

A nonlinear connectivity code larger than every linear code*

Matěj Kripner Václav Rozhoň Robert Šámal
Martin Šindelář Adrián Zámečník

Abstract

A connectivity code on a graph H is a family of edge sets of H whose pairwise symmetric differences are connected spanning subgraphs. Alon asked whether a nonlinear connectivity code can ever be larger than every linear one on the same graph. We answer this affirmatively: we construct a 4-regular graph on fifteen vertices on which the largest connectivity code has nine members and the largest linear connectivity code has eight. The code is obtained by lifting a quaternary code of length six with pairwise agreement one through clique gadgets.

1 Introduction

1.1 Background and the source problem

Classical minimum-distance conditions control how many coordinates distinguish two codewords. Graph codes instead prescribe the *shape* of the difference: when codewords are edge sets of a graph, one may require that any two of them differ in a spanning tree, a Hamiltonian cycle, a triangle, or some other structure. This point of view was developed by Alon, Gajiczer, Körner, Milojević, and Simonyi [1] and by Alon [3]; Alon’s survey [4] gives a broad account. Connectivity is the most basic structural requirement, since it asks that the difference cross every nontrivial vertex cut.

Let $H = (V, E)$ be a finite simple graph. Edge sets are identified with vectors in $\mathbb{F}_2^E$, and vector addition is symmetric difference, denoted by Δ .

Definition 1.1 (Connectivity codes; Alon [2]). A family $\mathcal{C} \subseteq 2^E$ is a *connectivity code* if $(V, A \Delta B)$ is connected for any distinct $A, B \in \mathcal{C}$. It is *linear* if it is a vector subspace of $\mathbb{F}_2^E$. Write $m(H)$ for the maximum size of a connectivity code and $m_{\text{lin}}(H)$ for the maximum size of a linear one.

Every codeword is regarded as a spanning subgraph, even if its edge set is empty. Connectivity is required of differences, not of individual codewords. Translating every codeword by one fixed edge set preserves all differences, so a code can always be normalized to contain the empty set.

Restricting a connectivity code to the edges of a nontrivial cut is injective, since otherwise some difference misses that cut. Hence $m(H) \leq 2^{\lambda(H)}$, where $\lambda(H)$ is the edge connectivity [2, Section 1]; for d -regular H this gives $m(H) \leq 2^d$. In each of the following exact results the optimum is attained by a linear code. For the complete graph, $m(K_n) = 2^{n-1}$ [1, Theorem 1], attained by the cut space of K_n . Bai, Gao, Ma, and Wu show $m(C_3 \square C_3) = 16$ [5], where $\square$ denotes the Cartesian product. Alon proves that every d -regular graph satisfying a mild cut-expansion hypothesis has a linear code of size 2^d [2, Theorem 1.1 and its proof], and Tian recently showed that for every $d \geq 4$ there is a d -regular graph with $m(H) = 2^d$ [12], again via linear codes. This motivates the following question.

Alon [2, Section 4, second concluding remark] asks whether nonlinear connectivity codes can be larger than linear codes on the same graph, and Tian [12, Section 6(c)] repeats the question, noting that his constructions are linear. In precise form:

*This work was produced within the Bolzano project (<https://bolzano.app>), using AI systems for mathematical research and manuscript preparation. The listed authors contributed to the project and the preparation of this release. The manuscript received only limited human checking.

Question 1.2. Is there a finite simple graph H for which $m(H) > m_{\text{lin}}(H)$?

The question asks for a strict separation on a single graph; it does not prescribe an asymptotic ratio or a graph family.

1.2 Main result

The graph has disjoint cliques $U_i = \{a_i, b_i, c_i, d_i\}$ and hubs h_i , for $i \in \mathbb{Z}/3\mathbb{Z}$. In addition to every edge inside each U_i , include

$$h_i a_i, \quad h_i b_i, \quad h_{i+1} c_i, \quad h_{i+1} d_i \quad (i = 0, 1, 2), \quad (1)$$

and no other edges. Denote the resulting graph by G .

Theorem 1.3. *The graph G in (1) is simple and 4-regular, has fifteen vertices and thirty edges, and satisfies*

$$m(G) = 9 > 8 = m_{\text{lin}}(G).$$

A nine-member code in G can contain the empty edge set.

Theorem 1.3 answers Question 1.2 affirmatively and determines both optima exactly. Since a linear code has power-of-two size, any code of size nine already separates; the content of the theorem is that nine is attainable in G and ten is not. We also show (Proposition 7.1) that fifteen vertices is the least order of a separating example within the doubled-six-cycle framework used here; smaller separating graphs outside that framework are not excluded.

1.3 Related work

Graph codes with structured differences. Alon, Gijgiczer, Körner, Milojević, and Simonyi [1] study families of graphs on $[n]$ whose pairwise differences are connected, k -connected, Hamiltonian, contain a spanning star, and so on, and they determine $m(K_n)$. Bai, Gao, Ma, and Wu [5] study how the maximum size changes as the class of admissible differences varies, establishing several phase transitions, and compute the torus value quoted above. Gishboliner, Jin, and Sudakov [8] solve a further problem from [1], determining the largest family of graphs on $[n]$ in which any two members have isomorphic differences. Alon [2] introduced connectivity codes on a fixed host graph and proved the expander theorem; Tian [12] completed the determination of the maximum over d -regular hosts.

Forbidden differences and intersecting families. A dual family of problems forbids a structure in the difference. Alon [3] considers families of graphs on $[n]$ in which no difference is a copy of a fixed graph F together with isolated vertices, and Versteegen [13] proves upper bounds for linear such codes; his introduction discusses whether linear codes may be optimal in that setting. These problems are closely related to intersecting families of graphs: Ellis, Filmus, and Friedgut [7] proved the Simonovits–Sós conjecture that a triangle-intersecting family of graphs on $[n]$ has at most $2^{\binom{n}{2}-3}$ members, and Berger and Zhao [6] proved the analogous statement for K_4 . Theorem 1.3 shows that, for connectivity codes on a fixed host, linear codes need not be optimal.

Linear versus nonlinear codes. In classical coding theory it has long been known that nonlinearity can pay. The Nordstrom–Robinson code [10] is a binary code of length 15 with 256 codewords and minimum distance 5, twice the size of the best binary linear code with the same parameters; this and the Kerdock and Preparata families were later explained as Gray-map images of linear codes over $\mathbb{Z}/4\mathbb{Z}$ [9]. The present construction is similarly driven by a code over a four-letter alphabet, although here the alphabet records pairs of edge indicators, and the obstruction to linearity is just the power-of-two cardinality.

Counting. The upper bound $m(G) \leq 9$ is a Plotkin-type double count [11, Theorem 1]: projected words over a four-letter alphabet must agree in at most one of six positions, and ten such words would create at least 48 coordinate agreements among only 45 pairs.

1.4 Proof overview

Contracting each clique U_i leaves a six-cycle alternating between clique parts and hubs, with two external edges at every cycle position; see Figure 1. A connected difference can miss at most one position. The auxiliary words record the two edge indicators at each position as one of four symbols; their pairwise differences therefore occupy five positions and connect the quotient along a path. The nine words arise from six partitions of the nine points of $\mathbb{F}_3^2$, arranged so that every pair of points shares exactly one block.

The internal edges require a more careful choice. Relabelling the symbols independently in each coordinate makes one distinguished external word zero without changing agreements. The cut space of K_4 has eight members, and every nonempty cut is connected and spanning. Assign its eight members bijectively to the other eight word indices, and toggle one fixed edge in each assignment. In a difference between two such indices, the toggled edge cancels, leaving a nonempty cut. This handles all differences except those against the distinguished zero word.

Against zero, the toggle can leave a disconnected internal graph. This is unavoidable: requiring every difference to be connected inside each clique would impose the complete-graph bound $m(K_4) = 8$. Instead, the coordinate labels are chosen so that each exceptional internal pattern receives the external edges that reconnect it through its adjacent hubs. Lemma 4.1 lists these repairs, and Theorem 5.2 shows that they assemble globally.

The matching linear construction uses six maps $\mathbb{F}_2^3 \rightarrow \mathbb{F}_2^2$ with distinct one-dimensional kernels. A nonzero input therefore makes at most one external position vanish, and vertex-bit cuts connect the clique interiors. This supplies eight codewords and completes the exact comparison.

Organization. Section 2 constructs the nine words (Lemma 2.1) and proves that ten are impossible (Lemma 2.2). Section 3 transfers this bound to G (Proposition 3.1). Sections 4 and 5 give the local repairs (Lemma 4.1) and the lift (Theorem 5.2). Section 6 gives the optimal linear code (Proposition 6.1), and Section 7 proves the conditional order bound (Proposition 7.1).

2 Nine words on a four-element alphabet

Put $X = \mathbb{F}_3^2$. All arithmetic used to define the following partitions is in $\mathbb{F}_3$. For $r \in \mathbb{F}_3$, the partition P_r has the triple

$$\{(r, b) : b \in \mathbb{F}_3\},$$

labelled 3, and the three pairs

$$\{(r+1, c), (r+2, c+1)\}, \quad c \in \mathbb{F}_3, \quad (2)$$

labelled c . The triple covers one row and the pairs cover the other two rows, each point once. For $t \in \mathbb{F}_3$, the partition Q_t has triple $\{(a, t) : a \in \mathbb{F}_3\}$, labelled 3, and pairs

$$\{(c, t+1), (c-1, t+2)\}, \quad c \in \mathbb{F}_3, \quad (3)$$

labelled c . These partition the remaining two columns. For $p \in X$, let $w(p)$ record its six block labels in the order

$$P_0, P_1, P_2, Q_0, Q_1, Q_2.$$

Thus every coordinate has symbol classes of sizes 3, 2, 2, 2.

Lemma 2.1. *The nine words $w(p)$ are distinct, and every two agree in exactly one coordinate.*

Proof. Agreement means that the two points belong to one block. Points with the same first coordinate a share the triple of P_a . They share no other triple, and no pair in (2) or (3) has equal first coordinates. Points with the same second coordinate share only the corresponding Q -triple, since every pair block has distinct second coordinates.

For points differing in both coordinates, interchange their order so they are (a, b) and $(a + 1, b')$. If $b' = b + 1$, they form the pair labelled b in P_{a-1} . If $b' = b - 1$, they form the pair labelled $a + 1$ in Q_{b+1} . These are the only possibilities in $\mathbb{F}_3$. A P -pair has difference $\pm(1, 1)$, and a Q -pair has difference $\pm(1, -1)$, so these possibilities do not overlap. The P index is fixed by the missing first-coordinate value; the Q index is fixed by the missing second-coordinate value. There is therefore exactly one shared block in this case too. No pair of distinct points shares all six labels, proving distinctness. $\square$

For reference, the complete auxiliary code is

p	P_0	P_1	P_2	Q_0	Q_1	Q_2
$(0, 0)$	3	2	0	3	1	0
$(0, 1)$	3	0	1	0	3	1
$(0, 2)$	3	1	2	1	0	3
$(1, 0)$	0	3	2	3	2	1
$(1, 1)$	1	3	0	1	3	2
$(1, 2)$	2	3	1	2	1	3
$(2, 0)$	2	0	3	3	0	2
$(2, 1)$	0	1	3	2	3	0
$(2, 2)$	1	2	3	0	2	3

Lemma 2.2. *A four-symbol code of length six in which distinct words agree at most once has at most nine words.*

Proof. If there were ten or more, select ten. At one coordinate let the four symbol multiplicities be n_0, n_1, n_2, n_3 , summing to ten. The number of agreeing pairs there is $\sum_s \binom{n_s}{2}$. If $n_s \geq n_t + 2$, moving one occurrence from s to t decreases this sum by $n_s - n_t - 1 > 0$. A minimum consequently has multiplicities 3, 3, 2, 2 and value eight. The six coordinates contribute at least 48 pair-coordinate agreements. There are only $\binom{10}{2} = 45$ pairs, each allowed at most one agreement, a contradiction. $\square$

This elementary agreement count and Lemma 2.1 determine the auxiliary optimum exactly. The graph construction below uses more than this optimum: it also uses the fact that every symbol class has at least two points.

3 The graph and its universal upper bound

The graph G has $3 \cdot 4 + 3 = 15$ vertices. Its internal edges number $3\binom{4}{2} = 18$, and (1) contributes twelve more. All edges have distinct endpoints and no listed pairs repeat. Each clique vertex has three internal neighbors and one hub neighbor. The neighbors of h_i are $a_i, b_i, c_{i-1}, d_{i-1}$. This proves simplicity and regularity in Theorem 1.3.

For an edge set A , record each external pair as a binary pair:

$$\begin{aligned}\pi(A)_{2i} &= (\mathbf{1}_{h_i a_i \in A}, \mathbf{1}_{h_i b_i \in A}), \\ \pi(A)_{2i+1} &= (\mathbf{1}_{h_{i+1} c_i \in A}, \mathbf{1}_{h_{i+1} d_i \in A}).\end{aligned}\tag{4}$$

The six coordinates follow the cyclic order

$$h_0, U_0, h_1, U_1, h_2, U_2, h_0.\tag{5}$$

Here each U_i becomes a single vertex when the clique is identified.

Proposition 3.1. *Every connectivity code in G has at most nine members, and every linear one has at most eight.*

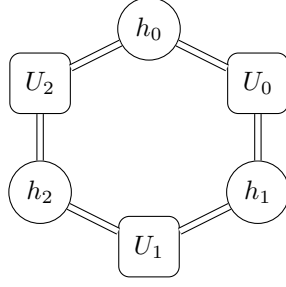


Figure 1: The external quotient. Each box represents a complete graph on four vertices, not a single vertex of G . Each double link represents the two distinct external edges in (1).

Proof. Suppose $A \triangle B$ is connected and spanning. Identify each U_i to one vertex, delete loops, and merge parallel edges. Connectivity is preserved by these operations. The resulting graph is a connected spanning subgraph of the six-cycle (5). A position is occupied exactly when $\pi(A)$ and $\pi(B)$ differ there. At least five positions must be occupied, since a connected graph on six vertices has at least five edges.

Hence π is injective on any connectivity code, and its images agree in at most one coordinate. The alphabet $\mathbb{F}_2^2$ has four elements, so Lemma 2.2 gives the upper bound nine. The cardinality of a binary vector subspace is a power of two; its size is therefore at most eight. $\square$

4 A four-vertex gadget and its repairs

For a vertex subset S of a graph K , write $\delta_K(S)$ for the cut consisting of edges with one endpoint in S . Edgewise parity gives

$$\delta_K(S) \triangle \delta_K(T) = \delta_K(S \triangle T). \quad (6)$$

If K is connected, its empty cut comes only from $\emptyset$ and the full vertex set. Thus two subsets have the same cut exactly when they are equal or complementary. The cut space of K_4 has eight members: the empty cut, four singleton cuts, and three cuts of type $2 + 2$. Every nonempty cut is a connected spanning complete bipartite graph.

These eight cuts can label eight of the desired nine edge sets. Toggling the same internal edge in each label preserves all differences between them. The remaining task is to make their differences with a distinguished empty edge set connected. For an empty internal cut, even all four external edges give two disjoint stars, one at each hub. The edge ac joins these stars. Toggling it causes two other cuts to lose internal connectivity, but each can be repaired by a full external pair on one side. The next lemma records these three repairs.

Lemma 4.1. *Let K be the clique on a, b, c, d , and put $e = ac$. Add distinct possible hubs L, R , with any selected edges from $\{La, Lb\}$ and $\{Rc, Rd\}$. Include a hub as a vertex only if at least one of its edges is selected. Starting with the internal edge set $F \triangle \{e\}$ for a cut F of K , the resulting local graph is connected in each of these situations:*

- (i) $F \notin \{\emptyset, \delta_K(\{a\}), \delta_K(\{c\})\}$, with arbitrary external selections;
- (ii) $F = \emptyset$, with all four external edges selected;
- (iii) $F = \delta_K(\{a\})$, with both right edges selected;
- (iv) $F = \delta_K(\{c\})$, with both left edges selected.

Proof. For the five cuts in (i), the internal graph is connected. The singleton cuts at b, d do not contain ac , so the operation adds an edge to a spanning star. Each $2 + 2$ cut is a $K_{2,2}$; adding ac preserves connectivity if it was absent, while deleting it leaves a spanning path if it was present. Every included hub attaches to this connected internal graph.

In (ii), the internal graph is the single edge ac , and the external edges supply the spanning path b, L, a, c, R, d . In (iii), the internal edges are ab, ad , and the path b, a, d, R, c connects the four clique vertices and R . Any included L attaches at a or b . In (iv), the internal edges are bc, cd , and a, L, b, c, d is a spanning path through the clique and L . Any included R attaches at c or d . These paths establish each case. $\square$

The lemma permits an internal difference to be disconnected when the external edges repair it. Requiring all nine codewords to have connected pairwise differences within a single clique would be impossible: restriction to the three edges incident with any vertex of K_4 is injective on such a family, bounding its size by $2^3 = 8$.

5 Lifting the nonlinear code

Fix $o = (0, 0) \in X$. Coordinates are numbered $0, \dots, 5$ in the order used above. For each $i \in \{0, 1, 2\}$, select $p_i \neq o$ whose word disagrees with $w(o)$ at both coordinates $2i, 2i + 1$. Such a point exists: at each coordinate at most two other points share o 's symbol, so the two coordinates exclude at most four of the eight candidates.

Let D_i^L, D_i^R be the symbol classes of p_i at these two coordinates. Each has at least two points, neither contains o , and

$$D_i^L \cap D_i^R = \{p_i\}. \quad (7)$$

Indeed, any second point in the intersection would agree with p_i twice. Choose $q_i \in D_i^R \setminus \{p_i\}$ and $r_i \in D_i^L \setminus \{p_i\}$. They are distinct from each other, from p_i , and from o . The following choices can be read directly from the auxiliary-word table:

i	Coordinates	p_i	q_i	r_i
0	P_0, P_1	(1, 0)	(1, 1)	(2, 1)
1	P_2, Q_0	(0, 1)	(2, 2)	(1, 2)
2	Q_1, Q_2	(0, 1)	(1, 0)	(1, 1)

Different gadgets may use the same points; no cross-gadget distinctness is required.

For $j \in \{2i, 2i + 1\}$ choose a bijection $\gamma_j : \{0, 1, 2, 3\} \rightarrow \mathbb{F}_2^2$ with

$$\gamma_j(w(o)_j) = 00, \quad \gamma_j(w(p_i)_j) = 11.$$

The two specified symbols are different, so this leaves exactly the two other binary pairs to assign. Set $\tilde{w}(p)_j = \gamma_j(w(p)_j)$. Agreements are unchanged, $\tilde{w}(o)$ is zero, and

$$\tilde{w}(p_i)_{2i} = \tilde{w}(p_i)_{2i+1} = 11, \quad \tilde{w}(q_i)_{2i+1} = 11, \quad \tilde{w}(r_i)_{2i} = 11. \quad (8)$$

Each coordinate belongs to just one gadget pair, so the six bijections can be chosen independently.

Let $K_i = G[U_i]$ and let $\mathcal{D}_i$ be its eight-element cut space. Choose a bijection $F_i : X \setminus \{o\} \rightarrow \mathcal{D}_i$ such that

$$F_i(p_i) = \emptyset, \quad F_i(q_i) = \delta_{K_i}(\{a_i\}), \quad F_i(r_i) = \delta_{K_i}(\{c_i\}). \quad (9)$$

The three prescribed points and cuts are distinct, so this extends by matching the remaining five points to the remaining five cuts. These choices are finite and explicit. For a completely fixed convention, send the remaining two symbols in increasing order to 01, 10, and send the five remaining points in lexicographic order to the remaining cuts in the lexicographic order of their three vertex bits on b_i, c_i, d_i , with the bit on a_i fixed to zero.

Set $A_o = \emptyset$. For $p \neq o$ define

$$A_p \cap E(K_i) = F_i(p) \triangle \{a_i c_i\}, \quad \pi(A_p) = \tilde{w}(p). \quad (10)$$

Internal and external edges are disjoint, so these prescriptions define every edge of A_p .

Example 5.1. Consider the word indexed by $p = (1, 0)$ in the first gadget. Its symbols at P_0, P_1 are 0, 3, while o has symbols 3, 2. The specified relabellings therefore give $\tilde{w}(p)_0 = \tilde{w}(p)_1 = 11$. Since $p = p_0$, its internal cut is empty, and its internal edge set after the toggle is just $a_0 c_0$. This single edge, together with the four external edges, supplies the path

$$b_0, h_0, a_0, c_0, h_1, d_0.$$

Thus the restriction to the clique is disconnected, but the local subgraph containing its two hubs is connected. This is the exceptional repair that allows the ninth codeword.

Theorem 5.2. *The nine sets A_p , for $p \in X$, form a connectivity code in G .*

Proof. The external words are distinct, so all nine sets are distinct. First take $p, q \neq o$ with $p \neq q$. Inside K_i , the toggled edge cancels and the difference is $F_i(p) \triangle F_i(q)$. By (6) it is a cut; by injectivity of F_i it is nonempty. It is therefore connected and spanning inside every clique. The external words agree exactly once, so their difference occupies five cycle positions and connects the six quotient vertices along a path. Replacing each clique vertex of that path by its connected internal graph proves that $A_p \triangle A_q$ is connected and spans G .

It remains to compare a nonzero index p with o . The external word $\tilde{w}(p)$ agrees with the zero word exactly once. Precisely one external position is absent. For each i , let T_i consist of all four clique vertices, the edges prescribed in (10), and those adjacent hubs incident with a present external edge. If $p = p_i$, both external pairs are full by (8), and Lemma 4.1(ii) applies. If $p = q_i$, the right pair is full and part (iii) applies. If $p = r_i$, the left pair is full and part (iv) applies. For every other p , bijectivity in (9) excludes the three exceptional cuts, so part (i) applies. Thus each T_i is connected and contains its whole clique.

Let k be the unique gadget incident with the absent cycle position. The two gadgets $i \neq k$ each have both external positions occupied. Their connected local graphs join h_i to h_{i+1} ; these two hub pairs form a path through all three hubs. Every gadget has at least one occupied external position, since only one is absent globally. Therefore every T_i meets this connected hub component. All fifteen vertices lie in one component, proving that $A_p = A_p \triangle A_o$ is connected and spanning. $\square$

Together with Proposition 3.1, this proves $m(G) = 9$.

6 The optimal linear code

Proposition 6.1. *The graph G has a three-dimensional linear connectivity code. Consequently $m_{\text{lin}}(G) = 8$.*

Proof. For $u = (x, y, z) \in \mathbb{F}_2^3$, prescribe the external word

$$L(u) = ((y, z), (x, z), (x, y), (x + y, z), (x + z, y), (y + z, x)). \quad (11)$$

The six coordinate maps have kernels generated respectively by

$$(1, 0, 0), (0, 1, 0), (0, 0, 1), (1, 1, 0), (1, 0, 1), (0, 1, 1).$$

These vectors are distinct. Thus any nonzero input makes at most one binary-pair coordinate zero. At least five external cycle positions are occupied.

Inside each clique assign the vertex bits $(0, x, y, z)$ to (a_i, b_i, c_i, d_i) and select an edge when its two bits differ. For nonzero u the bits are not constant, so the selected internal edges form a nonempty cut and are connected and spanning. With the external prescription (11), let the resulting edge set be B_u . Its internal graphs connect through a spanning path of occupied quotient positions, so B_u is connected and spanning whenever $u \neq 0$.

Every edge indicator is a linear function of u : internally it is the sum of its endpoint bits, and externally it is a coordinate of (11). Hence $B_u \triangle B_v = B_{u+v}$. No nonzero input maps to the empty set, because its image is connected and spans fifteen vertices. The map is therefore injective, and its image is a three-dimensional linear code with eight members. Proposition 3.1 gives the matching upper bound. $\square$

This completes Theorem 1.3. The existence question is answered without assuming that fifteen is the smallest possible graph order for a separation.

7 Order within the doubled-six-cycle framework

The construction does have an order-minimality property under explicit hypotheses.

Proposition 7.1. *Let J be a simple graph partitioned into six nonempty parts $V_0, \dots, V_5$, indexed cyclically. Suppose exactly two edges join each consecutive pair and no other edges join different parts. If $m(J) \geq 9$, then $|V(J)| \geq 15$. The graph G attains this bound in this class.*

Proof. Restriction to the edges incident with a vertex v is injective on any connectivity code: a difference with zero restriction isolates v , whereas the graph has at least six vertices. A code with nine members therefore forces $9 \leq 2^{d_J(v)}$, and hence $d_J(v) \geq 4$ for every vertex.

A part of size t has exactly four outgoing edges. Simplicity bounds its internal edge count by $\binom{t}{2}$. Summing degrees over that part gives

$$4t \leq \sum_{v \in V_i} d_J(v) = 2|E(J[V_i])| + 4 \leq t(t-1) + 4.$$

Thus $(t-1)(t-4) \geq 0$, so a nonempty part has size one or at least four. Adjacent parts cannot both be singletons, since two distinct edges cannot join the same two vertices in a simple graph. At most three parts are therefore singletons. If their number is $s \leq 3$, then

$$|V(J)| \geq s + 4(6-s) = 24 - 3s \geq 15.$$

For G , the alternating parts $\{h_0\}, U_0, \{h_1\}, U_1, \{h_2\}, U_2$ meet all these hypotheses and have total size fifteen. Theorem 5.2 gives the required nine codewords. $\square$

8 Concluding remarks

The strict inequality is an existence result of the type asked in Question 1.2, with exact optima on a specified graph. It does not establish a growing nonlinear-to-linear ratio or characterize all graphs admitting a separation. The conditional order bound also does not exclude smaller examples outside the doubled-six-cycle framework, or examples whose optimal code has another size. The local repair shows that connected differences need not be assembled from independently connected internal pieces; their external incidence can supply the missing connections.

References

- [1] N. Alon, A. Gujgiczer, J. Körner, A. Milojević, and G. Simonyi, *Structured codes of graphs*, SIAM Journal on Discrete Mathematics **37** (2023), no. 1, 379–403.
<https://doi.org/10.1137/22M1487989>.
- [2] N. Alon, *Connectivity graph-codes*, Random Structures & Algorithms **65** (2024), no. 3, 451–459. <https://doi.org/10.1002/rsa.21222>. Revised author manuscript:
<https://web.math.princeton.edu/~nalon/PDFS/GCII3.pdf>.
- [3] N. Alon, *Graph-codes*, European Journal of Combinatorics **116** (2024), article 103880.
<https://doi.org/10.1016/j.ejc.2023.103880>. <https://arxiv.org/abs/2301.13305>.
- [4] N. Alon, *Graph-codes: questions, results and methods*, Japanese Journal of Mathematics (2026), published online 12 July 2026. <https://doi.org/10.1007/s11537-026-2536-1>.
- [5] B. Bai, Y. Gao, J. Ma, and Y. Wu, *Phase transitions of structured codes of graphs*, SIAM Journal on Discrete Mathematics **38** (2024), no. 2, 1902–1914.
<https://doi.org/10.1137/23M1614572>. Preprint: <https://arxiv.org/abs/2307.08266v1>.
- [6] A. Berger and Y. Zhao, *K_4 -intersecting families of graphs*, Journal of Combinatorial Theory, Series B **163** (2023), 112–132. <https://doi.org/10.1016/j.jctb.2023.07.005>.
- [7] D. Ellis, Y. Filmus, and E. Friedgut, *Triangle-intersecting families of graphs*, Journal of the European Mathematical Society **14** (2012), no. 3, 841–885.
<https://doi.org/10.4171/JEMS/320>.
- [8] L. Gishboliner, Z. Jin, and B. Sudakov, *Difference-isomorphic graph families*, Journal of Combinatorial Theory, Series B **180** (2026), 25–61. <https://doi.org/10.1016/j.jctb.2026.06.001>.
- [9] A. R. Hammons, P. V. Kumar, A. R. Calderbank, N. J. A. Sloane, and P. Solé, *The $\mathbb{Z}_4$ -linearity of Kerdock, Preparata, Goethals, and related codes*, IEEE Transactions on Information Theory **40** (1994), no. 2, 301–319. <https://doi.org/10.1109/18.312154>.
- [10] A. W. Nordstrom and J. P. Robinson, *An optimum nonlinear code*, Information and Control **11** (1967), no. 5–6, 613–616. [https://doi.org/10.1016/S0019-9958\(67\)90835-2](https://doi.org/10.1016/S0019-9958(67)90835-2).
- [11] M. Plotkin, *Binary codes with specified minimum distance*, IRE Transactions on Information Theory **6** (1960), no. 4, 445–450. <https://doi.org/10.1109/TIT.1960.1057584>.
- [12] C. Tian, *Maximal connectivity graph-codes exist in every degree at least four: $f(d) = 2^d$ for all $d \geq 4$, answering a question of Alon*, arXiv:2609.02953v1, 2 September 2026, Section 6(c). Listed on arXiv as *Alon’s Question on Connectivity Graph-Codes: $f(d) = 2^d$ for Every $d \geq 4$* .
<https://arxiv.org/abs/2609.02953v1>.
- [13] L. Versteegen, *Upper bounds for linear graph codes*, Random Structures & Algorithms **66** (2025), no. 1, article e21263. <https://doi.org/10.1002/rsa.21263>.
<https://arxiv.org/abs/2310.19891>.