



Storage Codes on Coset Graphs with Asymptotically Unit Rate

Alexander Barg¹ · Moshe Schwartz^{2,3} · Lev Yohananov^{4,5}

Received: 22 December 2022 / Revised: 8 March 2024 / Accepted: 10 June 2024

© The Author(s), under exclusive licence to János Bolyai Mathematical Society and Springer-Verlag GmbH Germany, part of Springer Nature 2024

Abstract

A storage code on a graph G is a set of assignments of symbols to the vertices such that every vertex can recover its value by looking at its neighbors. We consider the question of constructing large-size storage codes on triangle-free graphs constructed as coset graphs of binary linear codes. Previously it was shown that there are infinite families of binary storage codes on coset graphs with rate converging to $3/4$. Here we show that codes on such graphs can attain rate asymptotically approaching 1. Equivalently, this question can be phrased as a version of hat-guessing games on graphs (e.g., Cameron et al., in: Electron J Combin 23(1):48, 2016). In this language, we construct triangle-free graphs with success probability of the players approaching one as the number of vertices tends to infinity. Furthermore, finding linear index codes of rate approaching zero is also an equivalent problem.

Keywords Storage codes · Coset graphs · Triangle-free graphs · Index coding · Hat games

✉ Alexander Barg
abarg@umd.edu

Moshe Schwartz
schwartz.moshe@mcmaster.ca

Lev Yohananov
levyuhananov@gmail.com

¹ Department of ECE and Institute for Systems Research, University of Maryland, College Park, MD 20742, USA

² School of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer Sheva 8410501, Israel

³ Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON L8S 4K1, Canada

⁴ School of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer Sheva 8410501, Israel

⁵ Institute for Systems Research, University of Maryland, College Park, MD 20742, USA

1 Introduction

Suppose we are given a connected graph $G(V, E)$ on N vertices. Denote by $\mathcal{N}(v)$ the neighborhood of v in G , i.e., the set of vertices of G adjacent to v , and let \mathbb{F}_q be a finite field of size q . A *storage code* for G is a set \mathcal{C} of vectors $c = (c_v)_{v \in V} \in \mathbb{F}_q^N$ such that for every $c \in \mathcal{C}$ and $v \in V$ the value of the coordinate c_v is uniquely determined by the values of c_w , $w \in \mathcal{N}(v)$. More formally, suppose that for every vertex v there is a function $f_v : \mathbb{F}_q^N \rightarrow \mathbb{F}_q$ such that for every $c \in \mathcal{C}$ and every $v \in V$ we have $c_v = f_v((c_w)_{w \in \mathcal{N}(v)})$, where we assume some implicit fixed ordering of the vertices.

The concept of storage codes was introduced around 2014 in [17, 21] and studied in subsequent papers [4, 18]. In coding theory literature, this concept was motivated by a more general notion of codes with locality [12], which has enjoyed considerable attention during the last decade [19].

The main problem associated with storage codes is constructing codes of large size, expressed as a function of the parameters of the graph. To make the comparison easier, define the *code rate* $R_q(\mathcal{C}, G) := \log_q(|\mathcal{C}|)/N$, and let $R_q(G) = \max_{\mathcal{C}} R_q(\mathcal{C}, G)$. The first observation is that for dense graphs it is easy to construct codes of large size or rate: for instance, if the graph is complete, $G = K_N$, then the condition of vertex recovery from its neighbors will be satisfied if the codewords satisfy a global parity check, i.e., $\sum_{v \in V} c_v = 0$ for every $c \in \mathcal{C}$. In this case $|\mathcal{C}| = q^{N-1}$ and $R(\mathcal{C}) = R_q(K_N) = \frac{N-1}{N}$ for any q . A variant of this idea clearly applies if G contains many cliques or large cliques, and therefore it makes sense to focus on the case of graphs with no cliques at all, i.e., triangle-free graphs. This case received considerable attention in the literature regarding both storage codes and guessing games on graphs which we discuss next.

The following version of guessing games on graphs, introduced in [20], turns out to be equivalent to the construction problem of storage codes. The vertices are assigned colors out of a finite set Q of size q , and each vertex attempts to guess its color based on the colors of its neighbors. The game is won if all the vertices correctly guess their colors. The strategy may be agreed upon before the start of the game, but no communication between the vertices is allowed once the colors have been assigned. Suppose that the assignment $x \in Q^N$ is chosen randomly from the available options, and the goal of the game is to minimize the probability of failure. The following strategy connects this game with storage codes. Let \mathcal{C} be a q -ary storage code for the graph. Every vertex v assumes that the assignment is a codeword in \mathcal{C} and guesses its color from its neighbors, then the probability of success is $P_s(\mathcal{C}, G) = \frac{|\mathcal{C}|}{q^N}$. This quantity is sometimes expressed via the *guessing number* of G , defined as $\text{gn}_q(G) = N + \log_q \max_{\mathcal{C}} P_s(\mathcal{C}, G)$ [5], so in our notation $\frac{1}{N}(\text{gn}_q(G)) = R_q(G)$.

It was soon realized that the problem of finding storage codes is also equivalent to constructing procedures for linear index coding with *side information graph* G . We refer to the introduction of [4] for a brief overview of the known results on storage codes as well as the connections to guessing and linear index coding, and to [2] for a

more detailed presentation. In particular, as shown in [1, 17]

$$1 - I_q(G) \leq R_q(G) \leq 1 - I_q(G) + N^{-1} \log_q(N \ln q),$$

where $I_q(G)$ is the rate of the linear index code on G .

This paper further develops the work started in [4], and it shares with it the general approach to the construction of storage codes. In particular, as in [4], we confine ourselves to *linear binary codes* and we assume that the repair function of a vertex f_v is simply a parity check. In other words, $c_v = \sum_{w \in \mathcal{N}(v)} c_w$ holds for all $c \in \mathcal{C}$ and all $v \in V$, where the sum is computed modulo 2. This repair rule involves an implicit assumption that vertex recovery relies on the *full parity*, i.e., all the neighbors contribute to the recovery (the general definition does not include this stipulation). Thus, the parities of the vertices are given by the corresponding rows of the matrix $A + I$, where $A = A(G)$ is the adjacency matrix of G and I is the identity (in other words, we are adding a self-loop to every $v \in V$). For a given graph G define the *augmented adjacency matrix* $\tilde{A}(G) := A(G) + I$. The dimension of the code \mathcal{C} equals $\dim(\mathcal{C}) = N - \text{rk } \tilde{A}(G)$, and thus we are interested in constructing graphs for which $\tilde{A}(G)$ has the smallest possible rank given the number of vertices N .

The assumption of using full parities is essential for our results. To explain the reasons, recall first that an $N \times N$ matrix M is said to fit a graph $G(V, E)$ if $M_{u,v} = 0$ whenever the vertices u and v with $u \neq v$ are not adjacent, and $M_{u,u} \neq 0$ for all $u \in V$. The minimum rank of G over \mathbb{F}_q is defined as $\text{minrk}_q(G) = \min\{\text{rk}_{\mathbb{F}_q}(M) \mid M \in \mathbb{F}_q^{N \times N}, M \text{ fits } G\}$. It is clear that the matrix M can be used to define a storage code on G : as long as there are no rows with just one nonzero element, we can take M instead of \tilde{A} as the parity-check matrix. This implies that

$$R_q(G) \geq 1 - \text{minrk}_q(G)/N,$$

and it is also known that $I_q(G) \leq \text{minrk}_q(G)/N$ [3]. The minimum rank of graphs was introduced by Haemers [13] in his work on Shannon capacity of graphs, and it was used later in studies of linear index codes [16] and complexity of arithmetic circuits [7]. In their recent work on a conjecture from [7], Golovnev and Haviv [10] estimated the minimum rank of generalized Kneser graphs (recall that the vertices of $K^<(n, s, m)$ are all s -subsets of $[n]$ and two sets are adjacent if they intersect on at most m elements). The next theorem combines Lemma 19 and Theorem 22 from [10] (see also [14]) and is rephrased to match our notation and terminology ([10] did not make a connection to either storage codes or index codes or guessing games).

Theorem 1 ([10]) *Let r be a multiple of 6. Then the graph $K^<(r, \frac{r}{2}, \frac{r}{6})$ contains no triangles, and its minimum rank over \mathbb{F}_2 is at most $2^{rh(1/3)} < 2^{0.92r}$, where $h(x) = -x \log_2 x - (1-x) \log_2(1-x)$ is the entropy function. Consequently, there exists a sequence of storage codes \mathcal{C}_r of length $N = \binom{r}{r/2}$ and rate $R(\mathcal{C}_r) \geq 1 - \frac{1}{\sqrt{\pi r}} 2^{-0.08r}$.*

The result of [10] is in fact more general than stated above in several respects. Namely, the Kneser graphs $K^<(r, \frac{r}{2}, m)$ with $m \leq n/(2l)$ do not contain odd cycles of any length up to l , the estimate of their minimum rank is valid for all finite fields, and by

considering subgraphs of $K^<$ one can extend the main claim to code lengths other than those in the theorem. Related to this, it has been shown in [11] that the minimum rank of general random graphs from the Erdős–Rényi ensemble behaves as $\Theta(N/\log N)$, i.e., is sublinear in N over any fixed finite field.

We now return to the main topic of this paper. For a d -regular graph it is easy to construct a code of rate $1/2$. This is a folklore result accomplished by placing bits on the edges and assigning to every vertex the d -vector of bits written on the edges incident to it. The rate of the obtained code over the alphabet of size 2^d is readily seen to be $1/2$. Several codes of higher rates were constructed in [5]. Namely, its authors observed that each of the following (regular, triangle-free) graphs: the Clebsch graph on 16 vertices, the Hoffman–Singleton graph on 50 vertices, the Gewirtz graph on 56 vertices, and the Higman–Sims graph on 100 vertices yield storage codes of rate above $1/2$; the highest rate of 0.77 attained by the last of these. See [5, Prop. 9] for details.

In their search of higher-rate codes, the authors of [4] studied triangle-free graphs formed as coset graphs of linear codes (see Sect. 2 for their definition). They constructed an infinite family of graphs G_r with $|V(G_r)| = 2^{r+1}$ vertices, $r \geq 4$, that admit storage codes of rate $R = \frac{3}{4} - 2^{-r}$. Moreover, [4] also gave an example of a triangle-free graph on $N = 2^{16}$ vertices with $\text{rk}(\tilde{A}) = 11818/65536$, i.e., code rate slightly above 0.8196. The authors of [4] speculated that there may exist families of triangle-free graphs that admit storage codes of rate approaching one. Without the assumption of full parities, an affirmative answer follows already from Theorem 1. As our main result, we show that this is also true if the code is constructed from the full adjacency matrix of its graph.

Theorem 2 *There exists an infinite family of connected triangle-free graphs $(G_m)_m$ on $N = N(m)$ vertices such that $\frac{1}{N} \text{rk}(\tilde{A}(G_m)) \rightarrow 0$ as $m \rightarrow \infty$. In other words,*

$$\lim_{m \rightarrow \infty} R_2(G_m) = 1$$

and

$$\text{gn}_2(G_m) = N(1 - o(1)).$$

If we let $R_2(N) = \max R_2(G_N)$ to be the largest code rate over all triangle-free graphs on N vertices, then this theorem implies that $\limsup_{N \rightarrow \infty} R_2(N) = 1$. Since $R_q(N) \geq R_2(N)$ for all alphabets with $q \geq 2$ a power of a prime, this claim is also true for all such q . Moreover, in the context of guessing games it has been shown that the storage capacity $R_q(G)$ is almost monotone in q . Namely, the following is true.

Theorem 3 ([5, 6, 9]) *For any graph G , alphabet size q , and $\epsilon > 0$ there exists $q_0(G, q, \epsilon)$ such that for all $q' > q_0$*

$$R_{q'}(G) \geq R_q(G) - \epsilon.$$

Graphs in the family $(G_m)_m$ are constructed as coset graphs of binary linear codes. Their explicit description will be given in Sect. 5 after we develop the tools for analyzing the rank of adjacency matrices of coset graphs. Along the way we illustrate

the usefulness of our approach by giving simple proofs of some of the earlier results concerning storage codes on coset graphs.

2 The Construction of [4] and Our Approach

To remind the reader the definition of coset graphs, suppose we are given a binary linear code $C \subset F^n$ with an $r \times n$ parity-check matrix H , where $F = \mathbb{F}_2$. This means that $C = \ker(H)$ and $\dim(C) \geq n - r$; in other words, H is allowed to contain dependent rows. Denote by S the set of columns of H . Now consider the Cayley graph $G = \text{Cay}(F^r, S)$ on the group $(F^r)_+$ with generators in S , where r is the number of rows of H . The vertex set of G is $V(G) = F^r$, and a pair of vertices v, v' are connected if there is a column (generator) $h \in S$ such that $v = v' + h$. Since the group is Abelian, the graph G is undirected. The vertices of G can be also viewed as cosets in \mathbb{F}_2^n/C , with two cosets connected if and only if the Hamming distance between them (as subsets) is one. It is clear that G is triangle-free once the minimum distance of the code C is at least four. We will assume this throughout and note that in particular, this implies that the columns in H are distinct.

Having constructed G from C , we consider the code $\mathcal{C} = \ker(\tilde{A}(G))$. Clearly, \mathcal{C} is a storage code for G . Below, to distinguish between the seed codes C and storage codes \mathcal{C} , we call the former *small codes* and the latter *big codes*. Generally, finding the parameters of \mathcal{C} from the parameters of C is a nonobvious task, and even if the small code is simply a repetition code, computing the rank of \tilde{A} is not immediate. This problem in fact was considered earlier in the context of constructing quantum codes [8], and we refer to [4] for a brief discussion of this connection. The family of storage codes in [4] is constructed starting with the parity-check matrix H_r obtained from the parity-check matrix of the extended Hamming code of length 2^{r-1} (see Sect. 5 for more details). Below we often find it convenient to include 0_r (the all-zero vector of length r) in the set S , and then the adjacency matrix $A(G)$ includes ones on the diagonal. Accordingly, in such cases we do not use the notation \tilde{A} and write A instead.

The approach taken in [4] relies on detailed analysis of the action of the adjacency operator A on the space of functions $f : F^r \rightarrow F$. While it indeed yields the value of the rank for the graphs G obtained from H_r , extending it to other families of codes looks difficult. The approach taken here relies on the following observations. Generators $h \in S$ act on F^r as permutations σ_h , and clearly $\sigma_h^2(v) = v$ for every $v \in F^r$. Therefore for a given $h \neq 0$, σ_h can be written as a product of disjoint transpositions (cycles of length 2), $\sigma_h = (v_0, v_1)(v_2, v_3) \dots (v_{2r-2}, v_{2r-1})$, where $v_{2i} + v_{2i+1} = h$ for all $i \geq 0$ (note that the labeling of the vectors depends on h). For any vector $v \in F^r$, not necessarily a generator, the action of σ_v can be written as a $2^r \times 2^r$ permutation matrix, which we denote below by Γ_v . For every $v, u, w \in F^r$ we have $(\Gamma_v)_{u,w} = \mathbb{1}(v = u + w)$, i.e., every row contains a single 1 in the column $u + v$. In particular, $\Gamma_0 = I$. Denote by $\mathcal{M}_r = \{\Gamma_v \mid v \in F^r\}$ the collection of matrices Γ_v and note that it forms a multiplicative group $(\mathcal{M}_r)_\times \cong (F^r)_+$; thus, $\Gamma_v^2 = I$ and $\Gamma_v \Gamma_w = \Gamma_w \Gamma_v$ for all v, w .

Our approach has a number of common features with [4], but it is phrased entirely in terms of permutation matrices and their combinations, which supplant actions of

the generators of the group. This view enables us to isolate subblocks of the adjacency matrix, and to manipulate those subblocks to rearrange the matrix so that it becomes possible to control the rank.

3 Properties of Permutation Matrices

In this section, we formulate some results about combinations (products and sums) of the matrices Γ_v , where the computations are performed modulo 2. Throughout we assume some fixed order of the vectors in F^r . Our first lemma expresses the following obvious fact: acting on F^r first by u and then by v is the same as acting by $u + v$.

Lemma 4 *For any $v, w \in F^r$, $\Gamma_v \Gamma_w = \Gamma_{v+w}$.*

Proof First, $(\Gamma_v \Gamma_w)_{u,u'} = 1$ if and only if $(\Gamma_v)_{u,z} = 1$ and $(\Gamma_w)_{z,u'} = 1$, i.e., $v = u+z$ and $w = z+u'$ for some $z \in F^r$, or $v+w = u+u'$. On the other hand, by definition $(\Gamma_{v+w})_{u,u'} = 1$ if and only if $v+w = u+u'$. \square

For any $u, w \in F^r$ there exists exactly one $v \in F^r$ such that $(\Gamma_v)_{u,w} \neq 0$, and thus $\sum_v \Gamma_v = 1_{2^r \times 2^r}$ (the $2^r \times 2^r$ all-ones matrix). Thus, if a binary matrix A can be written as a sum of permutations, this representation is unique (up to an even number of repeated summands). Therefore, if $A_V := \sum_{v \in V} \Gamma_v$, where $V \subseteq F^r$ is some subset, is a sum of permutation matrices, the quantity $s(A_V) := |V|$ is well defined. Let $\mathcal{P}_r := \{A_V \mid V \subseteq F^r\}$ be the set of all sums of the permutation matrices. Below we usually suppress the subscript V from the notation.

Proposition 5 *The set \mathcal{P}_r with operations “+” and “·” forms a commutative matrix ring with identity.*

Proof By definition, the matrices in \mathcal{P}_r form a commutative additive group. The remaining properties of the ring follow immediately from Lemma 4 which reduces sums of products of Γ ’s to simply sums. \square

As above, the rank in the next lemma is computed over \mathbb{F}_2 .

Lemma 6 *Let $A, B \in \mathcal{P}_r$.*

- (1) *If $s(A)$ is odd, then $A^2 = I$ and hence $\text{rk}(A) = 2^r$.*
- (2) *If both $s(A)$ and $s(B)$ are odd, then $s(AB)$ is odd.*
- (3) *If $s(A)$ is odd, then there is $C \in \mathcal{P}_r$ such that $AC = CA = B$.*
- (4) *For any $\Gamma \in \mathcal{M}_r$, $\text{rk}(A) = \text{rk}(A\Gamma)$.*

Proof (1) For odd ℓ let $A = \sum_{i=1}^{\ell} \Gamma_{v_i}$ such that $\Gamma_{v_i} \in \mathcal{M}_r$. Thus,

$$A^2 = \left(\sum_{i=1}^{\ell} \Gamma_{v_i} \right)^2 = \sum_{i=1}^{\ell} \Gamma_{v_i}^2 + \sum_{i \neq j} \Gamma_{v_i+v_j}.$$

The last sum has every term appearing twice, so it vanishes mod 2, and the first is formed of an odd number of identities I , so it equals I . Thus, A is nonsingular.

(2) For ℓ and m odd, let $A = \sum_{i=1}^{\ell} \Gamma_{v_i}$, $B = \sum_{j=1}^m \Gamma_{w_j} \in \mathcal{P}_r$ such that $\Gamma_{v_i}, \Gamma_{w_j} \in \mathcal{M}_r$. Therefore,

$$AB = \sum_{i=1}^{\ell} \Gamma_{v_i} \cdot \sum_{j=1}^m \Gamma_{w_j} = \sum_{i=1}^{\ell} \sum_{j=1}^m \Gamma_{v_i + w_j}.$$

Since ℓ and m are odd, the number of terms in this sum is odd. If some of them are repeated, they cancel in pairs, without affecting the parity.

(3) Let $C = AB$. Since \mathcal{P}_r is closed under multiplication, $C \in \mathcal{P}_r$. Next, $CA = AC = A^2B = B$.

(4) This is obvious since multiplying by a permutation matrix over any field preserves the rank. \square

The next lemma establishes another simple property of the matrices in \mathcal{P}_r .

Lemma 7 For $t \geq 2$, Let A_1, A_2, \dots, A_t and B be matrices in \mathcal{P}_r and suppose that $s(B)$ is odd. Let $D = (A_1|A_2|\dots|A_t)$ and $D' = (A_1B|A_2B|\dots|A_tB)$, then $\text{RowSpan}(D) = \text{RowSpan}(D')$.

Proof Observe that

$$\begin{aligned} D' &= (A_1B|A_2B|\dots|A_tB) = (BA_1|BA_2|\dots|BA_t) = BD, \\ D &= (A_1|A_2|\dots|A_t) = (BA_1B|BA_2B|\dots|BA_tB) = BD'. \end{aligned}$$

The first of these relations implies that $\text{RowSpan}(D') \subseteq \text{RowSpan}(D)$, and the second implies the reverse inclusion. \square

Lemma 8 The action of \mathcal{M}_r on the set \mathcal{P}_r partitions this set into equivalence classes.

Proof Transitivity follows by the remark before Lemma 4 (or from the lemma itself): if $A_1 = \Gamma_u A_2$ and $A_2 = \Gamma_v A_3$ then $A_1 = \Gamma_w A_3$ for $w = u + v$. \square

Denote by $[A]$ the equivalence class of $A \in \mathcal{P}_r$ and note that, since the action of \mathcal{M}_r is faithful, $|[A]| = |\mathcal{M}_r| = 2^r$.

Lemma 9 Let $D_i, i = 1, \dots, t$ and A be matrices from \mathcal{P}_r and let $D = (D_1|D_2|\dots|D_t)$. If $D_i \in [A], i = 1, \dots, t$ then $\text{rk}(A) = \text{rk}(D)$.

Proof For simplicity, let $t = 2$ and let $D_1 = A\Gamma_u$ and $D_2 = A\Gamma_v$ for some $u, v \in F^r$. Clearly $\text{rk}(A\Gamma_u|A\Gamma_v) = \text{rk}(A\Gamma_u|A\Gamma_u)$ since $A\Gamma_u$ and $A\Gamma_v$ share the same column set. Next, we have

$$\begin{aligned} \text{rk}(D) &= \text{rk}(D_1|D_2) = \text{rk}(A\Gamma_u|A\Gamma_v) \\ &= \text{rk}(A\Gamma_u|A\Gamma_u) = \text{rk}(\Gamma_u A|\Gamma_u A) = \text{rk}(\Gamma_u(A|A)) = \text{rk}(A). \end{aligned}$$

\square

4 Partitioning the Adjacency Matrix

Suppose we are given a small code $C \subset F^n$, and let $S = \{h_1, \dots, h_n\}$ be the set of columns of its parity-check matrix. We begin with an obvious claim, stated in the next lemma.

Lemma 10 *Let A be the adjacency matrix of the graph $G = \text{Cay}(F^r, S)$, where S may or may not include 0, then*

$$A = \Gamma_{h_1} + \Gamma_{h_2} + \dots + \Gamma_{h_n}.$$

Our aim at this point is to operate on submatrices of A , transforming it to a more convenient form while preserving the rank. With this in mind, we will define permutations derived from the matrix H but acting on subspaces of F^r .

For integers $s, t, 1 \leq s \leq t \leq r$ and a vector $x = (x_1, \dots, x_r) \in F^r$ let $x^{(s,t)} = (x_s, x_{s+1}, \dots, x_t)$. Given $\ell \leq r$ and a vector $u \in F^\ell$, let S_u be the set of suffixes of the columns whose prefix is u :

$$S_u = \{h^{(\ell+1,r)} \mid h \in S, h^{(1,\ell)} = u\}. \quad (1)$$

Finally, consider permutations $\Gamma_v, v \in F^{r-\ell}$ acting on $F^{r-\ell}$ by adding vectors from S_u . For a given u define the matrix

$$D_u = \sum_{v \in S_u} \Gamma_v. \quad (2)$$

Assume by definition that if $S_u = \emptyset$ then $D_u = 0_{2^{r-\ell} \times 2^{r-\ell}}$.

To give an example, let $r = 3, \ell = 1$, and take

$$H = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

then $S_0 = \{00, 10\}$, $S_1 = \{00, 11\}$ and

$$D_0 = I + \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad D_1 = I + \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix},$$

where we assumed that the rows and columns of the matrices are indexed by F^2 in lexicographical order.

Now let us look at how the matrices defined in (2) relate to the adjacency matrix of the graph G . For $\ell \in \{1, 2, \dots, r\}$, let $F^\ell = \{v_0, v_1, \dots, v_{2^\ell-1}\}$ where the vectors are ordered lexicographically. Starting with a small code C with a fixed parity-check matrix H , let us form the matrices $D_{v_0}, D_{v_1}, \dots, D_{v_{2^\ell-1}}$.

Lemma 11 For any $0 \leq \ell \leq r$ the matrix A can be written in the form

$$A = \begin{pmatrix} v_0 & v_1 & \dots & v_{2\ell-1} \\ v_0 & D_{v_0+v_0} & D_{v_0+v_1} & \dots & D_{v_0+v_{2\ell-1}} \\ v_1 & D_{v_1+v_0} & D_{v_1+v_1} & \dots & D_{v_1+v_{2\ell-1}} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ v_{2\ell-1} & D_{v_{2\ell-1}+v_0} & D_{v_{2\ell-1}+v_1} & \dots & D_{v_{2\ell-1}+v_{2\ell-1}} \end{pmatrix},$$

where the $2^{r-\ell} \times 2^{r-\ell}$ blocks $D_{v_i+v_j}$ are defined in (2).

Proof Given $x, y \in F^r$, we have $A_{x,y} = \sum_{h \in S} \mathbb{1}(x + h = y)$. Let $v_i = x^{(1,\ell)}$ and $v_j = y^{(1,\ell)}$ be the ℓ -prefixes of x and y and let D_{v_i,v_j} be the block in A at the intersection of the stripes v_i and v_j . Our goal is to show that $D_{v_i,v_j} = D_{v_i+v_j}$.

Let $S_{v_i+v_j}$ be the set defined in (1). Given $t, u \in F^{r-\ell}$, the element

$$(D_{v_i,v_j})_{t,u} = \mathbb{1}(t + h' = u),$$

where $h' \in S_{v_i+v_j}$ is an $(\ell - r)$ tail vector. Rephrasing and using (2),

$$D_{v_i,v_j} = \sum_{h' \in S_{v_i+v_j}} \Gamma_{h'} = D_{v_i+v_j}.$$

□

4.1 Zero-Codeword-Only Codes and Repetition Codes

As an example of using the above approach, consider the adjacency matrices of coset graphs of the zero-codeword-only codes and repetition codes. Their ranks are known [4, 8], but we rederive them using the tools developed in the previous sections. Let $J = \Gamma_{1_r}$ be the matrix defined as $J_{u,v} = \mathbb{1}(u + v = 1_r)$, where 1_r denotes the all-ones vector of length r . Since we assumed that the vectors in F^r are ordered lexicographically, J is antidiagonal.

Proposition 12 [8, Prop.9] Let $C = \{0_r\}$ be the zero-codeword code with S given by the standard basis $e_i, i = 1, \dots, r$. The adjacency matrix A_r of the coset graph $\text{Cay}(F^r, S)$ has rank 2^r if r is even and 2^{r-1} if r is odd.

Proof If r is odd, then $A_r = \sum_{i=1}^r \Gamma_{e_i}$ has full rank by Lemma 6(1). Let us consider the case of r even, writing the matrix C as in Lemma 11. Let $\ell = 1$ and note that $D_0 = A_{r-1}$ and $D_1 = I_{r-1}$, both of rank $r - 1$. Then

$$A_r = \begin{pmatrix} A_{r-1} & I \\ I & A_{r-1} \end{pmatrix}.$$

Since $s(A_{r-1})$ is odd, Lemma 7 implies that we can multiply the upper stripe by A_{r-1} block-by-block without affecting the row space in this part, thus with no effect on the rank. Upon multiplying, we obtain

$$\begin{pmatrix} I & A_{r-1} \\ I & A_{r-1} \end{pmatrix},$$

which obviously is of rank 2^{r-1} . \square

We could of course simply eliminate all the entries in one of the submatrices A_{r-1} by row operations, but the above procedure models our approach in other constructions. We next exemplify it in a more complicated case of repetition codes. Let C' be the repetition code of length $r+1$ and redundancy r defined by the parity-check matrix $H' = [I|1_r]$, where I is the identity matrix of order r . Form the matrix $H_r = [H'|0_r]$ and consider the code C of length $r+2$ for which H_r is a parity-check matrix. In the next lemma, we compute the rank of the adjacency matrix of the coset graph of C .

Proposition 13 *Let $S = \{e_1, \dots, e_r, 1_r, 0_r\}$. The adjacency matrix A_r of the coset graph $\text{Cay}(F^r, S)$ satisfies*

$$\text{rk}(A_r) = \begin{cases} 2^r & \text{if } r \text{ is odd} \\ \frac{1}{2}(2^r - 2^{\frac{r}{2}}) & \text{if } r \text{ is even.} \end{cases}$$

Proof If r is odd then $s(A_r)$ is odd, and by Lemma 6(1) A_r is a full-rank matrix, i.e., $\text{rk}(A_r) = 2^r$. For even r we prove the result by induction on r . Take $r = 2$, then $H_2 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$. Since $S = F^2$, the matrix A_2 is an all-ones matrix of rank 1, verifying the base case.

Let us assume that $\text{rk}(A_{r-2}) = \frac{1}{2}(2^{r-2} - 2^{\frac{r-2}{2}})$ and let us consider the matrix A_r . We decompose it into blocks taking $\ell = 2$ in the construction of Lemma 11. Observe that

$$D_{(0,0)} = A_{r-2} + J, \quad D_{(0,1)} = I, \quad D_{(1,0)} = I \text{ and } D_{(1,1)} = J.$$

By Lemma 11 we obtain

$$A_r = \begin{pmatrix} A_{r-2} + J & I & I & J \\ I & A_{r-2} + J & J & I \\ I & J & A_{r-2} + J & I \\ J & I & I & A_{r-2} + J \end{pmatrix}.$$

Next multiply the top horizontal stripe by $A_{r-2} + J$ and the bottom one by J , then we obtain the following matrix:

$$\begin{pmatrix} I & A_{r-2} + J & A_{r-2} + J & A_{r-2}J + I \\ I & A_{r-2} + J & J & I \\ I & J & A_{r-2} + J & I \\ I & J & J & A_{r-2}J + I \end{pmatrix}.$$

Since $s(A_{r-2} + J)$ is odd, by Lemma 7 this has no effect on the rank. Next, let us eliminate the bottom stripe adding to it the first three stripes row-by-row, and after that cancel two matrices in each of the two middle stripes. Overall we obtain

$$\begin{pmatrix} I & A_{r-2} + J & A_{r-2} + J & A_{r-2}J + I \\ 0 & 0 & A_{r-2} & A_{r-2}J \\ 0 & A_{r-2} & A_{r-2} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Let $D_1 = (A_{r-2}|A_{r-2}J)$ and $D_2 = (A_{r-2}|A_{r-2})$. By Lemma 9, $\text{rk}(D_1) = \text{rk}(D_2) = \text{rk}(A_{r-2})$. Therefore, considering the first three stripes independently, we obtain

$$\begin{aligned} \text{rk}(A_r) &= \frac{1}{4} \cdot 2^r + \frac{1}{2} \cdot (2^{r-2} - 2^{\frac{r-2}{2}}) + \frac{1}{2} \cdot (2^{r-2} - 2^{\frac{r-2}{2}}) \\ &= \frac{1}{2} \cdot \left(2^r - 2^{\frac{r}{2}}\right). \end{aligned}$$

□

5 A New Code Family

The starting point of the construction is the binary Hamming code of length $2^r - 1$, $r \geq 2$. Let \mathfrak{H}_r denote its parity-check matrix written in the standard form in which all the r -columns are ordered lexicographically with $0_{r-1}|1$ on the left and 1_r on the right. Since \mathfrak{H}_r contains all the nonzero columns, its coset graph is a complete graph K_{2^r} .

Lemma 14 *Consider the matrix $(\mathfrak{H}_r^\top | 0_{(2^r-1) \times m})^\top$, $m \geq 1$ and let S be the set of its columns. Let $G = \text{Cay}(F^{r+m}, S)$, then $\text{rk}(\tilde{A}(G)) = 2^m$.*

Proof We give two proofs, of which the second paves the way for later results in this section.

There are 2^r vertices $v \in F^{r+m}$ that share a common m -suffix. They form a clique K_{2^r} , which is a connected component of the graph G . Thus $\tilde{A}(G)$ is a $2^m \times 2^m$ block-diagonal matrix with each block of rank 1, and so its rank is 2^m .

Alternatively, with Lemma 11 in mind, let $\ell = r$ and note that for any fixed r -prefix u the set $S_u = \{0_m\}$. Thus the matrix $\tilde{A}(G)$ can be written as a $2^r \times 2^r$ block matrix with each block equal to I_{2^m} , confirming again that its rank is 2^m . □

The approach taken in [4], as well as in our work, is to add rows and columns to \mathfrak{H}_r in order to remove the codewords of weight 3 in the small code, while keeping the rank $\text{rk}(\tilde{A})$ low. The parity-check matrix H_s of the small code is formed of s rows at the top and some combinations of the matrices \mathfrak{H}_r underneath them. The resulting big codes are denoted by $\mathcal{C}_{s,r}$. The construction is recursive, starting with $s = 2$ (the base case) and adding one extra row to the top part in each step.

5.1 The Case of $s = 2$

Consider the following $(r + 2) \times (2^r + 2)$ parity-check matrix of the small code:

$$H_2 = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 1 & 1 \\ 0 & 1 & 1 & \dots & 1 & 0 & 1 \\ 0 & & & & 0 & 0 & \\ \vdots & & \mathfrak{H}_r & & \vdots & \vdots & \\ 0 & & & & 0 & 0 & \end{pmatrix}. \quad (3)$$

By inspection, any three columns of this matrix not including the first column are linearly independent, so the coset graph of the code $\ker(H_2)$ is triangle-free. The family of big codes $\mathcal{C}_{2,r}$ obtained from H_2 was originally discovered in [4] (up to a minor difference that is not essential for the results). As shown there, the rate of the storage code is $R(\mathcal{C}_{2,r}) = 3/4 - 2^{-r}$ for all $r \geq 4$. While the argument in [4] is somewhat complicated, here we give a straightforward proof based on the decomposition of Lemma 11 with $\ell = 2$.

Proposition 15 *Let S be the set of columns of the matrix H_2 and let $G_2 := \text{Cay}(F^r, S)$ be the coset graph of the code $\ker(H_2)$. Then $\text{rk}(A(G_2)) \leq 2^{r+2}/4 + 4$.*

Proof Let P be the adjacency matrix of K_{2^r} . We partition the adjacency matrix of G_2 into blocks of order 2^r according to the 2-prefix of the generator. The result can be written in the form

$$A(G_2) = \begin{pmatrix} I & P + I & I & I \\ P + I & I & I & I \\ I & I & I & P + I \\ I & I & P + I & I \end{pmatrix} = \begin{pmatrix} I & I & I & I \\ I & I & I & I \\ I & I & I & I \\ I & I & I & I \end{pmatrix} + \begin{pmatrix} 0 & P & 0 & 0 \\ P & 0 & 0 & 0 \\ 0 & 0 & 0 & P \\ 0 & 0 & P & 0 \end{pmatrix},$$

where the locations of P correspond to D_{01} ; see (2) and Lemma 11. Since $\text{rk}(P) = 1$, the result follows. \square

Corollary 16 *The rate of the storage code is*

$$R(\mathcal{C}_{2,r}) \geq 1 - \text{rk}(A)/2^{r+2} = \frac{3}{4} - 2^{-r} \rightarrow \frac{3}{4}$$

as $r \rightarrow \infty$.

Next we increase the dimensions of the matrix. We show this procedure in detail for $s = 3$ and then state and prove the general claim.

5.2 The Case of $s = 3$

Form the matrix $(H_2|H_2)$ and

(1) add a new row $0_{2^r+2}1_{2^r+2}$ at the top of this matrix.

We obtain an $(r+3) \times (2^{r+1}+4)$ matrix of the form

$$\left(\begin{array}{cccc|cccc|cccc} 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 & 1 & 1 & \dots & 1 & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 & 1 & 0 & 0 & 0 & \dots & 0 & 1 & 1 \\ 0 & 1 & 1 & \dots & 1 & 0 & 1 & 0 & 1 & 1 & \dots & 1 & 0 & 1 \\ 0 & & & & 0 & 0 & 0 & & & & & 0 & 0 \\ \vdots & & \mathfrak{H}_r & & \vdots & \vdots & \vdots & & \mathfrak{H}_r & & \vdots & \vdots \\ 0 & & 0 & 0 & 0 & 0 & 0 & & & & & 0 & 0 \end{array} \right). \quad (4)$$

This matrix contains linearly dependent triples that do not include the first column. To take care of them,

(2) add r rows of zeros at the bottom,
 (3) replicate the column $1|0_{2^r+2}$ immediately to the right of the vertical divider $2^r - 1$ times,
 (4) replace the $r \times (2^r - 1)$ matrix of zeros at the bottom of the new columns with \mathfrak{H}_r .

The resulting $(2r+3) \times (3 \cdot 2^r + 2)$ matrix H_3 is shown below in Eq. (5), with boxes around the rows and columns formed in steps (2)–(4).

$$H_3 = \left(\begin{array}{cc|cc|cc|cc|cc} 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 & 1 & \dots & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 & 1 & 0 & 0 & \dots & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & \dots & 1 & 0 & 1 & 0 & 0 & \dots & 0 & 1 & 1 & 0 \\ \hline 0 & & & & 0 & 0 & & & & & & 0 & 0 \\ \vdots & & \mathfrak{H}_r & & \vdots & \vdots & & 0 & & \mathfrak{H}_r & & \vdots & \vdots \\ 0 & & 0 & 0 & & & & & & & & 0 & 0 \\ \hline \boxed{0} & & 0 & 0 & & & & & & & & 0 & 0 \\ \vdots & & 0 & & \vdots & \vdots & & \mathfrak{H}_r & & 0 & & \vdots & \vdots \\ 0 & & 0 & 0 & & & & & & & & 0 & 0 \end{array} \right). \quad (5)$$

In the next proposition, we establish simple properties of the matrix H_3 . Denote by S the set of its columns.

Proposition 17 *The graph $G_3 = \text{Cay}(F^{2r+3}, S)$ is connected and triangle-free.*

Proof Let $S_i, i = 1, 2, 3$ be the sets of columns that contain the first, second, and third Hamming matrices, respectively, and let S_4 be the remaining set of 4 nonzero columns. Let $0 \in F^{2r+3}$ and $x \in F^{2r+3}, x \neq 0$ be two group elements. To prove the first claim, it suffices to show that there is a path in G_3 that connects them. Suppose $x = (x_1, x_2, x_3, x_4, \dots, x_{2r+3})$. Since S_4 contains a basis of F^3 , any assignment

$x_1x_2x_3$ of the first three coordinates in x can be reached independently of the remaining coordinates. Since the Hamming matrices contain all the nonzero columns, it is also possible to reach any vector of the form $(000, x_4, \dots, x_{2r+3})$.

For the second claim, we need to show that all triples of columns that do not include the zero column are linearly independent. This is shown by a straightforward case study. Let $h_1, h_2, h_3 \in S \setminus \{0\}$ be three such columns and let $b = |\{h_1, h_2, h_3\} \cap S_4|$. If $b \in \{1, 2, 3\}$ then the claim is obvious. If $b = 0$, then h_1, h_2, h_3 can be chosen to intersect one, two, or all three of the Hamming matrices. In each of these cases, direct inspection shows that the triples cannot add to zero. \square

Proposition 18 $\text{rk}(A(G_3)) \leq 2^{2r} + \frac{3}{2} \cdot 2^{r+3}$.

Proof Let P_1 be the adjacency matrix of the Cayley graph in F^{2r} whose generators are the columns of the matrix $(\mathfrak{H}_r^\top | 0_{(2r-1) \times r})^\top$ and 0_{2r} , and let P_2 be the same for the matrix $(0_{(2r-1) \times r} | \mathfrak{H}_r^\top)^\top$ and 0_{2r} . We arrange the matrix $A(G_3)$ in block form, where the blocks are indexed by binary 3-vectors (prefixes) ordered lexicographically. Using Lemma 11 for the matrix H_3 , we obtain

$$A(G_3) = \begin{pmatrix} I & P_1 + I & I & I & P_2 + I & P_1 + I & I & I \\ P_1 + I & I & I & I & P_1 + I & P_2 + I & I & I \\ I & I & I & P_1 + I & I & I & P_2 + I & P_1 + I \\ I & I & P_1 + I & I & I & I & P_1 + I & P_2 + I \\ P_2 + I & P_1 + I & I & I & I & P_1 + I & I & I \\ P_1 + I & P_2 + I & I & I & P_1 + I & I & I & I \\ I & I & P_2 + I & P_1 + I & I & I & I & P_1 + I \\ I & I & P_1 + I & P_2 + I & I & I & P_1 + I & I \end{pmatrix}.$$

The dimensions of this matrix are $N \times N$, with $N = 2^{2r+3}$, and each of the blocks is a square matrix of order 2^{2r} . We can write $A(G_3)$ as a sum of three matrices,

$$A(G_3) = A^{(1)} + A^{(2)} + A^{(3)}, \quad (6)$$

where $A^{(1)} = (I)_{8 \times 8}$ is the matrix formed of identity blocks, $A^{(2)} = \begin{pmatrix} B & B \\ B & B \end{pmatrix}$, and

$$B = \begin{pmatrix} 0 & P_1 & 0 & 0 \\ P_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & P_1 \\ 0 & 0 & P_1 & 0 \end{pmatrix}, \quad A^{(3)} = \begin{pmatrix} 0 & 0 & 0 & 0 & P_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & P_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & P_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & P_2 \\ P_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & P_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & P_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & P_2 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (7)$$

Now, Lemma 14 says that $\text{rk}(P_1) = \text{rk}(P_2) = 2^r$, and thus from (6), (7), the rank of $A(G_3)$ is at most

$$\text{rk}(A(G_3)) \leq 2^{2r} + 4 \cdot 2^r + 8 \cdot 2^r = 2^{2r} + \frac{3}{2} \cdot 2^{r+3}.$$

□

For obvious reasons, below we call matrices of the form $A^{(3)}$ *block permutation matrices*.

Corollary 19 *The rate of the storage code $\mathcal{C}_{3,r} = \ker(A(G_3))$ is*

$$R(\mathcal{C}_{3,r}) \geq 1 - \frac{2^{2r} + \frac{3}{2} \cdot 2^{r+3}}{2^{2r+3}} = \frac{7}{8} - \frac{3}{2} \cdot 2^{-r}.$$

The general induction step is not different from the transition from $s = 2$ to 3. Namely, we form the matrix H_s by performing steps (1)-(4) described above on the matrix H_{s-1} . As we will show shortly, this results in storage codes of rate $(2^s - 1)/2^s - o(1)$.

Following the construction procedure, we find that the matrix H_s is formed of s horizontal stripes, $s - 1$ of which contain zero matrices and several Hamming matrices \mathfrak{H}_r . The set of columns is formed of $2^{s-1} - 1$ vertical stripes each of which contains one Hamming matrix and $2^{s-1} + 1$ other columns counting the all-zeros column. Thus the dimensions of H_s are

$$((s-1)r + s) \times ((2^{s-1} - 1)(2^r - 1) + 2^{s-1} + 1).$$

Accordingly, the vertex set of the graph G_s is of size $N = 2^{(s-1)r+s}$. The $N \times N$ matrix $A(G_s)$ can be written as a $2^s \times 2^s$ block matrix with each block of size $2^{(s-1)r}$. The structure of the matrix is similar to $A(G_2)$ and $A(G_3)$: namely, it is a sum of several $2^s \times 2^s$ block matrices, one of which is formed of identity matrices only. The remaining matrices are block permutation matrices similar to the third term in (6), where in each of them, the block in question is the (augmented) adjacency matrix of the Cayley graph of the kind given in Lemma 14. Namely, the set of generators of this Cayley graph is the set of columns of the matrix

$$M_j := [0_{(2^r-1) \times (jr)} \mid \mathfrak{H}_r^\top \mid 0_{(2^r-1) \times ((s-2-j)r)}]^\top,$$

for some $j = 0, 1, \dots, s - 2$ (thus overall the matrices M_j are of dimensions $((s-1)r) \times (2^r-1)$). Lemma 14 implies that for each of these Cayley graphs, the adjacency matrix is of rank $2^{(s-2)r}$. Let us write the $N \times N$ matrix $A(G_s)$ as a sum similar to (6),

$$A(G_s) = \sum_{j=1}^s A^{(j)},$$

where $A^{(1)} = (I)_{2^s \times 2^s}$ is a matrix formed of identity blocks and $A^{(s)}$ is a block permutation matrix. Further, $\text{rk}(A^{(s)}) = 2^{(s-2)r+s}$, and for each of the matrices

A^{s-j} , $j = 1, 2, \dots, s-2$, the rank $\text{rk}(A^{(s-j)}) = \frac{1}{2} \text{rk}(A^{(s-j+1)})$. Combining these data, we find that

$$\begin{aligned} \text{rk}(A(G_s)) &\leq 2^{(s-1)r} + \left(1 + \frac{1}{2} + \dots + \frac{1}{2^{s-2}}\right)2^{(s-2)r+s} \\ &\leq 2^{(s-1)r} + 2^{(s-2)r+s+1} \\ &= N(2^{-s} + 2^{-r+1}). \end{aligned} \quad (8)$$

We obtain the following result.

Theorem 20 *For any $s \geq 2$ there exists a sequence of storage codes $\mathcal{C}_{s,r}$ on triangle-free graphs on $N = 2^{(s-1)r+s}$ vertices, $r = 4, 5, \dots$ of rate*

$$R(\mathcal{C}_{s,r}) \geq 1 - 2^{-s} - 2^{-r+1}. \quad (9)$$

Proof Since $R = 1 - \text{rk}(A(G_s))/N$, the claim about the rate follows from (8). \square

We have constructed an infinite family of code sequences $\mathcal{C}_{s,r}$, $r = 4, 5, \dots$; $s = 2, 3, \dots$ whose rates are given by (9). Taking $s, r \rightarrow \infty$, we obtain a sequence of codes of length $N = N(s, r)$ whose rate converges to one. This proves Theorem 2.

Acknowledgements We thank the editors for a careful reading of our manuscript and helpful suggestions. We are grateful to Ishay Haviv for drawing our attention to papers [7] and [10]. Following the release of our preprint, Hexiang Huang and Qing Xiang informed us of their forthcoming work (since then published as [15]) on another generalization of the construction of [4], also attaining code rate one. This research was partially supported by NSF grants CCF2110113 (NSF-BSF) and CCF2104489, and by a German Israeli Project Cooperation (DIP) Grant under Grant PE2398/1-1.

Data availability There is no data associated with this paper. Data availability statement is not applicable.

References

1. Alon, N., Lubetzky, E., Stav, U., Weinstein, A., Hassidim, A.: Broadcasting with side information. In: Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science, pp. 823–832 (2008). <https://doi.org/10.1109/FOCS.2008.41>
2. Arbabjolfaei, F., Kim, Y.-H.: Fundamentals of index coding. Found. Trends Commun. Inf. Theory **14**(3–4), 163–346 (2018). <https://doi.org/10.1561/0100000094>
3. Bar-Yossef, Z., Birk, Y., Jayram, T.S., Kol, T.: Index coding with side information. IEEE Trans. Inform. Theory **57**(3), 1479–1494 (2011). <https://doi.org/10.1109/TIT.2010.2103753>
4. Barg, A., Zémor, G.: High-rate storage codes on triangle-free graphs. IEEE Trans. Inform. Theory **68**(12), 7787–7797 (2022). <https://doi.org/10.1109/TIT.2022.3191309>
5. Cameron, P.J., Dang, A.N., Riis, S.: Guessing games on triangle-free graphs. Electron. J. Combin. **23**(1), 48 (2016). <https://doi.org/10.37236/4731>
6. Christofides, D., Markström, K.: The guessing number of undirected graphs. Electron. J. Combin. **18**(1), 192 (2011). <https://doi.org/10.37236/679>
7. Codenotti, B., Pudlák, P., Resta, G.: Some structural properties of low-rank matrices related to computational complexity. Theor. Comput. Sci. **235**(1), 89–107 (2000). [https://doi.org/10.1016/S0304-3975\(99\)00185-1](https://doi.org/10.1016/S0304-3975(99)00185-1)
8. Couvreur, A., Delfosse, N., Zémor, G.: A construction of quantum LDPC codes from Cayley graphs. IEEE Trans. Inform. Theory **59**(9), 6087–6098 (2013). <https://doi.org/10.1109/TIT.2013.2261116>

9. Gadouleau, M., Riis, S.: Graph-theoretical constructions for graph entropy and network coding based communications. *IEEE Trans. Inform. Theory* **57**(10), 6703–6717 (2011). <https://doi.org/10.1109/TIT.2011.2155618>
10. Golovnev, A., Haviv, I.: The (generalized) orthogonality dimension of (generalized) Kneser graphs: bounds and applications. In: 36th Computational Complexity Conference. Leibniz International Proceedings in Informatics, vol. 200, pp. 8.1–8.15. Schloss Dagstuhl – Leibniz Center for Informatics, Wadern (2021) <https://doi.org/10.4230/LIPIcs.CCC.2021.8>
11. Golovnev, A., Regev, O., Weinstein, O.: The Minrank of random graphs. *IEEE Trans. Inform. Theory* **64**(11), 6990–6995 (2018). <https://doi.org/10.1109/TIT.2018.2810384>
12. Gopalan, P., Huang, C., Simitci, H., Yekhanin, S.: On the locality of codeword symbols. *IEEE Trans. Inform. Theory* **58**(11), 6925–6934 (2011). <https://doi.org/10.1109/TIT.2012.2208937>
13. Haemers, W.H.: An upper bound for the Shannon capacity of a graph. In: Lovász, L., Sós, V. (eds.) *Algebraic Methods in Graph Theory*, Szeged, 1978. Colloquia Mathematica Societatis János Bolyai, vol. 25, pp. 267–272. North-Holland, Amsterdam (1981). <https://research.tilburguniversity.edu/en/publications/>
14. Haviv, I.: On Minrank and forbidden subgraphs. *ACM Trans. Comput. Theory* **11**(4), Art. 20 (2019). <https://doi.org/10.1145/3322817>
15. Huang, H., Xiang, Q.: Construction of storage codes of rate approaching one on triangle-free graphs. *Des. Codes Cryptogr.* **91**(12), 3901–3913 (2023). <https://doi.org/10.1007/s10623-023-01278-6>
16. Lubetzky, E., Stav, U.: Nonlinear index coding outperforming the linear optimum. *IEEE Trans. Inform. Theory* **55**(8), 3544–3551 (2009). <https://doi.org/10.1109/TIT.2009.2023702>
17. Mazumdar, A.: Storage capacity of repairable networks. *IEEE Trans. Inform. Theory* **61**(11), 5810–5821 (2015). <https://doi.org/10.1109/TIT.2015.2472521>
18. Mazumdar, A., McGregor, A., Vorotnikova, S.: Storage capacity as an information-theoretic vertex cover and the index coding rate. *IEEE Trans. Inform. Theory* **65**(9), 5580–5591 (2019). <https://doi.org/10.1109/TIT.2019.2910026>
19. Ramkumar, V., Balaji, S.B., Sasidharan, B., Vajha, M., Krishnan, M.N., Kumar, P.V.: Codes for distributed storage. *Found. Trends Commun. Inform. Theory* **19**, 547–813 (2022). <https://doi.org/10.1561/0100000115>
20. Riis, S.: Information flows, graphs and their guessing numbers. *Electron. J. Combin.* **14**, Research paper 44 (2007). <https://doi.org/10.37236/962>
21. Shanmugam, K., Dimakis, A.G.: Bounding multiple unicasts through index coding and locally repairable codes. In: 2014 IEEE International Symposium on Information Theory, pp. 296–300 (2014). <https://doi.org/10.1109/ISIT.2014.6874842>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.