and replace Conjecture 1.5 by Theorem 3.1. One naive way of fixing this is removing elements from the sets I_i until the condition $I_i \cap I_{i'} \cap I_{i''} = \emptyset$ for distinct $i, i', i'' \in [t]$ is satisfied. Specifically, for each $j \in [n]$ such that there exist more than two sets I_{i_1}, \dots, I_{i_s} containing j , we pick two sets (say I_{i_1} and I_{i_2}) and remove j from all the other sets. The resulting sets I'_1, \dots, I'_t satisfy the condition $I'_i \cap I'_{i'} \cap I'_{i''} = \emptyset$ for distinct $i, i', i'' \in [t]$ and we can now apply Theorem 3.1 to conclude that $M_{k, (I'_1, \dots, I'_t)}$ is nonsingular.

The problem with this idea, however, is that $\text{wt}(I'_{[t]})$ is generally much smaller than $\text{wt}(I_{[t]})$, possibly by a factor of $\Theta(t) = \Omega(1/\varepsilon)$. So in order to achieve $\text{wt}(I'_{[t]}) \geq (t-1)k$ as required by Theorem 3.1,⁵ we need to start with sets I_i such that $\text{wt}(I_{[t]}) \gg (t-1)k$. As a consequence, implementing this idea directly only yields RS codes of rate $\Omega(\varepsilon^2)$.

To mitigate this problem, we perform a random sampling of the collection $\{I_1, \dots, I_t\}$ before removing elements from I_i . Namely, we choose a random subset $J \subseteq [t]$ of some appropriate cardinality to be determined later. Then, we remove elements from the sets I_i just like before, but only for $i \in J$, so that the resulting sets I'_i satisfy the condition $I'_i \cap I'_{i'} \cap I'_{i''} = \emptyset$ for distinct $i, i', i'' \in J$. Finally, we apply Theorem 3.1 to conclude that the $|J|$ -wise intersection matrix $M_{k, (I'_i)_{i \in J}}$ is nonsingular, which can still be used to prove the list-decodability of the RS code.

The advantage of replacing $[t]$ by the random sample $J \subseteq [t]$ is that the condition $\text{wt}(I'_{[t]}) \geq (t-1)k$ is replaced by $\text{wt}(I'_J) \geq (|J|-1)k$. It turns out that the condition $I'_i \cap I'_{i'} \cap I'_{i''} = \emptyset$ for distinct $i, i', i'' \in J$ is easier to satisfy since $|J|$ may be much smaller than t . Consequently, we are able to show that there exist RS codes of rate $\Omega(\frac{\varepsilon}{\log(1/\varepsilon)+1})$ using this improved method.

Finally, we explain how to choose the cardinality of the sample J . Let $j \in [n]$ and denote by s_j the number of sets among I_1, \dots, I_t that contain j . Then for the index j , it is best to choose $|J| = \Theta(t/s_j)$. However, the number s_j may vary when j ranges over $[n]$, meaning that there may not be a single choice of $|J|$ that works best for all $j \in [n]$ simultaneously.

We solve this problem using the following trick: Create a logarithmic number of “buckets” and put $j \in [n]$ in the i -th bucket if $2^{i-1} \leq s_j < 2^i$. Then choose $|J|$ according to the heaviest bucket. Here, we lose a factor of $O(\log(1/\varepsilon)+1)$ in the rate because there are about $\log L = O(\log(1/\varepsilon)+1)$ buckets.

Generalization to list recovery. In the case of list decoding, we choose each set I_i to be the subset of coordinates where a codeword c_i and the received word y agree. In the more general setting of list recovery, there are multiple received words $y^{(1)}, \dots, y^{(\ell)}$ in the input list, so we need to keep track of multiple sets $I_i^{(1)}, \dots, I_i^{(\ell)}$ for each $i \in [t]$.

One way of extending our proof to list recovery is choosing $r \in [\ell]$ that maximizes $\text{wt}(I'_{[t]}^{(r)}) = \text{wt}(I_1^{(r)}, \dots, I_t^{(r)})$ and then proceeding as in the case of list decoding, with I_1, \dots, I_t replaced by $I_1^{(r)}, \dots, I_t^{(r)}$. It is not hard to show that this yields RS codes of rate $\Omega(\frac{\varepsilon}{\ell(\log(1/\varepsilon)+1)})$ which are list-recoverable from radius $1 - \varepsilon$ with input list size ℓ and output list size $O(\ell/\varepsilon)$.

With a more careful analysis, we show that we can achieve a better rate $\Omega(\frac{\varepsilon}{\sqrt{\ell}(\log(1/\varepsilon)+1)})$, as stated by Theorem 1.2. Our analysis is inspired by [LP20] which proved a similar result on the list-recoverability of randomly punctured codes with a different setting of parameters.

5.2 A Combinatorial Lemma

In this subsection, we state a combinatorial lemma (Lemma 5.3). It guarantees the existence of a subset $J \subseteq [t]$ and sets $I'_i \subseteq I_i$ for $i \in J$ that satisfy certain conditions, particularly the condition $I'_i \cap I'_{i'} \cap I'_{i''} = \emptyset$ for distinct $i, i', i'' \in J$. We then use this lemma together with Theorem 3.1 to prove Theorem 1.2. The proof of this combinatorial lemma is postponed to Subsection 5.4.

First, we need the following generalization of the weight function $\text{wt}(\cdot)$.

Definition 5.2 (Generalized weight function). *Let $n, t \in \mathbb{N}^+$ and $I_1, \dots, I_t \subseteq [n]$. Let $S_j = \{i \in [t] : j \in I_i\}$ for $j \in [n]$. For $J \subseteq [t]$ and $\ell \in \mathbb{N}^+$, define the ℓ -th generalized weight $\text{wt}_\ell(I_J)$ of I_J to*

⁵Theorem 3.1 requires the stronger condition $\text{wt}(I'_{[t]}) = (t-1)k$, but this can be achieved by further removing elements from the sets I'_i .

be

$$\text{wt}_\ell(I_J) := \sum_{j=1}^n \max\{|S_j \cap J| - \ell, 0\}.$$

Note that $\text{wt}(I_J) = \sum_{i \in J} |I_i| - |\bigcup_{i \in J} I_i| = \text{wt}_1(I_J)$. Also note that

$$\text{wt}_\ell(I_J) \geq \sum_{j=1}^n (|S_j \cap J| - \ell) = \sum_{i \in J} |I_i| - \ell n. \quad (8)$$

The proof of Theorem 1.2 uses the following combinatorial lemma, which we prove in the next subsection.

Lemma 5.3. *Let $k, n, t, \ell \in \mathbb{N}^+$, $\varepsilon \in (0, 1]$, $\delta > 0$, and $I_1^{(r)}, \dots, I_t^{(r)} \subseteq [n]$ for $r \in [\ell]$. Let $I_i = \bigcup_{r=1}^\ell I_i^{(r)}$ for $i \in [t]$. Suppose $t \geq (1 + \delta)\ell/\varepsilon$, $|I_i| \geq \varepsilon n$ for $i \in [t]$, and*

$$\text{wt}_\ell(I_{[t]}) \geq \left(c\sqrt{\ell} \left(\log\left(\frac{1}{\varepsilon}\right) + \log\left(\frac{1+\delta}{\delta}\right) + 1 \right) \right) \cdot tk.$$

where $c > 0$ is a large enough absolute constant. Then there exist $J \subseteq [t]$ and a collection $(I'_i)_{i \in J}$ of subsets of $[n]$ indexed by J such that $|J| \geq 2$, $I'_i \subseteq I_i$ for $i \in J$, and the following conditions are satisfied:

- (1) $I'_i \cap I'_{i'} \cap I'_{i''} = \emptyset$ for distinct $i, i', i'' \in J$.
- (2) $\text{wt}(I'_{J'}) \leq (|J'| - 1)k$ for all nonempty $J' \subseteq J$.
- (3) $\text{wt}(I'_J) = (|J| - 1)k$.
- (4) For every $j \in [n]$, there exists $r_j \in [\ell]$ such that $\{i \in J : j \in I'_i\} \subseteq \{i \in J : j \in I_i^{(r_j)}\}$.

Remark 5.4. Condition (4) is introduced for list recovery. For the case $\ell = 1$, which corresponds to list decoding, Condition (4) is automatically satisfied by choosing $r_j = 1$ for $j \in [n]$ since in this case $I'_i \subseteq I_i = I_i^{(1)}$ for $i \in J$.

We also need the following lemma that bounds the number of pairs $(J, (I'_i)_{i \in J})$.

Lemma 5.5. *The number of $(J, (I'_i)_{i \in J})$ satisfying Condition (1) of Lemma 5.3 is at most $2^t(1 + t + \binom{t}{2})^n$.*

Proof. There are at most 2^t choices of J . Now fix $J \subseteq [t]$. For $j \in [n]$, let $T_j = \{i \in J : j \in I'_i\}$. Note that we have $|T_j| \leq 2$ for all $j \in [n]$ by Condition (1) of Lemma 5.3. So for each $j \in [n]$, the number of choices of T_j is at most $1 + t + \binom{t}{2}$. Also note that the sets I'_i are determined by the sets T_j by $I'_i = \{j \in [n] : i \in T_j\}$. So the number of choices of $(J, (I'_i)_{i \in J})$ is at most $2^t(1 + t + \binom{t}{2})^n$. \square

5.3 Proof of Theorem 5.1

Now we are ready to prove our main theorem. For the reader's convenience, we restate it below.

Theorem (Theorem 5.1, restated). *Let $k, n, L, \ell \in \mathbb{N}^+$, $\varepsilon \in (0, 1]$, and $\delta > 0$ such that $L \geq (1 + \delta)\ell/\varepsilon - 1$ and*

$$k/n \leq \frac{\varepsilon}{c\sqrt{\ell} \left(\frac{1+\delta}{\delta} \right) \left(\log\left(\frac{1}{\varepsilon}\right) + \log\left(\frac{1+\delta}{\delta}\right) + 1 \right)},$$

where $c > 0$ is the constant in Lemma 5.3. Consider the RS code

$$\mathcal{C} = \{(f(\alpha_1), \dots, f(\alpha_n)) : f(x) \in \mathbb{F}_q[x], \deg(f) < k\}$$

where $q \geq 2^{c'(L+n \log L)}$ for a large enough constant $c' > 0$ and $\alpha_1, \dots, \alpha_n$ are chosen uniformly and independently from \mathbb{F}_q at random. Then with high probability, the code \mathcal{C} has rate $R = k/n$ and is list-recoverable up to relative distance $1 - \varepsilon$ with input list size ℓ and output list size L . In particular, by choosing δ to be any positive constant, we could achieve $L = O(\ell/\varepsilon)$ and $R = \Omega\left(\frac{\varepsilon}{\sqrt{\ell(\log(1/\varepsilon)+1)}}\right)$.

Proof. Let $t = L + 1$. Consider the following two conditions:

- (1) $\alpha_i \neq \alpha_j$ for all distinct $i, j \in [n]$.
- (2) For all $J \subseteq [t]$ and $(I'_i)_{i \in J}$ satisfying Conditions (1)–(3) of Lemma 5.3, we have

$$\det(M_{k, (I'_i)_{i \in J}}(\alpha_1, \dots, \alpha_n)) \neq 0,$$

where $M_{k, (I'_i)_{i \in J}}$ denotes the $\binom{|J|}{2}k \times \binom{|J|}{2}k$ variable matrix⁶

$$M_{k, (I'_i)_{i \in J}} = \left(\frac{\mathcal{B}_{|J|} \otimes \mathcal{I}_k}{\text{diag}\left(V_k(I'_i \cap I'_j) : \{i, j\} \in \binom{J}{2}\right)} \right).$$

The first condition is satisfied with probability at least $1 - \binom{n}{2}/q$. For the second condition, consider fixed $J \subseteq [t]$ and $(I'_i)_{i \in J}$ satisfying Conditions (1)–(3) of Lemma 5.3. We know $\det(M_{k, (I'_i)_{i \in J}}) \neq 0$ by Theorem 3.1. Also note that $\det(M_{k, (I'_i)_{i \in J}})$ is a multivariate polynomial of total degree at most $(|J| - 1)k(k - 1) \leq Lk^2$. So by Lemma 4.2, $\det(M_{k, (I'_i)_{i \in J}})(\alpha_1, \dots, \alpha_n) \neq 0$ holds with probability at least $1 - Lk^2/q$ for fixed J and $(I'_i)_{i \in J}$. The number of choices of $(J, (I'_i)_{i \in J})$ is at most $2^t(1 + t + \binom{t}{2})^n$ by Lemma 5.5. By the union bound, the two conditions are simultaneously satisfied with probability at least

$$1 - \binom{n}{2}/q - 2^t \left(1 + t + \binom{t}{2}\right)^n Lk^2/q = 1 - o(1)$$

over the random choices of $\alpha_1, \dots, \alpha_n$, where we use the assumption that $q \geq 2^{c'(L+n \log L)}$ and $c' > 0$ is a large enough constant.

Fix $\alpha_1, \dots, \alpha_n \in \mathbb{F}_q$ that satisfy the above two conditions. By the first condition, the code \mathcal{C} has rate exactly k/n . It remains to show that \mathcal{C} is list-recoverable up to relative distance $1 - \varepsilon$ with input list size ℓ and output list size L . Assume to the contrary that this does not hold. Then there exist t distinct polynomials $f_1, \dots, f_t \in \mathbb{F}_q[x]$ of degree less than k and ℓ received words $y^{(r)} = (y_1^{(r)}, \dots, y_n^{(r)}) \in \mathbb{F}_q^n$, where $r = 1, 2, \dots, \ell$, such that for all $i \in [t]$, the cardinality of the set

$$I_i := \{j \in [n] : \text{there exists } r \in [\ell] \text{ such that } f_i(\alpha_j) = y_j^{(r)}\}$$

is at least εn .

⁶The number of rows of $M_{k, (I'_i)_{i \in J}}$ is $\binom{|J|-1}{2}k + \sum_{\{i, j\} \in \binom{J}{2}} |I'_i \cap I'_j|$, which equals $\binom{|J|-1}{2}k + \sum_{i \in J} |I'_i| - |\bigcup_{i \in J} I'_i| = \binom{|J|-1}{2}k + \text{wt}(I'_j)$ by Condition (1) of Lemma 5.3. This number further equals $\binom{|J|-1}{2}k + (|J| - 1)k = \binom{|J|}{2}k$ by Condition (3) of Lemma 5.3.

Let $I_i^{(r)} = \{j \in [n] : f_i(\alpha_j) = y_j^{(r)}\}$ for $i \in [t]$ and $r \in [\ell]$, i.e., $I_i^{(r)}$ denotes the set of coordinates where $C(f_i) := (f_i(\alpha_1), \dots, f_i(\alpha_n))$ and $y^{(r)}$ agree. So $I_i = \bigcup_{r=1}^{\ell} I_i^{(r)}$ for $i \in [t]$. As $t = L + 1 \geq (1 + \delta)\ell/\varepsilon$ and $k/n \leq \frac{\varepsilon}{c\sqrt{\ell}(\frac{1+\delta}{\delta})(\log(\frac{1}{\varepsilon}) + \log(\frac{1+\delta}{\delta} + 1))}$, we also have

$$\text{wt}_{\ell}(I_{[t]}) \stackrel{(8)}{\geq} t\varepsilon n - \ell n \geq \frac{\delta}{1 + \delta} \cdot t\varepsilon n \geq \left(c\sqrt{\ell} \left(\log \left(\frac{1}{\varepsilon} \right) + \log \left(\frac{1 + \delta}{\delta} \right) + 1 \right) \right) \cdot tk.$$

By Lemma 5.3, there exist $J \subseteq [t]$ and $(I'_i)_{i \in J}$ such that $|J| \geq 2$, $I'_i \subseteq I_i$ for $i \in J$, and Conditions (1)–(4) of Lemma 5.3 are satisfied.

Let u be the vector $(f_{ij} : \{i, j\} \in \binom{J}{2}, i < j) \in \mathbb{F}_q^{\binom{|J|}{2}k}$, where $f_{ij} := f_i - f_j$, as defined in the proof of Lemma 4.4. As $|J| \geq 2$ and f_1, \dots, f_t are distinct, we have $u \neq 0$.

We claim that

$$M_{k, (I'_i)_{i \in J}}(\alpha_1, \dots, \alpha_n) \cdot u^T = 0. \quad (9)$$

To see this, first note that $(\mathcal{B}_{|J|} \otimes \mathcal{I}_k) \cdot u^T = 0$ (cf. (6) in the proof of Lemma 4.4). Now consider a row v of the submatrix

$$\text{diag} \left(V_k(I'_i \cap I'_j) : \{i, j\} \in \binom{J}{2} \right) (\alpha_1, \dots, \alpha_n),$$

of $M_{k, (I'_i)_{i \in J}}(\alpha_1, \dots, \alpha_n)$, which corresponds to some $\{i, j\} \in \binom{J}{2}$ with $i < j$ and $s \in I'_i \cap I'_j$. By definition, we have $v \cdot u^T = f_{ij}(\alpha_s)$, i.e., the row v represents the linear constraint $f_{ij}(\alpha_s) = 0$. By Condition (4) of Lemma 5.3, we have $s \in I_i^{(r_s)} \cap I_j^{(r_s)}$ for some $r_s \in [\ell]$, which implies $f_i(\alpha_s) = y_s^{(r_s)}$ and $f_j(\alpha_s) = y_s^{(r_s)}$. So $v \cdot u^T = f_{ij}(\alpha_s) = f_i(\alpha_s) - f_j(\alpha_s) = 0$. This proves (9).

By (9), we have $\det(M_{k, (I'_i)_{i \in J}})(\alpha_1, \dots, \alpha_n) = 0$. But this contradicts the choice of $\alpha_1, \dots, \alpha_n$. \square

5.4 Proof of Lemma 5.3

We present the proof of Lemma 5.3 in this subsection.

Let $k, n, t, \ell \in \mathbb{N}^+$, $\varepsilon \in (0, 1]$, $\delta > 0$, and the sets $I_i^{(r)}$, $I_i \subseteq [n]$ for $i \in [t]$ and $r \in [\ell]$ be as in Lemma 5.3. That is, we have $t \geq (1 + \delta)\ell/\varepsilon$, $I_i = \bigcup_{r=1}^{\ell} I_i^{(r)}$ and $|I_i| \geq \varepsilon n$ for $i \in [t]$, and

$$\text{wt}_{\ell}(I_{[t]}) \geq \left(c\sqrt{\ell} \left(\log \left(\frac{1}{\varepsilon} \right) + \log \left(\frac{1 + \delta}{\delta} \right) + 1 \right) \right) \cdot tk. \quad (10)$$

where $c > 0$ is a large enough absolute constant. We may assume without loss of generality that $I_i^{(1)}, \dots, I_i^{(\ell)}$ are pairwise disjoint for all $i \in [t]$: if an element appears in both $I_i^{(r)}$ and $I_i^{(r')}$ for $r \neq r'$, we can remove it from one of them, and the set I_i does not change. Thus, if we can prove Lemma 5.3 when these pairwise disjoint conditions hold, we can prove it in general, since we can choose the same subsets $I'_i \subseteq I_i$ after removing redundant elements.

For $j \in [n]$, $S_j := \{i \in [t] : j \in I_i\}$. By definition, we have

$$\text{wt}_{\ell}(I_{[t]}) = \sum_{j=1}^n \max\{|S_j| - \ell, 0\}. \quad (11)$$

Assume for a moment that there exists an integer $K \in \mathbb{N}^+$ such that $\max\{|S_j| - \ell, 0\}$ equals either K or zero for all $j \in [n]$. Then by (11), the number of $j \in [n]$ for which $\max\{|S_j| - \ell, 0\} = K$ holds (or equivalently, $|S_j| = K + \ell$ holds) is precisely $\text{wt}(I_{[t]})/K$. The next lemma extends this fact to the general case with only logarithmic loss.

Lemma 5.6. *There exists an integer $K > 0$ such that the number of $j \in [n]$ satisfying $|S_j| \geq K + \ell$ is at least $\frac{\text{wt}_\ell(I_{[t]})}{c_0 K (\log(\frac{1}{\varepsilon}) + \log(\frac{1+\delta}{\delta}) + 1)}$, where $c_0 > 0$ is some absolute constant.*

Proof. By (8) and the fact that $t \geq (1 + \delta)\ell/\varepsilon$, we have

$$\text{wt}_\ell(I_{[t]}) \geq t\varepsilon n - \ell n \geq \frac{\delta}{1 + \delta} \cdot t\varepsilon n. \quad (12)$$

For $i = 0, 1, 2, \dots$, let $B_i = \{j \in [n] : 2^i \leq |S_j| - \ell < 2^{i+1}\}$. Then

$$\text{wt}_\ell(I_{[t]}) = \sum_{j=1}^n \max\{|S_j| - \ell, 0\} = \sum_{i=0}^{\lceil \log t \rceil - 1} \sum_{j \in B_i} (|S_j| - \ell).$$

Let $d = \lfloor \log(\frac{\delta}{1+\delta} \cdot t\varepsilon/2) \rfloor$. Note that d could be negative (possibly $\frac{\delta}{1+\delta} \cdot t\varepsilon/2 \in (0, 1)$). Then

$$\sum_{0 \leq i < d} \sum_{j \in B_i} (|S_j| - \ell) \leq n2^d \leq \frac{\delta}{1 + \delta} \cdot t\varepsilon n/2 \stackrel{(12)}{\leq} \text{wt}_\ell(I_{[t]})/2.$$

Therefore

$$\sum_{i=\max\{d, 0\}}^{\lceil \log t \rceil - 1} \sum_{j \in B_i} (|S_j| - \ell) \geq \text{wt}_\ell(I_{[t]})/2. \quad (13)$$

Let $\Delta = \lceil \log t \rceil - \max\{d, 0\} = O(\log(\frac{1}{\varepsilon}) + \log(\frac{1+\delta}{\delta}) + 1)$. By (13), there exists an integer i_0 such that $\max\{d, 0\} \leq i_0 \leq \lceil \log t \rceil - 1$ and

$$\sum_{j \in B_{i_0}} (|S_j| - \ell) \geq \frac{\text{wt}_\ell(I_{[t]})}{2\Delta}. \quad (14)$$

Choose $K = 2^{i_0}$. Then $K \leq |S_j| - \ell < 2K$ for all $j \in B_{i_0}$. The upper bound $|S_j| - \ell < 2K$ for $j \in B_{i_0}$, together with (14), implies $|B_{i_0}| \geq \frac{\text{wt}_\ell(I_{[t]})}{4K\Delta}$. So the number of $j \in [n]$ satisfying $|S_j| \geq K + \ell$ is at least $\frac{\text{wt}_\ell(I_{[t]})}{4K\Delta} = \Omega\left(\frac{\text{wt}_\ell(I_{[t]})}{K(\log(\frac{1}{\varepsilon}) + \log(\frac{1+\delta}{\delta}) + 1)}\right)$. \square

Fix K satisfying Lemma 5.6. Define

$$A := \{j \in [n] : |S_j| \geq K + \ell\} \subseteq [n].$$

By the choice of K and Lemma 5.6, we have

$$|A| \geq \frac{\text{wt}_\ell(I_{[t]})}{c_0 K (\log(\frac{1}{\varepsilon}) + \log(\frac{1+\delta}{\delta}) + 1)}. \quad (15)$$

For $j \in [n]$ and $r \in [\ell]$, let $S_j^{(r)} := \{i \in [t] : j \in I_i^{(r)}\}$. So $S_j = \bigcup_{r=1}^{\ell} S_j^{(r)}$ for $j \in [n]$. Note that $S_j^{(1)}, \dots, S_j^{(\ell)}$ are pairwise disjoint for all j : if $i \in S_j^{(r)} \cap S_j^{(r')}$, we must have $j \in I_i^{(r)} \cap I_i^{(r')}$, but we have assumed that $I_i^{(1)}, \dots, I_i^{(\ell)}$ are pairwise disjoint for all i .

We also need the following technical lemma.

Lemma 5.7. *For real numbers $p \in (0, \frac{1}{2}]$ and $x \geq 0$, we have $(1-p)^x(1+px) \leq 1 - \frac{1}{8}p^2x^2$ if $x \leq \frac{1}{p}$, and $(1-p)^x(1+px) \leq \frac{2}{e}$ otherwise.*

Proof. Fix p and let $f(y) = (1-p)^y(1+py)$. For $y \geq 0$, the derivative $f'(y)$ satisfies

$$\begin{aligned} f'(y) &= (1-p)^y(\ln(1-p) \cdot (1+py) + p) \\ &\leq (1-p)^y(-p(1+py) + p) \\ &= -p^2y(1-p)^y. \end{aligned} \tag{16}$$

So $f'(y) \leq -p^2y(1-p)^x$ for $y \in [0, x]$. As $f(0) = 1$, we have $f(x) \leq 1 + \int_0^x -p^2y(1-p)^x dy = 1 - \frac{1}{2}p^2x^2(1-p)^x$. If $x \leq \frac{1}{p}$, we have $(1-p)^x \geq (1-p)^{1/p} \geq 1/4$, as $(1-p)^{1/p}$ is decreasing with p , and thus is minimized at $p = \frac{1}{2}$. Hence, we have $f(x) \leq 1 - \frac{1}{8}p^2x^2$. By (16), $f(y)$ is decreasing and thus maximized at $y = 1/p$ on the interval $[1/p, \infty)$, so for $x \geq 1/p$, we have $f(x) \leq f(1/p) = (1-p)^{1/p}(1+1) \leq \frac{2}{e}$. \square

The above lemma is used to prove the following statement.

Lemma 5.8. *Choose a random subset $J \subseteq [t]$ by independently including each $i \in [t]$ in J with probability $p = \min\{\sqrt{\ell}/(2K), \frac{1}{2}\}$. Let*

$$A_J := \{j \in A : \text{there exists } r \in [\ell] \text{ such that } |S_j^{(r)} \cap J| \geq 2\}.$$

Then $\mathbb{E}[|A_J|] = \Omega(|A|)$.

Proof. Fix $j \in A$. It suffices to prove that $\Pr[j \in A_J] \geq c$ for some constant c . Let

$$t_r := \max\{|S_j^{(r)}| - 1, 0\}$$

for $r = 1, \dots, \ell$. Let $K' := \sum_{r=1}^{\ell} t_j$. Since $j \in A$ and $S_j = \bigcup_{r=1}^{\ell} S_j^{(r)}$, we have

$$K' = \sum_{r=1}^{\ell} t_j \geq |S_j| - \ell \geq K.$$

For all $r = 1, \dots, \ell$, we have

$$\Pr[|S_j^{(r)} \cap J| \leq 1] = (1-p)^{t_r}(1+t_r p)$$

This is because the probability is exactly $(1-p)^{t_r+1} + (t_r+1)p(1-p)^{t_r}$ when $t_r \geq 1$, and is exactly 1 when $t_r = 0$.

As $S_j^{(1)}, \dots, S_j^{(\ell)}$ are disjoint, the events that $|S_j^{(r)} \cap J| \geq 2$ are independent. Thus, the probability that $j \in A_J$ is

$$\begin{aligned} \Pr[j \in A_J] &= 1 - \prod_{r=1}^{\ell} \Pr[|S_j^{(r)} \cap J| \leq 1] = 1 - \prod_{r=1}^{\ell} (1-p)^{t_r}(1+t_r p) \\ &= 1 - (1-p)^{K'} \prod_{r=1}^{\ell} (1+t_r p). \end{aligned}$$

We now bound this below by a constant. First consider the case $K' \leq \ell$. Then $\ell \geq K$ and $p \geq \frac{1}{2\sqrt{K}} \geq \frac{1}{2\sqrt{K'}}$. When x_1, \dots, x_{ℓ} are constrained to be nonnegative integers with a fixed sum, if there exists $x_i \leq x_j - 2$, we can strictly increase the product $f(x_1, \dots, x_{\ell}) := \prod_{r=1}^{\ell} (1+x_r p)$ by

replacing x_i with $x_i + 1$ and x_j with $x_j - 1$. Thus, the maximum value of $f(x_1, \dots, x_\ell)$ occurs when K' of the x_i are 1 and the rest are zero. Hence, we have

$$\Pr[j \in A_J] \geq 1 - (1-p)^{K'}(1+p)^{K'} = 1 - (1-p^2)^{K'} \geq 1 - \left(1 - \frac{1}{4K'}\right)^{K'} \geq 1 - e^{-1/4},$$

as desired.

Now suppose $K' > \ell$. Note that $p = \min\{\sqrt{\ell}/(2K), \frac{1}{2}\} \geq \sqrt{\ell}/(2K')$. As $\log(1+xp)$ is concave for nonnegative real numbers x , we have that $f(x_1, \dots, x_\ell) = \prod_{r=1}^{\ell}(1+x_r p)$ subject to $x_1 + \dots + x_\ell = K'$ is maximized when all the x_i 's are equal. Hence,

$$\begin{aligned} \Pr[j \in A_J] &= 1 - (1-p)^{K'} \prod_{r=1}^{\ell} (1+t_r p) \geq 1 - (1-p)^{K'} \left(1 + \frac{K'}{\ell} p\right)^{\ell} \\ &= 1 - \left(\left((1-p)^{K'/\ell}\right) \left(1 + \frac{K'}{\ell} p\right)\right)^{\ell} \end{aligned}$$

as desired. If $x := K'/\ell \leq 1/p$, then, by Lemma 5.7, we have

$$\Pr[j \in A_J] \geq 1 - \left(1 - p^2 \frac{(K')^2}{8\ell^2}\right)^{\ell} \geq 1 - \left(1 - \frac{1}{32\ell}\right)^{\ell} \geq 1 - e^{-1/32}$$

where the second inequality uses the fact $p \geq \sqrt{\ell}/(2K')$. If $x \geq 1/p$, then by Lemma 5.7, we have $\Pr[j \in A_J] \geq 1 - (2/e)^{\ell} \geq 1 - 2/e$. In all cases, $\Pr[j \in A_J]$ is bounded below by a constant, as desired. \square

Corollary 5.9. *There exists $J \subseteq [t]$ of cardinality at most $c_1 \sqrt{\ell t}/K$ such that the cardinality of the set A_J as defined in Lemma 5.8 is at least $c_2 |A|$, where $c_1, c_2 > 0$ are absolute constants.*

Proof. Choose a random set $J \subseteq [t]$ as in Lemma 5.8. Then $\mathbb{E}[|A_J|] = \Omega(|A|)$ by Lemma 5.8. As $|A_J| \leq |A|$, we have $\Pr[|A_J| \geq c_2 |A|] \geq c_3$ for some absolute constants $c_2, c_3 > 0$.

Observe that by Lemma 5.8 and the linearity of expectation we have $\mathbb{E}[|J|] = pt = O(\sqrt{\ell t}/K)$. Moreover, by Markov's inequality we have $\Pr[|J| > c_1 \sqrt{\ell t}/K] \leq c_3/2$ for some sufficiently large constant $c_1 > 0$. By the union bound, we know the conditions $|J| \leq c_1 \sqrt{\ell t}/K$ and $|A_J| \geq c_2 |A|$ are simultaneously satisfied with probability at least $c_3/2 > 0$, so there exists $J \subseteq [t]$ that satisfies these two conditions. \square

Fix $J \subseteq [t]$ as in Corollary 5.9, so that $|J| \leq c_1 \sqrt{\ell t}/K$ and $|A_J| \geq c_2 |A|$. As the constant c in (10) is large enough, we may assume $c \geq c_0 c_1 / c_2$, where c_0 is as in (15). Then we have

$$|A_J| \geq c_2 |A| \stackrel{(15)}{\geq} c_2 \cdot \frac{\text{wt}_{\ell}(I_{[t]})}{c_0 K (\log(\frac{1}{\varepsilon}) + \log(\frac{1+\delta}{\delta}) + 1)} \stackrel{(10)}{\geq} (c_1 \sqrt{\ell t}/K) k > (|J| - 1)k. \quad (17)$$

For each $j \in A_J$, choose a subset $T_j \subseteq S_j \cap J$ and an index $r_j \in [\ell]$ such that $|T_j| = 2$ and $T_j \subseteq S_j^{(r_j)}$. This is possible by the definition of A_J in Lemma 5.8. For $j \in [n] \setminus A_J$, let $T_j = \emptyset$. So for $j \in [n]$, we have

$$|T_j| = \begin{cases} 2 & j \in A_J, \\ 0 & j \notin A_J. \end{cases}$$

Let $I'_i = \{j \in [n] : i \in T_j\} \subseteq I_i$ for $i \in J$. We have

$$\text{wt}(I'_J) = \sum_{j=1}^n \max\{|T_j| - 1, 0\} = |A_J| \stackrel{(17)}{\geq} (|J| - 1)k.$$

Moreover, the fact $|T_j| \leq 2$ for $j \in [n]$ implies that $I'_i \cap I'_{i'} \cap I'_{i''} = \emptyset$ for distinct $i, i', i'' \in J$, by noting that $T_j = \{i \in J : j \in I'_i\}$.

For $j \in A_J$, we have

$$\{i \in J : j \in I'_i\} = T_j \subseteq S_j^{(r_j)} \cap J = \{i \in J : j \in I_i^{(r_j)}\}.$$

And for $j \in [n] \setminus A_J$, we have

$$\{i \in J : j \in I'_i\} = T_j = \emptyset \subseteq \{i \in J : j \in I_i^{(r)}\} \text{ for any } r \in [\ell].$$

Finally, we have $|J| \geq 2$ as $|A_J| \geq c_2|A| > 0$. To summarize, we have proved the following weaker version of Lemma 5.3.

Lemma 5.10. *Under the assumption of Lemma 5.3, there exist $J \subseteq [t]$ and a collection $(I'_i)_{i \in J}$ of subsets of $[n]$ such that $|J| \geq 2$, $I'_i \subseteq I_i$ for $i \in J$, and the following conditions are satisfied:*

- (1) $I'_i \cap I'_{i'} \cap I'_{i''} = \emptyset$ for distinct $i, i', i'' \in J$.
- (2) $\text{wt}(I'_J) \geq (|J| - 1)k$.
- (3) For every $j \in [n]$, there exists $r_j \in [\ell]$ such that $\{i \in J : j \in I'_i\} \subseteq \{i \in J : j \in I_i^{(r_j)}\}$.

Now we are ready to prove Lemma 5.3.

Proof of Lemma 5.3. Choose the sets J and $(I'_i)_{i \in J}$ satisfying Lemma 5.10 such that $|J| \geq 2$ is minimized. Note that removing one element from I'_i for some $i \in J$ preserves (1) and (3) of Lemma 5.10 and reduces $\text{wt}(I'_J)$ by at most one. Removing elements from the sets in $(I'_i)_{i \in J}$ one by one until $\text{wt}(I'_J) = (|J| - 1)k$ holds. Then J and $(I'_i)_{i \in J}$ satisfy (1), (3), and (4) of Lemma 5.3.

The minimality of $|J|$ guarantees that $\text{wt}(I'_{J'}) \leq (|J'| - 1)k$ for all nonempty $J' \subseteq J$. (When $|J'| = 1$, this holds since $\text{wt}(I'_{J'}) = 0$.) So J and $(I'_i)_{i \in J}$ satisfy (2) of Lemma 5.3 as well. \square

6 Towards Conjecture 1.4: A Hypergraph Nash-Williams–Tutte Conjecture

Recall that Conjecture 1.4 states that RS codes of rate R are list-decodable from radius $1 - R - \varepsilon$ with list size at most $\lceil \frac{1-R-\varepsilon}{\varepsilon} \rceil$. As discussed in the introduction, it was shown in [ST20a, Theorem 5.8] that resolving Conjecture 1.5 (about the non-singularity of intersection matrices) would resolve Conjecture 1.4 (about list-decoding).

Our approach above was to show in Theorem 3.1 that intersection matrices are nonsingular under the additional assumption that $I_i \cap I_j \cap I_\ell = \emptyset$ for all $1 \leq i < j < \ell \leq t$, and then use that to conclude our main result about list-recovery. However, to prove Conjecture 1.4 in full, we need to remove the additional three-wise intersection assumption. In this section, we describe an approach that could potentially resolve Conjecture 1.4 in full. To do so, we conjecture a hypergraph generalization of the Nash-Williams–Tutte theorem [NW61, Tut61] that may be of independent interest, and prove that this conjecture implies Conjecture 1.4. Indeed, one might hope that such a generalization would be useful for Conjecture 1.4, because the Nash-Williams–Tutte theorem has been instrumental in proving several of the results in this paper, including Theorem 1.1, Theorem 1.2, Theorem 3.1, and Theorem 1.3. Along the way, we give a second proof of our main list decoding theorem, Theorem 1.1, by combining this approach with known results on hypergraph packings [CS07, CCV09]. In Section 6.1, we describe our conjecture, Conjecture 6.1, and in Section 6.2, we prove that it implies the optimal list decoding conjecture, Conjecture 1.4.

6.1 A Hypergraph Nash-Williams–Tutte Conjecture

In this section we state a conjecture (Conjecture 6.1 below), and prove that it implies our main goal Conjecture 1.4. We are not able to prove Conjecture 6.1, but give some evidence for it, pointing out that special cases and relaxations are known to be true.

Throughout, we use t as the number of vertices in a (hyper)graph. This variable corresponds to the same t used in t -wise intersection matrices. A (multi)graph G is called *k -partition-connected* if every partition \mathcal{P} of the vertex set has at least $k(|\mathcal{P}| - 1)$ edges crossing the partition. By the Nash-Williams–Tutte theorem, this is equivalent to the graph having k edge-disjoint spanning trees. The parameter k here is the same k used as the dimension of the Reed–Solomon code and the same k used for the Vandermonde matrix degrees in the intersection matrices.

We say a hypergraph H is *k -weakly-partition-connected*⁷ if, for every partition \mathcal{P} of the vertices of H , we have

$$\sum_{e \in E(H)} (\mathcal{P}(e) - 1) \geq k(|\mathcal{P}| - 1), \quad (18)$$

where $\mathcal{P}(e)$ is the number of parts of \mathcal{P} that e intersects. For example, any k -partition-connected graph is k -weakly-partition-connected as a hypergraph. As another example, k copies of a hyperedge covering all t vertices of H is also k -weakly partition-connected.

An *edge-labeled* graph is a graph G where each edge is assigned a label from some set E . Let H be a hypergraph. A *tree-assignment* of H is an edge-labeled graph G obtained by replacing each edge e of H with a tree F_e of $|e| - 1$ edges on the vertices of e . Furthermore, each edge of the graph F_e is labeled with e . The graph G is thus the union of the graphs F_e for $e \in H$.

A *k -tree-decomposition* of a graph on $k(t - 1)$ edges is a partition of its edges into k edge-disjoint spanning trees T_0, \dots, T_{k-1} . We say *tree-decomposition* when k is understood. In an edge-labeled graph T with edge-labels from some set E , let $v^T \in \mathbb{N}^E$ be the vector counting the edge-labels in T . Specifically, v_e^T is the number of edges of label e in T . For a tree-decomposition (T_0, \dots, T_{k-1}) of an edge-labeled graph, define its *signature* $v^{(T_0, \dots, T_{k-1})}$ by

$$v^{(T_0, \dots, T_{k-1})} := \sum_{i=0}^{k-1} v^{T_i}. \quad (19)$$

An edge-labeled graph G on t vertices is called *k -distinguishable* if G has $k(t - 1)$ edges and there exists a tree-decomposition T_0, \dots, T_{k-1} of G with a unique signature. That is, for any tree-decomposition T'_0, \dots, T'_{k-1} with the same signature $v^{(T'_0, \dots, T'_{k-1})} = v^{(T_0, \dots, T_{k-1})}$, we have $T'_i = T_i$ for $i = 0, \dots, k - 1$.

With these definitions, we can now conjecture a hypergraph version of the Nash-Williams–Tutte theorem.

Conjecture 6.1. *Let t and k be positive integers. Every k -weakly-partition-connected hypergraph H on t vertices has a k -distinguishable tree-assignment.*

The key result of this section is that our optimal list decoding of Reed–Solomon codes conjecture follows from our hypergraph Nash-Williams–Tutte conjecture. This connection is proved in Section 6.2 below.

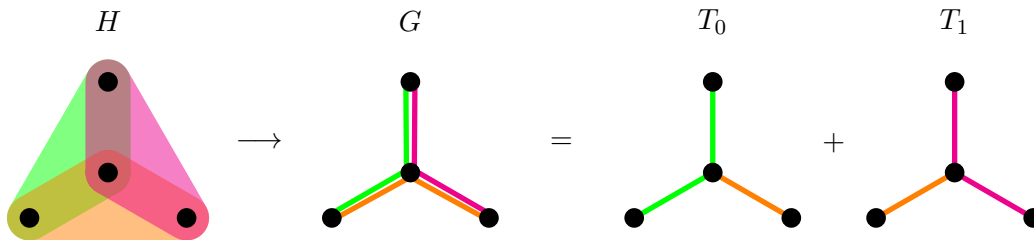
⁷There is also a notion of “ k -partition-connected” for hypergraphs which uses $\min\{\mathcal{P}(e) - 1, 1\}$ in the sum. In other words, a hypergraph is k -partition-connected if any partition \mathcal{P} has at least $k(|\mathcal{P}| - 1)$ crossing edges. This notion admits a Nash-Williams–Tutte type theorem: any k -partition-connected hypergraph can be decomposed into k 1-partition-connected hypergraphs [FKK03]

Theorem 6.2. *Conjecture 6.1 implies Conjecture 1.5 and thus Conjecture 1.4.*

One should convince themselves that Conjecture 6.1 (if true) is a generalization of the Nash-Williams–Tutte theorem. Indeed, when H is a (non-hyper) graph, the conjecture boils down to the Nash-Williams–Tutte theorem. If H is a graph on $k(t - 1)$ edges, then there is only one tree-assignment G of H , namely H itself with each edge labeled by itself. All edges have distinct edge-labels, so for any tree-decomposition T_0, \dots, T_{k-1} , the signature $v^{(T_0, \dots, T_{k-1})}$ is unique. Thus, showing G is distinguishable, is equivalent to showing G has k edge-disjoint spanning trees, which follows from the Nash-Williams–Tutte theorem. In the correspondence between hypergraph partitions and intersection matrices, the special case when H is a graph corresponds to Theorem 3.1.

To give more intuition when H is not a graph, we give the following example.

Example 6.3. *Let $t = 4$, and $k = 2$. Below, H is a 2-weakly-partition-connected hypergraph. We take a tree assignment of H to obtain an edge-labeled graph G on 6 edges, where each edge is labeled by its color. The tree-decomposition $T_0 \cup T_1$ demonstrates that G is 2-distinguishable: We have $v^{T_0} = (2, 1, 0)$ and $v^{T_1} = (0, 1, 2)$ so the signature is $v^{(T_0, T_1)} = 0 \cdot v^{T_0} + 1 \cdot v^{T_1} = (0, 1, 2)$. One can check that any other tree decomposition (T'_0, T'_1) of G has a different signature $v^{(T'_0, T'_1)} \neq (0, 1, 2)$. Thus, G is 2-distinguishable. Hence, H is a 2-weakly partition-connected hypergraph with a 2-distinguishable tree-assignment, as predicted by Conjecture 6.1.*



As evidence towards Conjecture 6.1, we point out that the “easy part” of the conjecture follows from the Nash-Williams–Tutte theorem. One can check that, even in general hypergraphs, every tree-assignment of a k -weakly-partition-connected hypergraph gives a k -partition connected graph. Thus, any tree-assignment graph can be partitioned into k spanning trees by the Nash-Williams–Tutte theorem, establishing the “existence” part of the graph G being k -distinguishable. Thus, the hard part of Conjecture 6.1 is the “uniqueness” part, finding a spanning tree partition with a unique signature.

As further evidence towards Conjecture 6.1, we point out that a relaxation of Conjecture 6.1 is true by assuming a larger weak-partition-connectivity [CS07, CCV09].

Theorem 6.4 (Follows from [CS07, CCV09]). *There exists an absolute constant C such that the following holds. Let t and k be positive integers. Every $C(\log t)k$ -weakly-partition-connected hypergraph H on t vertices has a tree-assignment with a k -distinguishable subgraph.*

The results in [CS07, CCV09] look slightly different from Theorem 6.4 and we briefly describe the connection here. Both of the works prove that every $C(\log t)k$ -weakly partition-connected hypergraph has k hyperedge-disjoint connected subhypergraphs:⁸ the first [CS07] proves an equivalent statement about bipartite Steiner tree packings, and the second [CCV09] proves a more general result about disjoint bases of polymatroids and also improves the constant C in front of the log factor

⁸A hypergraph is connected if, for every two vertices v and v' , there is a path $v = v_0, v_1, \dots, v_\ell = v'$ such that, for all i , v_{i-1}, v_i share a hyperedge.

over [CS07]. Furthermore, it is not difficult to show that, for any hypergraph H with k hyperedge-disjoint connected subhypergraphs H_0, \dots, H_{k-1} , any tree assignment of H has a k -distinguishable subgraph: briefly, the k -distinguishable subgraph G is the union of spanning trees T_0, \dots, T_{k-1} of the tree assignments of H_0, \dots, H_{k-1} that are implied by the tree assignments of H , and the spanning trees T_0, \dots, T_{k-1} give the k edge-disjoint spanning trees of G that certify k -distinguishability. Hence, any $C(\log t)k$ -weakly-partition-connected hypergraph has k hyperedge-disjoint connected subhypergraphs and thus is k -distinguishable, which is Theorem 6.4.

While it is known that the log factor in the results of [CS07, CCV09] cannot be removed [BJT03]⁹, one can still hope that Conjecture 6.1 is true. The results in [CS07, CCV09] are used to obtain k -distinguishability by taking a tree packing where edges of the same label are always in the same tree. Keeping the same edge labels in the same trees is not necessary in general to obtain distinguishability. Indeed Example 6.3 illustrates that a tree-assignment can be k -distinguishable even when the original hypergraph does not have a partition into k connected subhypergraphs. Thus, we hope that the log factor can still be saved by splitting edge labels of the tree-assignment graph across the k spanning trees. As we pointed out earlier, the existence of a tree-packing in the tree-assignment graph follows from the Nash-Williams–Tutte theorem, so the hard part of the conjecture is not finding the tree-packing, but finding a signature-unique one.

We point out that Theorem 6.4 can be used to obtain a second proof of Theorem 1.1 by following the proof of Theorem 6.2, which connects hypergraph partitions to intersection matrices and then applies a polynomial method to obtain list decodable Reed–Solomon codes. The extra $O(\log t)$ factor in the weak-partition-connectivity of Theorem 6.4 appears as an $O(\log \frac{1}{\varepsilon})$ factor loss in the rate of the Reed–Solomon code. Since one proof of Theorem 1.1 is already given, and the key ideas for this second proof are covered throughout the rest of the paper, we omit the details of this second proof of Theorem 1.1.

In addition to implying optimal list decoding of Reed Solomon codes (Theorem 6.4), Conjecture 6.1 may be of independent interest as a candidate hypergraph Nash-Williams–Tutte generalization. On one hand, a hypergraph Nash-Williams–Tutte generalization is known for *partition-connectivity* [FKK03]. A hypergraph is k -partition-connected if any partition \mathcal{P} has at least $k(|\mathcal{P}| - 1)$ crossing edges. Frank, Király, and Kriesell [FKK03] showed that every k -partition-connected hypergraph has k edge-disjoint 1-partition-connected subhypergraphs. On the other hand, Nash-Williams–Tutte generalizations for *weak-partition-connectivity* seem to be less studied, and Conjecture 6.1 provides a plausible generalization for weak-partition-connectivity.

6.2 Proof of Theorem 6.2

In this section we prove Theorem 6.2. To do so, we show the first implication, establishing the connection between hypergraph partitions and intersection matrices outlined in Section 6.1. The second implication was proved in [ST20a]. At a high level, the first implication of Theorem 6.2 holds because the uniqueness of the signature of k edge-disjoint spanning trees implies the uniqueness of a monomial in the determinant expansion of an intersection matrix. Because such a monomial is unique, it does not cancel with any other terms in the determinant expansion, implying that the determinant is nonzero. We now give the details.

We first derive a sufficient condition for a hypergraph being k -weakly-partition-connected.

⁹[BJT03] shows there exist $\Omega(\log n)$ -weakly-partition-connected hypergraphs without two edge-disjoint connected subhypergraphs.

Lemma 6.5. *Let H be hypergraph on the vertex set $[t]$ where for all $J \subsetneq [t]$,*

$$\begin{aligned} \sum_{e \in E(H)} \max(0, |e \cap J| - 1) &\leq k(|J| - 1) \\ \text{and } \sum_{e \in E(H)} (|e| - 1) &\geq k(t - 1). \end{aligned} \tag{20}$$

Then H is k -weakly-partition-connected.

Proof. According to (18) it suffices to show that for any partition \mathcal{P} of the vertices of H , $\sum_{e \in E(H)} (\mathcal{P}(e) - 1) \geq k(|\mathcal{P}| - 1)$. To see this, assume that $\mathcal{P} = \{V_1, \dots, V_s\}$. Then $\sum_{i=1}^s |V_i| = t$, and for each $e \in E(H)$, $|e| = \sum_{i=1}^s |e \cap V_i|$. By the last equality, it is not hard to check that

$$|e| = \mathcal{P}(e) + \sum_{i=1}^s \max\{0, |e \cap V_i| - 1\}.$$

It follows that

$$\begin{aligned} \sum_{e \in E(H)} (\mathcal{P}(e) - 1) &= \sum_{e \in E(H)} \left(|e| - \sum_{i=1}^s \max\{0, |e \cap V_i| - 1\} - 1 \right) \\ &= \sum_{e \in E(H)} (|e| - 1) - \sum_{i=1}^s \sum_{e \in E(H)} \max\{0, |e \cap V_i| - 1\} \\ &\geq k(t - 1) - \sum_{i=1}^s k(|V_i| - 1) = k(s - 1), \end{aligned}$$

where the last inequality follows from (20). □

We now present the proof of Theorem 6.2.

Proof of Theorem 6.2. Assume Conjecture 6.1 is true. Let $I_1, \dots, I_t \subseteq [n]$ be subsets satisfying the conditions of Conjecture 1.5. For all $i \in [n]$, let $e_i = \{j \in [t] : i \in I_j\}$. Let H be a (multi)hypergraph with vertex set $[t]$ and edge set $E(H) = \{e_i : i \in [n]\}$.

Claim 6.6. *For all subsets $J \subseteq [t]$, we have $\sum_{e \in E(H)} \max(0, |e \cap J| - 1) = \text{wt}(I_j : j \in J)$.*

Proof. We have

$$\sum_{i \in [n]} \max\{0, |e_i \cap J| - 1\} = \sum_{i \in [n]} |e_i \cap J| - \left| \bigcup_{j \in J} I_j \right| = \sum_{j \in J} |I_j| - \left| \bigcup_{j \in J} I_j \right| = \text{wt}(I_j),$$

where the first equality follows from the fact $\bigcup_{j \in J} I_j = \{i \in [n] : |e_i \cap J| \geq 1\}$, the second equality follows from easy double-counting, and the last equality follows from (1). □

The following is an easy consequence of Lemmas 6.5 and Claim 6.6.

Claim 6.7. *H is k -weakly-partitioned-connected.*

Proof. By Claim 6.6 and the setup of Conjecture 1.5, it is clear that for each $J \subsetneq [t]$

$$\sum_{e \in E(H)} \max\{0, |e \cap J| - 1\} = \text{wt}(I_J) \leq k(|J| - 1)$$

and

$$\sum_{e \in E(H)} \max\{0, |e| - 1\} = \text{wt}(I_{[t]}) \geq k(t - 1).$$

It follows by Lemma 6.5 that H is k -weakly-partition-connected. \square

By our assumption that Conjecture 6.1 is true, there exists a tree-assignment G of H that is k -distinguishable. Note that by definition G has $k(t - 1)$ edges, which are labeled by the hyperedges of H . Let $S \subseteq [n]$ be the subset so that $\{e_s : s \in S\}$ forms the set of those labels. Then, an edge $\{j, j'\}$ of G has label e_s for some $s \in S$ if and only if $s \in I_j \cap I_{j'}$.

Recall that

$$M_{k, (I_1, \dots, I_t)} = \left(\frac{\mathcal{B}_t \otimes \mathcal{I}_k}{\text{diag}\left(V_k(I_j \cap I_{j'}) : \{j, j'\} \in \binom{[t]}{2}\right)} \right),$$

and that the $\binom{t}{2}k$ columns are labeled by the pairs $\{j, j'\} \in \binom{[t]}{2}$, according to the $\binom{t}{2}$ Vandermonde matrices in the bottom diagonal. Our goal is to show that $M_{k, (I_1, \dots, I_t)}$ is nonsingular. As in the proof of Theorem 3.1, it suffices to show the nonsingularity of

$$M'_{k, (I_1, \dots, I_t)} := \left(\frac{\mathcal{I}_k \otimes \mathcal{B}_t}{(\mathcal{C}_i : 0 \leq i \leq k - 1)} \right),$$

where $\mathcal{C}_i = \text{diag}\left(V_k^{(i)}(I_j \cap I_{j'}) : \{j, j'\} \in \binom{[t]}{2}\right)$ and $V_k^{(i)}(I_j \cap I_{j'})$ is the $(i + 1)$ -th column of $V_k(I_j \cap I_{j'})$. Note that $M'_{k, (I_1, \dots, I_t)}$ is obtained by permuting the columns of $M_{k, (I_1, \dots, I_t)}$, with the column labels remaining unchanged.

The following fact is easy to verify by definition.

Fact 6.8. *Each row of $(\mathcal{C}_i : 0 \leq i \leq k - 1)$ has exactly k nonzero entries, which has the form $x_s^0, x_s^1, \dots, x_s^{k-1}$ for some $s \in S$. Moreover, there is $\{j, j'\} \in \binom{[t]}{2}$ so that $s \in I_j \cap I_{j'}$, and those k nonzero entries are all contained in $\{j, j'\}$ -labeled columns.*

Let us consider $\binom{t}{2}k \times \binom{t}{2}k$ submatrix M' of $M'_{k, (I_1, \dots, I_t)}$ obtained as follows:

1. Keep the top $\binom{t-1}{2}k \times \binom{t}{2}k$ submatrix $\mathcal{I}_k \otimes \mathcal{B}_t$.
2. For every edge $\{j, j'\}$ in G of label e_s , keep the row in $(\mathcal{C}_i : 0 \leq i \leq k - 1)$ with nonzero entries x_s^0, \dots, x_s^{k-1} in $\{j, j'\}$ -labeled columns (this is well-defined according to Fact 6.8).
3. Remove all other rows.

As G has $k(t - 1)$ edges, precisely $k(t - 1)$ rows are kept in step 2. Therefore, M' has $\binom{t-1}{2}k + k(t - 1) = \binom{t}{2}k$ rows and is thus square.

Below we show that M' has a nonzero determinant, thereby implying that $M'_{k, (I_1, \dots, I_t)}$ and hence $M_{k, (I_1, \dots, I_t)}$ are nonsingular, which is the claim of Conjecture 1.5, and thus establishing Theorem 6.2.

For that purpose, it is enough to show that there is a monomial that appears as a nonvanishing term in the determinant expansion of M' . To find such a monomial, we use the fact that G is k -distinguishable.

Recall that for a subgraph F of G , we use $v^F \in \mathbb{N}^S$ to denote the vector that counts the edge labels in F , where for $s \in S$, v_s^F is the number of edges with label e_s . Note that v^F is a vector of length $|S|$ whose coordinates are indexed by elements in S . For spanning trees T, T_0, \dots, T_{k-1} of G , define

$$x^T := \prod_{s \in S} x_s^{v_s^T} \quad \text{and} \quad x^{(T_0, \dots, T_{k-1})} := \prod_{i=0}^{k-1} (x^{T_i})^i. \quad (21)$$

Observe that, by the definition in (19), we have $v_s^{(T_0, \dots, T_{k-1})} = \sum_{i=0}^{k-1} i \cdot v_s^{T_i}$ for all $s \in S$. Hence, it follows from (21) that

$$x^{(T_0, \dots, T_{k-1})} = \prod_{s \in S} x_s^{v_s^{(T_0, \dots, T_{k-1})}}. \quad (22)$$

Since G is k -distinguishable, there exists a tree decomposition T_0, \dots, T_{k-1} such that for any other tree-assignment T'_0, \dots, T'_{k-1} , we have that the signatures $v^{(T_0, \dots, T_{k-1})} \neq v^{(T'_0, \dots, T'_{k-1})}$. Thus, it follows by (22) that the monomials $x^{(T_0, \dots, T_{k-1})} \neq x^{(T'_0, \dots, T'_{k-1})}$.

Claim 6.9. *Let T_0, \dots, T_{k-1} be spanning trees as defined above. Then, $x^{(T_0, \dots, T_{k-1})}$ appears as a nonvanishing term in the determinant expansion of M' .*

Proving Claim 6.9 establishes the nonsingularity of M' and thus, as discussed above, Theorem 6.2. For that purpose, we identify the nonzero entries in the bottom $(t-1)k$ rows of M' by tuples $(s, \{j, j'\}, i)$, where $s \in S \cap (I_j \cap I_{j'})$, $\{j, j'\} \in \binom{[t]}{2}$, and $0 \leq i \leq k-1$. Indeed, such a tuple corresponds to the entry x_s^i in the $\{j, j'\}$ -labeled column of \mathcal{C}_i . It is worth mentioning that we used two types of labeling here: first, each column of $\mathcal{I}_k \otimes \mathcal{B}_t$ and $(\mathcal{C}_i : 0 \leq i \leq k-1)$ is labeled by some edge $\{j, j'\} \in \binom{[t]}{2}$; second, each edge of G is labeled by some variable x_s , $s \in S$.

Let U denote the set of all $(t-1)k^2$ nonzero entries in the bottom $(t-1)k$ rows of M' . For $Q \subset U$, let M_Q denote the submatrix of M' obtained by removing all of the rows and columns that contain some entry in Q . We say Q is a *partial transversal* if it contains exactly one element in each of the bottom $(t-1)k$ rows of M' , and no two share a column. By the definition of determinant,

$$\det(M') = \sum_{Q \text{ partial transversal}} \pm \det(M_Q) \cdot \prod_{(s, \{j, j'\}, i) \in Q} x_s^i. \quad (23)$$

For a partial transversal Q and $0 \leq i \leq k-1$, let Q_i be the subgraph of G that corresponds to the tuples $(s, \{j, j'\}, i) \in Q$, namely,

$$Q_i = \left\{ \{j, j'\} \in \binom{[t]}{2} : (s, \{j, j'\}, i) \in Q \text{ for some } s \in I_j \cap I_{j'} \right\}.$$

Note that we view each Q_i as a labeled subgraph that preserves the labeling of G . Moreover, as Q forms a partial transversal, each Q_i is a simple graph with no multiple edges, while G could be a multigraph.

We have the following claim.

Claim 6.10. *Let $Q \subseteq U$ be a partial transversal. Then, $\det(M_Q) \neq 0$ if and only if Q_0, \dots, Q_{k-1} form pairwise edge-disjoint spanning trees of G .*

Proof of Claim 6.10. For the “only if” part, note, that by the definition of a partial transversal, M_Q is a $\binom{t-1}{2}k \times \binom{t-1}{2}k$ square matrix with k diagonal blocks, where for each $0 \leq i \leq k-1$, the $(i+1)$ -th diagonal block is obtained by removing from \mathcal{B}_t all of the columns that are labeled by the edges of Q_i . As \mathcal{B}_t has full row rank, $\det(M_Q) \neq 0$ if and only if each of the k diagonal blocks also has full row rank. By Claim 3.5, the $(i+1)$ -th block has full row rank if and only if the labels of the removed columns form an acyclic graph on the vertices $[t]$, namely, Q_i is acyclic.

Since $k(t-1)$ columns are removed in total, and an acyclic subgraph on t vertices can have at most $t-1$ edges, $\det(M_Q) \neq 0$ can only happen if each Q_i is a spanning tree. Moreover, as elements in Q form a partial transversal, we never have $(s, \{j, j'\}, i)$ and $(s, \{j, j'\}, i')$ both in Q , for $i \neq i'$. It follows that the Q_i 's are also pairwise edge-disjoint, completing the proof of the “only if” part.

According to the discussions above, it is not hard to see that the “if” part follows fairly straightforwardly from Claim 3.5. Therefore, we omit its proof.

We are now in a position to present the proof of Claim 6.9.

Proof of Claim 6.9. With the notation above, it is not hard to check by definition that for a partial transversal $Q \subseteq U$ with $\det(M_Q) \neq 0$,

$$\prod_{(s, \{j, j'\}, i) \in Q} x_s^i = \prod_{s \in S} \prod_{i=0}^{k-1} (x_s^{v_s^{Q_i}})^i = \prod_{s \in S} x_s^{v_s^{(Q_0, \dots, Q_{k-1})}} = x^{(Q_0, \dots, Q_{k-1})}. \quad (24)$$

It thus follows from (23), (24), and Claim 6.10 that

$$\det(M') = \sum_{Q_0, \dots, Q_{k-1} \text{ edge-disj. spanning trees of } G} \pm \det(M_Q) \cdot x^{(Q_0, \dots, Q_{k-1})}.$$

It is clear that by the definition of T_0, \dots, T_{k-1} , in the above summation the monomial $x^{(T_0, \dots, T_{k-1})}$ appears exactly once, and hence appears as a nonvanishing term in the determinant expansion, completing the proof of Claim 6.9 and thus the proof of Theorem 6.2. \square

Acknowledgments

Zeyu Guo is supported by NSF-BSF grant CCF-1814629 and 2017732 and the Milgrom family grant for Collaboration between the Technion and University of Haifa. He wants to thank Noga Ron-Zewi for helpful discussions.

Ray Li is supported by NSF GRFP grant DGE-1656518 and by Jacob Fox's Packard Fellowship. He thanks Bruce Spang for helpful discussions.

Chong Shangguan is supported by the Qilu Scholar Program of Shandong University, the National Key Research and Development Program of China under Grant No. 2020YFA0712100, and the Israel Science Foundation (ISF grant number 1030/15).

Itzhak Tamo is partially supported by the European Research Council (ERC grant number 852953), and by the Israel Science Foundation (ISF grant number 1030/15).

Mary Wootters is supported by NSF grant CCF-1844628 and NSF-BSF grant CCF-1814629, and by a Sloan Research Fellowship. She thanks Noga Ron-Zewi for helpful discussions.

We thank Karthik Chandrasekaran for helpful discussions about hypergraph packing theorems and for the reference [CCV09].

A Examples of Intersection Matrices

Example A.1 (4-wise intersection matrices). Given four subsets $I_1, I_2, I_3, I_4 \subseteq [n]$, the 4-wise intersection matrix $M_{k, (I_1, I_2, I_3, I_4)}$ is the $(3k + \sum_{1 \leq i < j \leq 4} |I_i \cap I_j|) \times 6k$ variable matrix

$$\left(\begin{array}{cccccc} \mathcal{I}_k & & & -\mathcal{I}_k & \mathcal{I}_k & \\ & \mathcal{I}_k & & -\mathcal{I}_k & & \mathcal{I}_k \\ & & \mathcal{I}_k & & -\mathcal{I}_k & \mathcal{I}_k \\ \hline V_k(I_1 \cap I_2) & & & & & \\ & V_k(I_1 \cap I_3) & & & & \\ & & V_k(I_2 \cap I_3) & & & \\ & & & V_k(I_1 \cap I_4) & & \\ & & & & V_k(I_2 \cap I_4) & \\ & & & & & V_k(I_3 \cap I_4) \end{array} \right).$$

Example A.2. For $k = 2$, instead of considering the following 4-wise intersection matrix

$$\left(\frac{\mathcal{B}_4 \otimes \mathcal{I}_2}{\text{diag}(V_k(I_i \cap I_j) : \{i, j\} \in \binom{[4]}{2})} \right) = \left(\begin{array}{c|c|c|c|c|c} 1 & & & -1 & 1 & \\ & 1 & & -1 & & 1 \\ \hline & & 1 & & -1 & 1 \\ & & & 1 & & -1 \\ \hline 1 & x_1 & & & & \\ & & 1 & x_2 & & \\ & & & & 1 & x_3 \\ & & & & & 1 & x_4 \\ & & & & & & 1 & x_5 \\ & & & & & & & 1 & x_6 \end{array} \right),$$

References

- [AG07] Noga Alon and Shai Gutner. Balanced families of perfect hash functions and their applications. In *International Colloquium on Automata, Languages, and Programming*, pages 435–446. Springer, 2007.
- [AG09] Noga Alon and Shai Gutner. Balanced hashing, color coding and approximate counting. In *International Workshop on Parameterized and Exact Computation*, pages 1–16. Springer, 2009.
- [AN96] Noga Alon and Moni Naor. Derandomization, witnesses for Boolean matrix multiplication and construction of perfect hash functions. *Algorithmica*, 16(4-5):434–449, 1996.
- [BBN93] Norman Biggs, Norman Linstead Biggs, and Biggs Norman. *Algebraic graph theory*, volume 67. Cambridge University Press, 1993.
- [BJT03] Jørgen Bang-Jensen and Stéphan Thomassé. Highly connected hypergraphs containing no two edge-disjoint spanning connected subhypergraphs. *Discrete Applied Mathematics*, 131(2):555–559, 2003.
- [BKR10] E. Ben-Sasson, S. Kopparty, and J. Radhakrishnan. Subspace polynomials and limits to list decoding of Reed-Solomon codes. *IEEE Trans. Inform. Theory*, 56(1):113–120, Jan 2010.
- [Bla00] Simon R. Blackburn. Perfect hash families: probabilistic methods and explicit constructions. *Journal of Combinatorial Theory, Series A*, 92(1):54–60, 2000.
- [Bla03] Simon R. Blackburn. Combinatorial schemes for protecting digital content. In *Surveys in combinatorics, 2003 (Bangor)*, volume 307 of *London Math. Soc. Lecture Note Ser.*, pages 43–78. Cambridge University Press, 2003.
- [BW98] Simon R. Blackburn and Peter R. Wild. Optimal linear perfect hash families. *Journal of Combinatorial Theory Series A*, 83(2):233–250, 1998.
- [CCV09] Gruia Călinescu, Chandra Chekuri, and Jan Vondrák. Disjoint bases in a polymatroid. *Random Structures & Algorithms*, 35(4):418–430, 2009.
- [CPS99] Jin-Yi Cai, Aduri Pavan, and D Sivakumar. On the hardness of permanent. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 90–99. Springer, 1999.
- [CS07] Joseph Cheriyan and Mohammad R Salavatipour. Packing element-disjoint steiner trees. *ACM Transactions on Algorithms (TALG)*, 3(4):47–es, 2007.
- [CW07] Q. Cheng and D. Wan. On the list and bounded distance decodability of Reed-Solomon codes. *SIAM J. Comput.*, 37(1):195–209, April 2007.
- [Die17] Reinhard Diestel. *Graph theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, Berlin, fifth edition, 2017.
- [Dou19] Ryan Dougherty. *Hash Families and Applications to t -Restrictions*. PhD thesis, Doctoral Dissertation Arizona State University, 2019.
- [DW20] Dean Doron and Mary Wootters. High-probability list-recovery, and applications to heavy hitters. In *Electron. Colloquium Comput. Complex.*, volume 27, page 162, 2020.
- [Eli57] Peter Elias. List decoding for noisy channels. *Wescon Convention Record, Part 2, Institute of Radio Engineers*, pages 99–104, 1957.
- [FK84] Michael L Fredman and János Komlós. On the size of separating systems and families of perfect hash functions. *SIAM Journal on Algebraic Discrete Methods*, 5(1):61–68, 1984.
- [FKK03] András Frank, Tamás Király, and Matthias Kriesell. On decomposing a hypergraph into k connected sub-hypergraphs. *Discrete Applied Mathematics*, 131(2):373–383, 2003.
- [FKS20] Asaf Ferber, Matthew Kwan, and Lisa Saueremann. List-decodability with large radius for reed-solomon codes. *arXiv preprint arXiv:2012.10584*, 2020.

- [GR06] V. Guruswami and A. Rudra. Limits to list decoding Reed–Solomon codes. *IEEE Trans. Inform. Theory*, 52(8):3642–3649, August 2006.
- [GR08] Venkatesan Guruswami and Atri Rudra. Explicit codes achieving list decoding capacity: Error-correction with optimal redundancy. *IEEE Transactions on Information Theory*, 54(1):135–150, 2008.
- [GRS19] Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. Essential coding theory. *Draft available at <http://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/>*, 2019.
- [GS99] Venkatesan Guruswami and Madhu Sudan. Improved decoding of Reed–Solomon and algebraic-geometry codes. *IEEE Transactions on Information Theory*, 45(6):1757–1767, 1999.
- [GS01] Venkatesan Guruswami and Madhu Sudan. Extensions to the johnson bound. *Manuscript, February*, 2001.
- [GST21] Eitan Goldberg, Chong Shangguan, and Itzhak Tamo. List-decoding and list-recovery of reed-solomon codes beyond the Johnson radius for any rate. *arXiv preprint arXiv:2105.14754*, 2021.
- [GUV09] Venkatesan Guruswami, Christopher Umans, and Salil Vadhan. Unbalanced expanders and randomness extractors from Parvaresh–Vardy codes. *Journal of the ACM (JACM)*, 56(4):1–34, 2009.
- [GW13] V. Guruswami and C. Wang. Linear-algebraic list decoding for variants of Reed-Solomon codes. *IEEE Trans. Inform. Theory*, 59(6):3257–3268, June 2013.
- [Joh62] Selmer Johnson. A new upper bound for error-correcting codes. *IRE Transactions on Information Theory*, 8(3):203–207, 1962.
- [Juk11] Stasys Jukna. *Extremal combinatorics: with applications in computer science*. Texts in Theoretical Computer Science. An EATCS Series. Springer, Heidelberg, second edition, 2011.
- [KM88] János Körner and Katalin Marton. New bounds for perfect hashing via information theory. *European Journal of Combinatorics*, 9(6):523–530, 1988.
- [Kop15] Swastik Kopparty. List-decoding multiplicity codes. *Theory of Computing*, 11(1):149–182, 2015.
- [Kör86] János Körner. Fredman–komlós bounds and information theory. *SIAM Journal on Algebraic Discrete Methods*, 7(4):560–570, 1986.
- [KRZSW18] S. Kopparty, N. Ron-Zewi, S. Saraf, and M. Wootters. Improved decoding of folded Reed-Solomon and multiplicity codes. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 212–223. IEEE, 2018.
- [LP20] Ben Lund and Aditya Potukuchi. On the list recoverability of randomly punctured codes. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2020)*, volume 176, pages 30:1–30:11, 2020.
- [Meh84] Kurt Mehlhorn. *Data structures and algorithms 1: Sorting and searching*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, Berlin, 1984.
- [Nil94] Alon Nilli. Perfect hashing and probability. *Combinatorics, Probability & Computing*, 3(3):407–409, 1994.
- [NW61] Crispin St. J. A. Nash-Williams. Edge-disjoint spanning trees of finite graphs. *Journal of the London Mathematical Society*, 1(1):445–450, 1961.
- [NW95] Ilan Newman and Avi Wigderson. Lower bounds on formula size of Boolean functions using hypergraph entropy. *SIAM Journal on Discrete Mathematics*, 8(4):536–542, 1995.
- [RS60] Irving S. Reed and Gustave Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960.

- [RW14] Atri Rudra and Mary Wootters. Every list-decodable code for high noise has abundant near-optimal rate puncturings. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC 2014, pages 764–773, 2014.
- [SG16] Chong Shangguan and Gennian Ge. Separating hash families: A Johnson-type bound and new constructions. *SIAM Journal on Discrete Mathematics*, 30(4):2243–2264, 2016.
- [Sin64] R. Singleton. Maximum distance q -nary codes. *IEEE Trans. Inform. Theory*, 10(2):116–118, April 1964.
- [ST20a] Chong Shangguan and Itzhak Tamo. Combinatorial list-decoding of Reed-Solomon codes beyond the Johnson radius. In *Proceedings of the 52nd Annual ACM Symposium on Theory of Computing*, STOC 2020, pages 538–551, 2020.
- [ST20b] Chong Shangguan and Itzhak Tamo. Degenerate turán densities of sparse hypergraphs. *Journal of Combinatorial Theory, Series A*, 173:105228, 2020.
- [STV01] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the xor lemma. *Journal of Computer and System Sciences*, 62(2):236–266, 2001.
- [Tut61] William T. Tutte. On the problem of decomposing a graph into n connected factors. *Journal of the London Mathematical Society*, 1(1):221–230, 1961.
- [Woz58] John M. Wozencraft. List decoding. *Quarterly Progress Report, Research Laboratory of Electronics, MIT*, 48:90–95, 1958.
- [XY19] Chaoping Xing and Chen Yuan. Beating the probabilistic lower bound on perfect hashing. *arXiv preprint arXiv:1908.08792*, 2019.