Index Coding with Side Information

Ziv Bar-Yossef Yitzhak Birk T. S. Jayram Tomer Kol

***Abstract*—Motivated by a problem of transmitting supplemental data over broadcast channels (Birk and Kol, INFOCOM 1998), we study the following coding problem: a sender communicates with n receivers $R_1, \ldots, R_n$. He holds an input $x \in \{0,1\}^n$ and wishes to broadcast a single message so that each receiver R_i can recover the bit x_i. Each R_i has prior *side information* about x, induced by a directed graph G on n nodes; R_i knows the bits of x in the positions $\{j \mid (i,j)$ is an edge of $G\}$. G is known to the sender and to the receivers. We call encoding schemes that achieve this goal INDEX *codes for* $\{0,1\}^n$ *with side information graph* G.**

In this paper we identify a measure on graphs, the *minrank*, which exactly characterizes the minimum length of linear and certain types of non-linear INDEX codes. We show that for natural classes of side information graphs, including directed acyclic graphs, perfect graphs, odd holes, and odd anti-holes, minrank is the optimal length of *arbitrary* INDEX codes.

For arbitrary INDEX codes and arbitrary graphs, we obtain a lower bound in terms of the size of the maximum acyclic induced subgraph. This bound holds even for randomized codes, but is shown not to be tight.

***Index Terms*—Error correction coding, broadcast channels, code length, information cost.**

I. Introduction

Source coding is one of the central areas of coding and information theory. Shannon's famous source coding theorem states that the average number of bits necessary and sufficient to encode a source is equal (up to one bit) to the entropy of the source. In many distributed applications, though, the receiver may have some prior *side information* about the source message, before it is sent. Source coding with side information addresses encoding schemes that exploit the side information in order to reduce the length of the code. Classical results in this area [1], [2], [3] describe how to achieve optimal rates with respect to the joint entropy of the source and the side information.

Witsenhausen [4] initiated the study of the zero-error side information problem. For every source input $x \in \mathcal{X}$, the receiver gets an input $y \in \mathcal{Y}$ that gives some information about x. This is captured by restricting the pairs (x,y) to belong to a fixed set $\mathcal{L} \subseteq \mathcal{X} \times \mathcal{Y}$. Both the sender and the receiver know $\mathcal{L}$, and thus each of them, given his own input, has information about the other's input. Witsenhausen showed that fixed-length side information codes are equivalent to colorings of a related object called the *confusion graph*, and thus the logarithm of the chromatic number of this graph tightly characterizes the minimum number of bits needed to encode the source. Further results by Alon and Orlitsky [5] and Koulgi *et al.* [6] showed that graph-theoretic information measures could be used to characterize both the average length of variable-length codes, as well as asymptotic rates of codes that simultaneously encode multiple inputs drawn from the same source.

In this paper, we study a new variant of source coding with side information, first proposed by Birk and Kol [7] in the context of a server that disseminates a set of data blocks over a broadcast channel to a set of caching clients. Each client possesses in its cache only a subset of the transmitted blocks, due to reception problems, limited storage space, rejection by an interest filter, etc. The client needs a certain subset of the data blocks, yet some of these blocks may be missing from its cache. The client uses a backward channel to request blocks that it needs but has not cached and to advise the server of the blocks it already has in its cache. The challenge is to minimize the amount of supplemental information that must be broadcast by the server in order to enable every client to derive all its requested blocks. See Section II for more details on this problem. In [7], the idea of coding on demand by an informed source (ISCOD) is proposed and explored. Specifically, a heuristic algorithm is used to judiciously partition the set of clients into subsets, and the requests of the clients in each subset are handled using an erasure correcting code such that each member of any given subset is able to derive the union of the blocks requested by that subset's members. This approach is related to the notion of a graph cover by "partial" cliques, and is shown to generally not provide the optimal solution even with an optimal partition. The paper first considers the case wherein each client requests a single, unique block, and then shows a simple reduction that represents a client requesting multiple blocks as several single-request clients. (Multiple requests for the same block are prohibited.)

The above scenario is formalized as a source coding with side information problem as follows (cf. [7]). There is a sender (server) who has an input (data) x from a source alphabet $\mathcal{X} = \{0,1\}^n$ (we assume here single bit blocks; see more details in Section II). There are n receivers (clients) $R_1, \ldots, R_n$, where for each i, R_i is interested in the bit (requested block) x_i. The side information is characterized by a simple directed graph G (no self loops or parallel edges) on $\{1, 2, \ldots, n\}$. For a subset $S \subseteq [n]$, $x[S]$ denotes the projection of x on the coordinates in S. The side information of R_i (cached blocks) equals $x[N(i)]$, where $N(i) \triangleq \{j \in V \mid (i,j)$ is an edge$\}$ denotes the set of out-neighbors of i in the graph G. The sender and the receivers are both assumed to know G.

An extended abstract of this paper appeared at the 47th IEEE Annual Symposium on Foundations of Computer Science (FOCS), pages 197–206, 2006.

Department of Electrical Engineering, Technion and Google Haifa Engineering Center, Haifa, Israel. Email: `zivby@ee.technion.ac.il`. Supported by the European Commission Marie Curie International Re-integration Grant.

Department of Electrical Engineering, Technion, Haifa 32000, Israel. Email: `birk@ee.technion.ac.il`.

IBM Almaden Research Center, 650 Harry Road, San Jose 95120, CA, USA. Email: `jayram@almaden.ibm.com`.

Department of Electrical Engineering, Technion and IBM Haifa Research Laboratory, Haifa 32000, Israel. Email: `tomer@tx.technion.ac.il`.

Example 1. For an input $x \in \{0,1\}^n$, each receiver R_i is interested in the value x_i but knows x_{i-1} as side information. (Abusing notation slightly, receiver R_1 knows x_n.) The side information graph is thus a directed cycle of length n. Since x_{i-1} is "independent" of x_i, it may not be clear at first how the sender can take advantage of the side information of the receivers to shorten the broadcast message. However, there is a strategy in which the sender can save one bit: rather than sending all the bits of x, the sender broadcasts the $n-1$ parities $x_1 \oplus x_2, x_2 \oplus x_3, \ldots, x_{n-1} \oplus x_n$. Now, each receiver R_i for $i > 1$ can recover x_i by taking the parity of $x_{i-1} \oplus x_i$ with x_{i-1}. Finally, receiver R_1 XORS the $n-1$ parities broadcast by the sender together with x_n to recover x_1.

Definition 2 (INDEX codes). A deterministic INDEX code $\mathcal{C}$ for $\{0,1\}^n$ with side information graph G on n nodes, abbreviated as "INDEX code for G", is a set of codewords in $\{0,1\}^\ell$ together with:

1) An encoding function E mapping inputs in $\{0,1\}^n$ to codewords, and
2) A set of decoding functions $D_1, D_2, \ldots D_n$ such that $D_i(E(x), x[N(i)]) = x_i$ for every i.

The graph G is known in advance to the sender and the receivers; thus the encoding and decoding functions typically depend on G. The *length* of $\mathcal{C}$, denoted by $\mathrm{len}(\mathcal{C})$, is defined to be ℓ.

The above problem can also be cast in an equivalent setting with a single receiver: The receiver is given an index i and the side information $x[N(i)]$ as inputs and wants to recover the value x_i. (The equivalence follows from the fact the sender does not know the index i given to the receiver, and thus has to use an encoding that enables the recovery of x_i, for *any* i.) Using this equivalent form, we can contrast our side information problem with Witsenhausen's zero-error side information problem. A first notable difference is that while in Witsenhausen's setting the *entire* input x has to be recovered, in our setting only a single bit x_i is needed. This allows significant savings in the encoding length, as the following example demonstrates: suppose the side information graph is a perfect matching on n nodes. Since the receiver has only a single bit of side information, then $n-1$ bits are necessary to recover the entire input. If, however, only a single bit is needed, then the sender can encode his input by the $n/2$ parities of pairs of matched bits. A second difference from Witsenhausen's setting is that the type of side information addressed in our problem is restricted to side information graphs. This natural restriction emanates from the broadcast application mentioned above and also imposes more structure that enables us to obtain an interesting combinatorial characterization of the minimum length of INDEX codes in terms of the side information graphs.

We also consider in this paper randomized INDEX codes, in which the encoding and decoding functions are allowed to be randomized and are even allowed to use a common public random string. Decoding needs to succeed only with *high probability*, taken over the random choices made by the encoding and decoding functions.

Our contributions. In this paper we identify a graph functional, called *minrank*, which we show to characterize the minimum length of INDEX codes, for natural types of codes and for wide classes of side information graphs. Let G be a directed graph on n vertices without self-loops. We say that a 0-1 matrix $A = (a_{ij})$ *fits* G if for all i and j: (i) $a_{ii} = 1$, and (ii) $a_{ij} = 0$ whenever (i,j) is *not* an edge of G. Thus, $A - I$ is the adjacency matrix of an *edge subgraph* of G, where I denotes the identity matrix. Let $\mathrm{rk}_2(\cdot)$ denote the 2-rank of a 0-1 matrix, namely, its rank over the field $GF(2)$.

Definition 3. $\mathrm{minrk}_2(G) \triangleq \min\{\mathrm{rk}_2(A) : A \text{ fits } G\}$.

The above measure for *undirected* graphs was considered by Haemers [8] in the context of proving bounds for the Shannon capacity Θ of undirected graphs. For an undirected graph G whose adjacency matrix is M, the 2-rank of $M + I$ (which fits G) has also been studied in the algebraic graph theory community. For example, Brouwer and van Eijl [9] and Peeters [10] study this quantity for strongly regular and distance-regular graphs, respectively. It has been shown by Peeters [11] that computing $\mathrm{minrk}_2(G)$ is NP-hard. Finally, it is known that minrk_2 has the "sandwich property", similar to other natural quantities such as the Lovász Theta function:

Proposition 4 ([12], [8]). *For any undirected graph G, $\omega(\overline{G}) \leq \Theta(G) \leq \mathrm{minrk}_2(G) \leq \chi(\overline{G})$, where $\overline{G}$ is the complement of G and $\omega(\cdot)$, $\Theta(\cdot)$, and $\chi(\cdot)$ are, respectively, the clique number, the Shannon capacity, and the chromatic number. Moreover, each of these inequalities is strict.*

Our first result (see Section III) shows that $\mathrm{minrk}_2(G)$ completely characterizes the minimum length of *linear* INDEX codes (i.e., ones whose encoding function is linear), for *arbitrary* directed side information graphs G:

Theorem 5. *For any side information graph G, there exists a linear INDEX code for G whose length equals $\mathrm{minrk}_2(G)$. This bound is optimal for all linear INDEX codes for G.*

This bound strictly improves a previous upper bound of Birk and Kol [7]. Birk and Kol showed a construction of a linear INDEX code, whose length is the "cover cost" of the side information graph (and showed that the construction is suboptimal). For undirected graphs, the cover cost is the same as the chromatic number of the complement graph. Since the minrank can be strictly smaller than this chromatic number, it immediately follows that the minrank bound beats the Birk and Kol bound. The lower bound for linear codes is of interest, since linear codes are possibly the most natural type of codes. In fact, all the existing INDEX codes (with or without side information) we are aware of are linear.

In Section IV we prove that $\mathrm{minrk}_2(G)$ characterizes not only the optimal length of linear codes, but also the optimal length of a wide class of *non-linear* codes. An INDEX code is called *linearly-decodable*, if all its n decoding functions are linear. A linearly-decodable code need not be linearly encodable. A simple argument shows that the length of a linearly-decodable INDEX code for any graph G is at least $\mathrm{minrk}_2(G)$. We relax the notion of linearly-decodable codes to "semi-linearly-decodable" codes. An INDEX code is k-

linearly-decodable if at least k of its decoding functions are linear. Note that n-linearly-decodable codes are simply linearly-decodable, while 0-linearly-decodable codes are unrestricted. We are able to prove that $\mathrm{minrk}_2(G)$ is the optimal length of k-linearly-decodable codes when $k \geq n-2$:

Theorem 6. *For any graph G, and for any $k \geq n-2$, the length of any k-linearly-decodable* INDEX *code for G is at least* $\mathrm{minrk}_2(G)$.

As our new linear INDEX code (proof of Theorem 5in Section III) is also linearly-decodable (and thus k-linearly-decodable, for any k), the bound in Theorem 6 is tight.

Our third contribution is a lower bound that holds for general INDEX codes including deterministic and randomized INDEX codes. This result is presented in Section V.

Theorem 7. *The length of any δ-error randomized* INDEX *code for G is at least* $\mathrm{MAIS}(G)\cdot(1-H_2(\delta))$, *where* $\mathrm{MAIS}(G)$ *is the size of the maximum acyclic induced subgraph of G and $H_2(\cdot)$ is the binary entropy function.*

This lower bound immediately gives a tight bound for directed acyclic graphs and undirected graphs G that satisfy $\omega(\overline{G}) = \mathrm{minrk}_2(G) = \chi(\overline{G})$. In particular, it holds for perfect graphs[1]. In Section VI, we are able to prove that minrank characterizes the minimum length of INDEX codes, even for non-perfect graphs, namely *odd holes* (undirected odd-length cycles of length at least 5) and *odd anti-holes* (complements of odd holes).

Theorem 8. *Let G be any graph, which is either a DAG, a perfect graph, an odd hole, or an odd anti-hole. Then, the length of any* INDEX *code for G is at least* $\mathrm{minrk}_2(G)$.

This theorem implies that our lower bound for general codes (Theorem 7) is not tight. For odd holes, $\mathrm{MAIS}(G)$ is the size of the largest independent set, i.e., $\omega(\overline{G})$, which we show to be strictly smaller than $\mathrm{minrk}_2(G)$.

The Strong Perfect Graph Theorem (conjectured by Berge [13] and proved by Chudnovsky *et al.* [14]) states that a graph is perfect if and only if it contains no (induced) odd hole or odd anti-hole. It follows that every undirected graph can be partitioned into induced subgraphs, each of which is either perfect, an odd hole, or an odd anti-hole. This motivated us to study the following direct sum-type problem: if a graph G can be partitioned into k induced sub-graphs $G_1, \ldots, G_k$, then is the length of the best INDEX code for G equal to the sum of the lengths of the best codes for $G_1, \ldots, G_k$? While we believe the answer to this general question to be negative, we were able to prove it for the case wherein $G_1, \ldots, G_k$ are disconnected components (i.e., there is no edge connecting G_i and G_j, for any $i \neq j$). A direct proof of this result seems to be elusive. In fact, an argument based on the techniques of Feder *et al.* [15] incurs a loss of an additive term that depends linearly on k. After lower bounding the length of a code by its *information cost* [16], [17], we are able to prove a tight direct sum theorem w.r.t. the information cost measure. We note that almost all our lower bounds hold not only for the length of INDEX codes but also for their information cost. This result is presented in Section V.

Techniques. We resort to a multitude of techniques from linear algebra, information theory, Fourier analysis, and combinatorics to prove the results presented in this paper.

The lower bounds for linearly-encodable and linearly-decodable codes are based on dimension arguments from linear algebra. To extend the lower bound for linearly-decodable codes to semi-linearly-decodable codes, we used an intriguing "balance property" of Boolean functions: if all linear Boolean functions are "balanced" on some set U (i.e., get the same number of 0's and 1's on the set), then all Boolean functions (whether linear or not) are balanced on U. To prove this property, we use Fourier analysis to represent arbitrary Boolean functions as linear combinations of linear functions. We then introduce the notion of "minimum dimension", which is dual to minrank, and explore its properties using the balance property. This in turn allows us to extend the lower bound for linearly-decodable codes to $(n-2)$-linearly-decodable codes.

The lower bound for general (randomized) codes and the direct sum theorem are proved via information theory arguments. We extend previous arguments from [17], [18] to obtain a direct sum theorem for the *information cost* of codes.

Finally, our lower bounds for odd holes and odd anti-holes are purely combinatorial. We employ a connection between vertex covers of a graph G and the structure of the confusion graph corresponding to INDEX codes for G. We note that dealing with odd holes, and with the pentagon in particular, turned out to be very challenging, because the standard technique of lower bounding the chromatic number of the corresponding confusion graph via its independence number does not work.

Related work. There are settings other than source coding in which INDEX codes have been addressed. Ambainis *et al.* [19] considered what they called "random access codes", which are identical to randomized INDEX codes without side information. Their main thrust was proving tight bounds on the length of the codes in the quantum setting, where inputs can be encoded by qubits rather than classical bits; their result applied to the classical setting is a special case of our Theorem 7 for the case when G is the empty graph.

The problem of INDEX coding with side information can also be cast as a *one-way communication complexity* problem of the *indexing* function [20] (from which the term INDEX codes was coined) with the additional twist of side information. Alice (the sender) is given an input x and sends a single message to Bob. Bob is given an index i and the side information $x[N(i)]$, and uses Alice's message to learn x_i. Another formulation of INDEX coding is in terms of *network coding* [21], [22]. As such, it represents a restricted case of a single source, a single encoder and a single channel, but with the important addition of a special flavor of side information. Parts of this information are known to different decoders, and the encoder is fully aware of this knowledge.

Subsequent work. Following the publication of the extended abstract of this work [23], Lubetzky and Stav [24] were able to make remarkable progress and prove that there could be an unbounded gap between $\mathrm{minrk}_2(G)$ and the length of an

[1] Recall that an undirected graph G is called *perfect*, if for every induced subgraph G' of G, $\omega(\overline{G'}) = \chi(\overline{G'})$. Perfect graphs include a wide class of graphs such as trees, bipartite graphs, interval graphs, chordal graphs, etc.

optimal INDEX code for G. They achieved this by constructing a new family of Ramsey graphs. It thus remains an open problem to find an exact characterization of the optimal length of INDEX codes for general codes and arbitrary graphs. It should nonetheless be noted that the commonly used codes are linear, for which the bounds presented in the current paper are tight.

Notation. Throughout the paper, we use the following notations. Let $[n]$ denote the set $\{1, 2, \ldots, n\}$. Let e_i denote the i-th standard basis vector. The dimension of this vector is understood from the context. For a subset $S \subseteq [n]$, we denote by $x[S]$ the projection of a vector $x \in \{0, 1\}^n$ on the coordinates in S.

II. Motivating application

Many important data dissemination systems employ a broadcast channel at the physical level. Prominent examples include satellite, terrestrial wireless systems, and systems employing coaxial cable. Broadcast channels are frequently used to disseminate high volumes of media-rich content, such as movies, episodes of TV series and video clips, as well as text and images such as the daily newspaper (which may soon include video clips).

The broadcast channel is extremely efficient for sending the same data to a large number of receivers, but its use for sending different data to different users is wasteful. For the case wherein users (may) need the same data but at different times (e.g., on-demand viewing of a "hot" movie), service providers can speculatively "push" data to clients; upon demand by a client, the already present data is presented immediately, as if it were sent on demand. XTV by News Data Systems[2] is an early example.

The above example demonstrates how abundant client storage capacity can be used in liew of true communication capability in order to increase perceived communication-related quality of service. With the rapid decline in the cost of non-volatile storage (disk drives and Flash memory), it is interesting to look for additional ways of exploiting its abundance in order to reduce demand on less abundant or more expensive resources. We next describe such an application, which has motivated the current work.

Despite the use of a broadcast channel, not all information is received by all clients. This may result from intermittent connectivity due to bad weather, a power outage at some receiver locations, intermittent reception by mobile receivers, or due to equipment being temporarily switched off. Finally, even data that is received by a given client may be discarded, be it for lack of space or by an "interest filter".

Following the broadcast of various content, each client thus typically has in its local storage some subset of the transmitted data. As for the remainder of this data, a given client may request some of it while not being interested in the rest. The question is then how to use a broadcast channel, which is very effective at sending the same data to all recipients, in order to efficiently send different *supplemental* data to the different clients. An important insight provided in [7] is that whenever each client stores a substantial fraction of the transmitted data while only requesting a single block, the probability that a pair of clients each have the block requested by the other is much higher than the probability that they both request the same block. This gave rise to the idea of using source coding for this purpose.

Motivated by the above, Birk and Kol posed in [7] the following coding problem. A server transmits a set of data blocks over a broadcast channel to a set of caching clients. Each client only stores a subset of the transmitted blocks. Each client needs a certain subset of the transmitted blocks, yet some of these blocks may be missing from its cache. The client can use a (slow) "upstream" channel to request blocks it has not cached and to advise the server of the blocks it already has in its cache. (By using large blocks, the amount of this metadata information can be made negligible.) The challenge posed was to design coding schemes that minimize the amount of supplemental information that must be broadcast in order to enable every client to derive all its requested blocks. It is important to note that the goal is not to provide all blocks to every client.

Blocks are usually compressed before being transmitted, and thus we can assume their bits to be independent. Therefore, a code for blocks of size B can be constructed by concatenating B instances of a code for single bit blocks: an instance encoding the first bit of all requested blocks, an instance encoding the second bit of all requested blocks, etc. Note that the metadata information for all bits in a requested/cached block is the same, and thus the client can still transmit this information once per block. The focus is thus on codes for single bit blocks. Finally, both [7] and the current paper assume that any given block is requested by at most one client (though others may have it, of course).

III. Linear codes

In this section we obtain a tight characterization of the length of linear INDEX codes for all side information graphs G.

Theorem 5 (restated) *For any side information graph G, there exists a linear* INDEX *code for G whose length equals* $\mathrm{minrk}_2(G)$. *This bound is optimal for all linear* INDEX *codes for G.*

Proof: Let A be the matrix that fits G whose 2-rank equals $\mathrm{minrk}_2(G) \triangleq k$. Assume without loss of generality that the span of the first k rows $A_1, \ldots, A_k$ equals the span of all the rows of A. The encoding function is simply the k bits $b_j \triangleq A_j \cdot x$ for $1 \leq j \leq k$.

Decoding proceeds as follows. Fix a receiver R_i for some $i \in [n]$ and let $A_i = \sum_{j=1}^{k} \lambda_j A_j$ for some choice of λ_j's. The receiver first computes $A_i \cdot x = \sum_{j=1}^{k} \lambda_j b_j$ using the k-bit encoding of x. Now, consider the vector $c_i = A_i - e_i$, where e_i is the i-th standard basis vector. Observe that the only non-zero entries in c_i correspond to coordinates that are among the neighbors of i in G. This means that the receiver can compute $c_i \cdot x$ using the side information. Receiver R_i can now recover x_i via $(A_i \cdot x) - (c_i \cdot x) = e_i \cdot x = x_i$.

[2] www.nds.com

For the lower bound, suppose $\mathcal{C}$ is an arbitrary linear INDEX code for G defined by the set $S = \{u_1, u_2, \ldots, u_k\}$, i.e., x is encoded by the taking its inner product with each vector in S.

Claim 9. *For every i, e_i belongs to the span of $S \cup \{e_j : j \in N(i)\}$.*

Before we prove the claim, we show how to finish the proof of the lower bound. For each $i \in [n]$, the claim shows that $e_i = \sum_{j=1}^{k} \lambda_j u_j + \sum_{j \in N(i)} \mu_j e_j$, for some choice of λ and μ. Rearranging, we have $\sum_{j=1}^{k} \lambda_j u_j = e_i - \sum_{j \in N(i)} \mu_j e_j \triangleq A_i$. It follows that A_i has value 0 in coordinates outside $N(i) \cup \{i\}$, A_i has value 1 in its i-th coordinate, and A_i belongs to the span of S. Therefore, the matrix A whose rows are given by $A_1, A_2, \ldots, A_n$ fits G and has rank at most k. We conclude that $k \geq \mathrm{rk}_2(A) \geq \mathrm{minrk}_2(G)$.

It remains to prove the claim. Fix an i and suppose to the contrary that e_i is *not* in the subspace W spanned by the vectors in $S \cup \{e_j : j \in N(i)\}$. Recall that the *dual* of W, denoted by $W^{\perp}$, consists of the set of vectors orthogonal to every vector in W, i.e., $W^{\perp} = \{v : v \cdot w = 0 \text{ for all } w \in W\}$. It is well-known that $W^{\perp\perp} = W$. Therefore, the assumption $e_i \notin W$ implies that there is a vector $x \in W^{\perp}$ such that $x \cdot e_i \overset{(*)}{\neq} 0$. On the other hand, since $x \in W^{\perp}$, we have that x is orthogonal to every vector in $S \cup \{e_j : j \in N(i)\}$. It follows that (i) the encoding for x equals 0^k, and (ii) the side information x_j available to receiver R_i equals 0 for all $j \in N(i)$. This violates the correctness of the encoding because the input 0^n also satisfies (i) and (ii), yet Equation (*) shows that it differs from x in coordinate i. ■

IV. Semi-linearly-decodable codes

In this section, we show that $\mathrm{minrk}_2(G)$ is a lower bound on the minimum length of *semi-linearly-decodable* INDEX codes for arbitrary graphs G.

Let $\mathcal{C}$ be an INDEX code for G. Let $D_1, \ldots, D_n$ be the n decoding functions of $\mathcal{C}$. Fix a codeword $c \in \mathcal{C}$, and for each index $i \in [n]$, we denote by D_i^c the function induced by fixing c as input to D_i: $D_i^c(x[N(i)]) = D_i(c, x[N(i)])$. Although D_i^c is applied only to the side information bits $x[N(i)]$, it will be convenient for us to view it as acting on the whole input x with the restriction that it depends only[3] on the set of coordinates $N(i)$. Thus, from now on, $D_i^c : \{0,1\}^n \to \{0,1\}$.

An INDEX code $\mathcal{C}$ is said to be *k-linearly-decodable*, if for every codeword $c \in \mathcal{C}$, at least k of the decoding functions $D_1^c, \ldots, D_n^c$ are linear. Note that the smaller k is, the less restricted is the class of k-linearly-decodable codes. When $k = n$, these codes are simply called *linearly-decodable*, while 0-linearly-decodable are unrestricted codes. Our upper bound (Theorem 5) is a linearly-decodable INDEX code (and thus also k-linearly-decodable, for any k).

Our goal is to obtain lower bounds on the length of k-linearly-decodable codes for a value of k as small as possible.

Theorem 6 (restated) *For any graph G, and for any $k \geq n-2$, the length of any k-linearly-decodable* INDEX *code for G is at least $\mathrm{minrk}_2(G)$.*

A. Kernel size

To prove the lower bound, we introduce the notion of *kernel*. The kernel of a Boolean function $f : \{0,1\}^n \to \{0,1\}$ is the set of inputs it maps to 0: $\ker(f) = \{x \mid f(x) = 0\}$. By extension, the kernel of a family of Boolean functions $\mathcal{F} = \langle f_i \mid i \in T \rangle$ (T is some index set) is the set of inputs that are mapped to 0 by all of the functions in the family: $\ker(\mathcal{F}) = \{x \mid f_i(x) = 0 \ \forall i\}$. We next show a connection between the length of INDEX codes and the size of the kernel of a suitably chosen family of functions.

Note that $D_i^c(x) = x_i$ for every x whose encoding $E(x)$ equals c. This can be also written as $D_i^c(x) + e_i \cdot x = 0$. If we view the vector e_i as a linear function operating over $\{0,1\}^n$, then we can say that x belongs to the kernel of the function $D_i^c + e_i$, i.e., $(D_i^c + e_i)(x) = 0$. As this holds for every i, we conclude the following:

Proposition 10. *For every codeword c, $\{x \mid E(x) = c\} \subseteq \ker(D_1^c + e_1, \ldots, D_n^c + e_n) = \{x \mid D_i^c(x) = x_i \ \forall i\}$.*

We obtain as an immediate corollary the following lower bound on the length of INDEX codes in terms of kernel size:

Proposition 11. *If $|\ker(D_1^c + e_1, \ldots, D_n^c + e_n)| \leq M$ for every codeword $c \in \mathcal{C}$, then $\mathrm{len}(\mathcal{C}) \geq \lceil n - \log M \rceil$.*

Proof: Consider any codeword $c \in \mathcal{C}$. Let $E^{-1}(c) = \{x \mid E(x) = c\}$ be the set of inputs whose corresponding codeword is c. By Proposition 10, $|E^{-1}(c)| \leq M$. Hence, the number of distinct codewords in $\mathcal{C}$ is at least $2^n/M$, and thus its length must be at least $\lceil n - \log M \rceil$. ■

Thus, to prove Theorem 6, it suffices to prove the following:

Theorem 12. *Let c be a codeword in a k-linearly-decodable code $\mathcal{C}$ with side information graph G, where $k \geq n-2$. Then, $|\ker(D_1^c + e_1, \ldots, D_n^c + e_n)| \leq 2^{n - \mathrm{minrk}_2(G)}$.*

We will in fact prove a more general version of Theorem 12. To state this more general form, we first need to extend the notion of *fitting*.

Fix a graph G on n nodes. We say that a function $f : \{0,1\}^n \to \{0,1\}$ *fits* an index $i \in [n]$, if $f = g + e_i$ for some function g that depends only on $N(i)$ (g is not necessarily linear). Note that $f(x) = g(x) + x_i$. Extending the definition, we say that a family of (not necessarily distinct) functions $\langle f_j : \{0,1\}^n \to \{0,1\} \mid j \in T \rangle$ *fits* a subset T of the indices in $[n]$, if f_j fits j for every $j \in T$.

Every linear function $f : \{0,1\}^n \to \{0,1\}$ corresponds to a vector v so that $f(x) = v \cdot x$. Therefore, f fits index i if and only if v can be written as $v = d + e_i$, where d is a vector whose value in every coordinate $j \notin N(i)$ equals 0. A matrix A fits $[n]$ (or, G), if the i-th row of A, for every i, fits index i. As the value of this row must be 1 in the i-th coordinate and 0 in every coordinate $j \notin N(i) \cup \{i\}$, this definition is consistent with our earlier definition for a matrix fitting a graph.

Fix an INDEX code $\mathcal{C}$ for G and a codeword $c \in \mathcal{C}$. Let $D_1^c, \ldots, D_n^c$ be the n decoding functions associated with c.

[3] A function $f : \{0,1\}^n \to \{0,1\}$ is said to *depend only* on a set of coordinates $S \subseteq [n]$, if for every two inputs x, y with $x[S] = y[S]$, $f(x) = f(y)$.

Note that each function $D_i^c + e_i$ fits index i, for all i, and thus the family $\langle D_1^c + e_1, \dots, D_n^c + e_n \rangle$ fits $[n]$.

We say that a family of functions $\langle f_j \mid j \in T \rangle$ is k*-linear*, if at least k of the functions in the family are linear. Note that if $\mathcal{C}$ is k-linearly-decodable, then the family $\langle D_1^c + e_1, \dots, D_n^c + e_n \rangle$ is k-linear.

The stronger version of Theorem 12 we will prove is the following:

Theorem 13. *Let G be a graph on n nodes and let $k \geq n - 2$. Then, for any k-linear family $\mathcal{F} = \langle f_j \mid j \in [n] \rangle$ of Boolean functions that fits $[n]$, $|\ker(\mathcal{F})| \leq 2^{n - \mathrm{minrk}_2(G)}$.*

Theorem 12 follows by setting $f_j = D_j^c + e_j$ for every j. The rest of this section is devoted to the proof of Theorem 13.

B. *Maximum dimension*

In this section we explore a new notion—the *maximum dimension*—which is dual to the minrank and plays a key role in the proof of Theorem 13.

To motivate the proof, consider the following simple argument for the case $k = n$ (i.e., all the functions f_j are linear). Since f_j is linear and fits index j, it is associated with a vector v_j so that $f_j(x) = v_j \cdot x$. Let A be the $n \times n$ Boolean matrix whose rows are $v_1, \dots, v_n$. Since f_j fits index j, it follows that A fits G, so $\mathrm{rk}_2(A) \geq \mathrm{minrk}_2(G)$. Next, observe that $\ker(\mathcal{F})$ is exactly the kernel of the matrix A. By standard linear algebra, the dimension of this kernel is $n - \mathrm{rk}_2(A) \leq n - \mathrm{minrk}_2(G)$, and therefore the size of the kernel is at most $2^{n - \mathrm{minrk}_2(G)}$.

To deal with the case $k < n$, we would like to generalize the above argument. When some of the functions in $\mathcal{F}$ are not linear, $\ker(\mathcal{F})$ is no longer a linear space and thus does not have a properly defined dimension. In order to address this difficulty, we introduce the new notion of *maximum dimension*.

Let S be any subset of $[n]$ and let $\mathcal{H}_S = \langle h_j \mid j \in S \rangle$ be any family of *linear* functions that fits S. Let $T \subseteq [n] \setminus S$. For any family $\mathcal{H}_T = \langle h_j \mid j \in T \rangle$ of (not necessarily linear) functions that fits T, we denote by $\mathcal{H}_{S \cup T}$ the union of the two families: $\langle h_j \mid j \in S \cup T \rangle$. When $\mathcal{H}_T$ is also a family of linear functions, $\ker(\mathcal{H}_{S \cup T})$ is a linear space and thus has a dimension. We define the *maximum dimension of T relative to $\mathcal{H}_S$*, denoted $\mathrm{maxdim}_2(T|\mathcal{H}_S)$, to be the maximum value of $\dim(\ker(\mathcal{H}_{S \cup T}))$, where the maximum is taken over all families $\mathcal{H}_T$ of linear functions that fit T. Note that when $S = \emptyset$, $T = [n]$, $\mathrm{maxdim}_2(T|\emptyset) = n - \mathrm{minrk}_2(G)$, and thus the maximum dimension can be viewed as dual to the minrank. The following are basic facts about the maximum dimension that will be used later in our analysis:

Proposition 14. *Fix any set $S \subseteq [n]$, any family $\mathcal{H}_S$ of linear functions that fits S, and any set $T \subseteq [n] \setminus S$. For simplicity, we shorthand $\mathrm{maxdim}_2(T)$ for $\mathrm{maxdim}_2(T|\mathcal{H}_S)$. The following are properties of $\mathrm{maxdim}_2(T)$:*

1) $\mathrm{maxdim}_2(\emptyset) = \dim(\ker(\mathcal{H}_S))$.
2) *For any $i \in T$, $\mathrm{maxdim}_2(T) \leq \mathrm{maxdim}_2(T \setminus \{i\}) \leq \mathrm{maxdim}_2(T) + 1$.*
3) *More generally, $\mathrm{maxdim}_2(T) \leq \mathrm{maxdim}_2(T') \leq \mathrm{maxdim}_2(T) + |T| - |T'|$ for any $T' \subseteq T$.*
4) $\dim(\ker(\mathcal{H}_S)) - |T| \leq \mathrm{maxdim}_2(T) \leq \dim(\ker(\mathcal{H}_S))$.
5) *If $\mathrm{maxdim}_2(T) = \dim(\ker(\mathcal{H}_S)) - |T|$, then $\mathrm{maxdim}_2(T') = \dim(\ker(\mathcal{H}_S)) - |T'|$ for every $T' \subseteq T$.*
6) *Suppose $\mathrm{maxdim}_2(\{j\}) = \dim(\ker(\mathcal{H}_S))$ for every $j \in T$. Then $\mathrm{maxdim}_2(T) = \dim(\ker(\mathcal{H}_S))$ as well.*
7) *Let $T = [n] \setminus S$. Then, $\mathrm{maxdim}_2(T) \leq n - \mathrm{minrk}_2(G)$.*

Proof: Part 1 follows simply by definition. Part 2 follows from the standard linear algebra fact that adding a single constraint to any subspace can only decrease its dimension, but by at most 1; an inductive argument yields Part 3. Setting $T' = \emptyset$ in Part 3 and then using Part 1 yields Part 4.

For Part 5, note that Part 4 implies that $\mathrm{maxdim}_2(T') \geq \dim(\ker(\mathcal{H}_S)) - |T'|$. By Part 3, $\mathrm{maxdim}_2(T') \leq \mathrm{maxdim}_2(T) + |T| - |T'| = \dim(\ker(\mathcal{H}_S)) - |T'|$, using the premise of Part 5. Therefore, $\mathrm{maxdim}_2(T') = \dim(\ker(\mathcal{H}_S)) - |T'|$ as well.

For Part 6, the premise says that there exist linear functions h_j for all $j \in T$ such that $h_j(x) = 0$ for all $x \in \ker(\mathcal{H}_S)$. Define the family $\mathcal{H}_T = \langle h_j : j \in T \rangle$. It can be seen that $\ker(\mathcal{H}_{S \cup T}) = \ker(\mathcal{H}_S)$ and thus $\dim(\ker(\mathcal{H}_{S \cup T})) = \dim(\ker(\mathcal{H}_S))$, which is the maximum value that $\mathrm{maxdim}_2(T)$ can attain by Part 4.

Finally, for Part 7, let $\mathcal{H}_T$ be the family of linear functions such that $\dim(\ker(\mathcal{H}_{S \cup T})) = \mathrm{maxdim}_2(T)$. Recall that $\mathcal{H}_{S \cup T}$ fits $S \cup T = [n]$, so let A be the matrix whose rows consist of the vectors that correspond to the functions in $\mathcal{H}_{S \cup T}$. It follows that A fits G. Since its kernel equals $\ker(\mathcal{H}_{S \cup T})$, we conclude:

$$\dim(\ker(\mathcal{H}_{S \cup T})) = n - \mathrm{rk}_2(A) \leq n - \mathrm{minrk}_2(G).$$

■

The following lemma is the main technical result that will be used to prove Theorem 13.

Lemma 15. *Let G be a graph on n nodes. Then, for any $S \subseteq [n]$, any family $\mathcal{H}_S$ of linear functions that fits S, any $T \subseteq [n] \setminus S$ with $|T| \leq 2$, and any family $\mathcal{H}_T$ of (not necessarily linear) functions that fits T, $|\ker(\mathcal{H}_{S \cup T})| \leq 2^{\mathrm{maxdim}_2(T|\mathcal{H}_S)}$.*

To derive Theorem 13, we choose S to be the set of indices of the k linear functions in $\mathcal{F}$, $\mathcal{H}_S$ to be these linear functions, $T = [n] \setminus S$, and $\mathcal{H}_T$ to be the rest of the functions in $\mathcal{F}$. Note that $\mathcal{F} = \mathcal{H}_{S \cup T}$. By Proposition 14, Part 7, we have $\mathrm{maxdim}_2(T|\mathcal{H}_S) \leq n - \mathrm{minrk}_2(G)$, which immediately yields Theorem 13.

Note that the restriction we have on k ($k \geq n - 2$) in Theorems 6, 12, and 13 derives from the restriction we have in Lemma 15 on $|T|$ ($|T| \leq 2$). It remains an open problem to find the largest value of $|T|$ (and thus the smallest value of k) for which the bound holds.

We first prove a stronger version of Lemma 15 for the special case when $\mathrm{maxdim}_2(T|\mathcal{H}_S)$ has the smallest possible value $\dim(\ker(\mathcal{H}_S)) - |T|$ (Proposition 14, Part 4), in which case the bound given by Lemma 15 is achieved for every T (even $|T| > 2$).

Lemma 16. *Let G, S, $\mathcal{H}_S$, T, and $\mathcal{H}_T$ be as defined above (except that $|T|$ need not be at most 2). If $\mathrm{maxdim}_2(T|\mathcal{H}_S) =$*

$\dim(\ker(\mathcal{H}_S)) - |T|$, then $|\ker(\mathcal{H}_{S\cup T})| = 2^{\dim(\ker(\mathcal{H}_S))-|T|}$.

Proof: As we will see below, proving the lemma for the case $\mathcal{H}_T$ is a family of linear functions is easy (follows from standard dimension arguments). To extend the proof to hold for unrestricted functions, we will use a "Balance Lemma", which is proved in the next section via Fourier analysis.

The lemma will be proved by gradually moving from a family $\mathcal{H}_T$ of linear functions to a family $\mathcal{H}_T$ of unrestricted functions. Formally, we will show the following:

Claim 17. *Let G, S, $\mathcal{H}_S$, T, and $\mathcal{H}_T$ be as defined in Lemma 16. Let $\ell \leq |T|$. If $\mathcal{H}_T$ is ℓ-linear and $\mathrm{maxdim}_2(T|\mathcal{H}_S) = \dim(\ker(\mathcal{H}_S)) - |T|$, then $|\ker(\mathcal{H}_{S\cup T})| = 2^{\dim(\ker(\mathcal{H}_S))-|T|}$.*

Applying Claim 17 with $\ell = 0$ will yield Lemma 16. We prove the claim by a double induction: an outer induction on $|T|$ and an inner induction on $|T| - \ell$.

The base case of the outer induction, $|T| = 0$, follows from standard linear algebra, because $|\ker(\mathcal{H}_S)| = 2^{\dim(\ker(\mathcal{H}_S))}$. For the base case of the inner induction, $\ell = |T|$, note that $\mathcal{H}_T$ is a linear family of functions. Therefore, $\ker(\mathcal{H}_{S\cup T})$ is a linear space and $\dim(\ker(\mathcal{H}_{S\cup T})) \leq \mathrm{maxdim}_2(T|\mathcal{H}_S) = \dim(\ker(\mathcal{H}_S)) - |T|$. On the other hand, $\ker(\mathcal{H}_{S\cup T}) \geq \dim(\mathcal{H}_S) - |T|$, because each constraint added to a linear sub-space can reduce its dimension by at most 1. Hence, $\ker(\mathcal{H}_{S\cup T}) = \dim(\mathcal{H}_S) - |T|$. Now, define $S' = S \cup T$ and $T' = \emptyset$. As all the functions in $\mathcal{H}_{S'}$ are linear and as $\mathrm{maxdim}_2(T'|\mathcal{H}_{S'}) = \dim(\ker(\mathcal{H}_{S'}))$ (Proposition 14, Part 1), then we can apply the base case of the outer induction to conclude that

$$\begin{aligned} |\ker(\mathcal{H}_{S\cup T})| &= |\ker(\mathcal{H}_{S'})| = 2^{\dim(\ker(\mathcal{H}_{S'}))} \\ &= 2^{\dim(\ker(\mathcal{H}_S))-|T|}. \end{aligned}$$

Let $1 \leq t \leq n$ and let $0 \leq \ell \leq t - 1$. For the induction step, assume that the claim holds for the following cases: (1) every T with $|T| < t$ and every family $\mathcal{H}_T$ that fits T (no linearity restrictions on $\mathcal{H}_T$); (2) every T with $|T| = t$ and every ℓ'-linear family $\mathcal{H}_T$ that fits T, where $\ell' > \ell$. We will show that the claim holds also for the case $|T| = t$ and $\mathcal{H}_T$ is ℓ-linear.

Let $\mathcal{H}_T$ be any ℓ-linear family of functions that fits T. At least ℓ of the functions in $\mathcal{H}_T$ are linear. If $\mathcal{H}_T$ has $\ell + 1$ linear functions or more, then it is in fact $(\ell + 1)$-linear, and therefore the statement of the claim follows in this case from the induction hypothesis. So suppose exactly ℓ of the functions in $\mathcal{H}_T$ are linear. As $\ell < |T|$, $\mathcal{H}_T$ has at least one non-linear function. Let h_i, where $i \in T$, be one such function.

Let $T_{-i} = T \setminus \{i\}$ and let $\mathcal{H}_{T_{-i}}$ be the family of functions obtained by removing h_i from $\mathcal{H}_T$. We will prove that $|\ker(\mathcal{H}_{S\cup T})| = 2^{\ker(\mathcal{H}_S)-|T|}$ in two steps. First, we will show that $|\ker(\mathcal{H}_{S\cup T_{-i}})| = 2^{\ker(\mathcal{H}_S)-|T|+1}$. Then, we will prove that h_i is *balanced* on the set $\ker(\mathcal{H}_{S\cup T_{-i}})$:

Definition 18 (Balanced function). A Boolean function $f : \{0,1\}^n \to \{0,1\}$ is said to be *balanced* on a subset U of its domain, if it is 0 on half of the inputs in U and it is 1 on the other half. That is, $|\ker(f) \cap U| = |U|/2$.

Having proved that $|\ker(\mathcal{H}_{S\cup T_{-i}})| = 2^{\ker(\mathcal{H}_S)-|T|+1}$ and that h_i is balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$, we will obtain the desired equality:

$$\begin{aligned} |\ker(\mathcal{H}_{S\cup T})| &= |\ker(h_i) \cap \ker(\mathcal{H}_{S\cup T_{-i}})| = \\ &= \frac{1}{2}|\ker(\mathcal{H}_{S\cup T_{-i}})| = 2^{\ker(\mathcal{H}_S)-|T|}. \end{aligned}$$

We start by showing that $|\ker(\mathcal{H}_{S\cup T_{-i}})| = 2^{\ker(\mathcal{H}_S)-|T|+1}$. Using Proposition 14, Part 5, since $\mathrm{maxdim}_2(T|\mathcal{H}_S) = \dim(\ker(\mathcal{H}_S)) - |T|$, then $\mathrm{maxdim}_2(T_{-i}|\mathcal{H}_S) = \dim(\ker(\mathcal{H}_S) - |T| + 1$. As $|T_{-i}| < t$, we can apply the induction hypothesis and obtain what we wanted:

$$|\ker(\mathcal{H}_{S\cup T_{-i}})| = 2^{\ker(\mathcal{H}_S)-|T|+1}. \tag{1}$$

Showing that h_i is balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$ is harder. To this end, we first prove that every *linear* function that fits index i must be balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$. We then prove a Balance Lemma, which shows that every function that fits index i, h_i included, must be balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$.

Let us start by proving that every linear function that fits index i is balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$. Let g_i be any such linear function and let $\mathcal{H}'_T = \mathcal{H}_{T_{-i}} \cup \{g_i\}$. Note that $\mathcal{H}'_T$ fits T and that it is $(\ell+1)$-linear. Let $\mathcal{H}'_S = \mathcal{H}_S$. Applying the induction hypothesis we obtain:

$$|\ker(\mathcal{H}'_{S\cup T})| = 2^{\ker(\mathcal{H}'_S)-|T|} = 2^{\ker(\mathcal{H}_S)-|T|}. \tag{2}$$

We can rewrite $\ker(\mathcal{H}'_{S\cup T})$ as follows:

$$\ker(\mathcal{H}'_{S\cup T}) = \ker(g_i) \cap \ker(\mathcal{H}'_{S\cup T_{-i}}) = \ker(g_i) \cap \ker(\mathcal{H}_{S\cup T_{-i}}). \tag{3}$$

Combining Equations 1, 2, and 3, we have:

$$|\ker(g_i) \cap \ker(\mathcal{H}_{S\cup T_{-i}})| = \frac{1}{2}|\ker(\mathcal{H}_{S\cup T_{-i}})|. \tag{4}$$

Equation 4 implies that the function g_i is balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$. As all we used is the linearity of g_i and the fact it fits index i, we conclude that every linear function that fits index i is balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$. The following Balance Lemma, which is proved in the next section, shows that every function that fits index i, whether linear or not, must be balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$.

Lemma 19 (Balance Lemma). *Let G be a graph on n nodes, let $i \in [n]$, and let $U \subseteq \{0,1\}^n$. If every linear function that fits index i is balanced on U, then every function that fits index i (whether linear or not) is balanced on U.*

We conclude that in particular h_i is balanced on $\ker(\mathcal{H}_{S\cup T_{-i}})$, which is what we wanted. Claim 17 and Lemma 16 follow. ■

We can now prove Lemma 15:

Proof of Lemma 15: For brevity of notation, throughout this proof we shorthand $\mathrm{maxdim}_2(T)$ for $\mathrm{maxdim}_2(T|\mathcal{H}_S)$.

We prove the lemma by induction on the size of T. The case $|T| = 0$, meaning $T = \emptyset$, follows simply from the fact that $\mathrm{maxdim}_2(\emptyset) = \dim(\mathcal{H}_S)$ (Proposition 14, Part 1):

$$|\ker(\mathcal{H}_{S\cup T})| = |\ker(\mathcal{H}_S)| = 2^{\dim(\mathcal{H}_S)} = 2^{\mathrm{maxdim}_2(T)}.$$

Let $t \in \{1, 2\}$. Assume that the statement of the lemma holds for all T such that $|T| < t$. We will prove it for $|T| = t$.

For $i \in T$, let $T_{-i} = T \setminus \{i\}$. By Proposition 14, Part 2, for every $i \in T$, $\mathrm{maxdim}_2(T_{-i}) \in \{\mathrm{maxdim}_2(T), \mathrm{maxdim}_2(T) + 1\}$. We split our analysis into two cases.

Case 1: For some $i \in T$, $\mathrm{maxdim}_2(T_{-i}) = \mathrm{maxdim}_2(T)$. In this case

$$|\ker(\mathcal{H}_{S\cup T})| \le |\ker(\mathcal{H}_{S\cup T_{-i}})| \le 2^{\mathrm{maxdim}_2(T_{-i})} = 2^{\mathrm{maxdim}_2(T)},$$

where the second inequality follows from the induction hypothesis and the last equality follows from our assumption in Case 1.

Case 2: For all $i \in T$, $\mathrm{maxdim}_2(T_{-i}) = \mathrm{maxdim}_2(T) + 1$. This is the case we know how to handle only for $|T| = 1, 2$. Suppose, first, that $|T| = 1$. Then, by the assumption of this case, $\mathrm{maxdim}_2(\emptyset) = \mathrm{maxdim}_2(T) + 1$. Since $\mathrm{maxdim}_2(\emptyset) = \dim(\ker(\mathcal{H}_S))$ (Proposition 14, Part 1), we rearrange and obtain $\mathrm{maxdim}_2(T) = \dim(\ker(\mathcal{H}_S)) - 1$. Hence, the statement follows in this case from Lemma 16.

Consider now the case $|T| = 2$ and let $T = \{i, j\}$. By the premise of Case 2, $\mathrm{maxdim}_2(\{i\}) = \mathrm{maxdim}_2(\{j\}) = \mathrm{maxdim}_2(\{i, j\}) + 1$. By Proposition 14, Part 2, either both $\mathrm{maxdim}_2(\{i\})$ and $\mathrm{maxdim}_2(\{j\})$ equal $\mathrm{maxdim}_2(\emptyset) = \dim(\ker(\mathcal{H}_S))$ or both are 1 less than $\dim(\ker(\mathcal{H}_S))$. The first case is impossible because by Proposition 14, Part 6, $\mathrm{maxdim}_2(\{i, j\}) = \dim(\mathcal{H}_S)$ as well, violating the premise of Case 2. Therefore, $\mathrm{maxdim}_2(\{i\}) = \mathrm{maxdim}_2(\{j\}) = \dim(\mathcal{H}_S) - 1$ implying that $\mathrm{maxdim}_2(\{i, j\}) = \dim(\mathcal{H}_S) - 2$. Hence, the statement follows in this case once again from Lemma 16. ■

C. Proof of the Balance Lemma

We next prove the Balance Lemma used in the proof of the lower bound for semi-linearly-decodable codes:

Lemma 19 (restated) *Let G be a graph on n nodes, let $i \in [n]$, and let $U \subseteq \{0, 1\}^n$. If every linear function that fits index i is balanced on U, then every function that fits index i (whether linear or not) is balanced on U.*

The proof of the lemma relies on a simple principle: under the mapping $0 \mapsto 1$ and $1 \mapsto -1$, a Boolean function f is balanced on the set U if and only if $\sum_{x\in U} f(x) = 0$. The linear Boolean functions in the ± 1 world are exactly the characters of the group $\mathbb{Z}_2^n$ and thus the lemma tells us that each of these characters sums to 0 on U. Fourier transform allows us to write any Boolean function f as a linear combination of characters. Therefore, if all characters sum to 0 on U, then also f must sum to 0 on U, and thus f is balanced.

To prove the lemma, we need to prepare some machinery from Fourier analysis of Boolean functions. Consider the group $\mathbb{Z}_2^n$, whose elements are the vectors $\{0, 1\}^n$. By mapping the standard 0 to 1, the standard 1 to -1, and the XOR operation to multiplication, we view the elements of the group as vectors in $\{-1, 1\}^n$, where coordinate-wise multiplication is the group operation. A complex function $f : \mathbb{Z}_2^n \to \mathbb{C}$ over this group can be viewed as a vector in $\mathbb{C}^{2^n}$. The inner product between two functions $f, g \in \mathbb{C}^{2^n}$ is defined as $\langle f, g\rangle = \frac{1}{2^n}\sum_{x\in\mathbb{Z}_2^n} f(x)g(x)$.

$\mathbb{Z}_2^n$ has 2^n *characters*. Each subset $S \subseteq [n]$ is associated with the character χ_S defined as: $\chi_S(x) = \prod_{i\in S} x_i$. The characters form an orthonormal basis of $\mathbb{C}^{2^n}$. The expansion of a function $f \in \mathbb{C}^{2^n}$ in this basis is its *Fourier Transform*. The coefficient of χ_S in this expansion is $\hat{f}(S) = \langle f, \chi_S\rangle$. Thus,

$$f = \sum_{S\subseteq[n]} \hat{f}(S)\chi_S.$$

A *Boolean function* is a function $f : \{-1, 1\}^n \to \{-1, 1\}$ (recall the mapping $0 \mapsto 1$ and $1 \mapsto -1$). The *kernel* of a Boolean function f is the set of inputs that is maps to 1: $\ker(f) = \{x | f(x) = 1\}$. It is easy to verify that the characters of $\mathbb{Z}_2^n$ are exactly the set of all Boolean linear functions on $\mathbb{Z}_2^n$.

To prove Lemma 19, we show two simple properties of Boolean functions.

Proposition 20. *Let $f : \{-1, 1\}^n \to \{-1, 1\}$ be a Boolean function that depends only on a set $S \subseteq [n]$. Then, the Fourier transform of f has non-zero coefficients only for characters χ_T with $T \subseteq S$.*

Proof: Let T be any subset of the coordinates that is not contained in S. We show that $\hat{f}(T) = 0$.

Since $T \not\subseteq S$, there exists a coordinate $i \in T \setminus S$. For each vector $x \in \mathbb{Z}_2^n$, let $x^{(i)}$ denote the vector obtained from x by flipping its i-th bit (from 1 to -1 or vice versa). Let Z^1 be the set of vectors in $\mathbb{Z}_2^n$ that have 1 at the i-th coordinate, and let Z^{-1} be the set of vectors in $\mathbb{Z}_2^n$ that have -1 at the i-th coordinate. The mapping $x \mapsto x^{(i)}$ induces a perfect matching of vectors in Z^1 with vectors in Z^{-1}.

Note that for a pair $(x, x^{(i)})$, $f(x) = f(x^{(i)})$, because the two inputs differ only outside the set S. However, $\chi_T(x) \neq \chi_T(x^{(i)})$ because x and $x^{(i)}$ differ only at the i-th coordinate and $i \in T$.

Consider now the coefficient $\hat{f}(T)$: $\hat{f}(T) = \frac{1}{2^n}\sum_{x\in\mathbb{Z}_2^n} f(x)\chi_T(x)$. We reorder the terms in the sum according to the above matching: $\hat{f}(T) = \frac{1}{2^n}\sum_{x\in Z^1}(f(x)\chi_T(x) + f(x^{(i)})\chi_T(x^{(i)})$. Since $f(x) = f(x^{(i)})$ and since $\chi_T(x) \neq \chi_T(x^{(i)})$, each of the terms in the above sum is 0. Therefore, $\hat{f}(T) = 0$, as desired. ■

Next, we characterize the set of Boolean linear functions that depend only on a set S:

Proposition 21. *The set of Boolean linear functions that depend only on S is exactly the set of characters $\{\chi_T\}_{T\subseteq S}$.*

Proof: Suppose $T \subseteq S$. We show that χ_T depends only on S. Let $x, x' \in \{-1, 1\}^n$ be two inputs s.t. $x[S] = x'[S]$. Since $T \subseteq S$, it follows that also $x[T] = x'[T]$. Therefore, $\prod_{i\in T} x_i = \prod_{i\in T} x'_i$, implying $\chi_T(x) = \chi_T(x')$.

For the other direction, suppose $T \not\subseteq S$. Let $i \in T \setminus S$. Let $\mathbf{1}$ be the all-one input (corresponding to the all-zero input in the 0-1 world) and let e_i be the standard unit vector (e_i is 1 in every coordinate, except for the i-th coordinate in which it is -1). Since $i \notin S$, $\mathbf{1}[S] = e_i[S]$. Clearly, $\chi_T(\mathbf{1}) = 1$. Since there is a single coordinate in T in which e_i is -1, then $\chi_T(e_i) = -1$. Thus, $\chi_T(\mathbf{1}) \neq \chi_T(e_i)$, implying χ_T does not depend only on S. ∎

We can now prove Lemma 19:

Proof of Lemma 19: A Boolean function $f : \{-1,1\}^n \to \{-1,1\}$ is balanced on U if and only if the number of inputs that it maps to 1 equals the number of inputs that it maps to -1. This in turn happens if and only if $\sum_{x\in U} f(x) = 0$.

By Proposition 21, the set of linear Boolean functions that depend only on $N(i)$ is the family of characters $\{\chi_T\}_{T\subseteq N(i)}$. Therefore, the set of linear functions that fit index i are of the form $\{\chi_T \cdot e_i\}_{T\subseteq N(i)}$. (Since we moved to the ±1 world, summation is mapped to multiplication, and the standard unit vector e_i is the all-one vector, except for the i-th coordinate which is -1.) The premise given in Lemma 19 implies that all these functions are balanced on U. That is, for every $T \subseteq N(i)$, $\sum_{x\in U} \chi_T(x)\cdot x_i = 0$.

Let f be any function (not necessarily linear) that fits index i. We can write $f = g \cdot e_i$, where g is a function that depends only on $N(i)$. By Proposition 20, g is a linear combination of the characters $\{\chi_T\}_{T\subseteq N(i)}$. Therefore,

$$\begin{aligned}
\sum_{x\in U} f(x) &= \sum_{x\in U} g(x)\cdot x_i = \sum_{x\in U}\left(\sum_{T\subseteq S}\hat{g}(T)\chi_T(x)\right)\cdot x_i \\
&= \sum_{x\in U}\sum_{T\subseteq S}\hat{g}(T)(\chi_T(x)\cdot x_i) \\
&= \sum_{T\subseteq S}\hat{g}(T)\sum_{x\in U}\chi_T(x)\cdot x_i \\
&= \sum_{T\subseteq S}\hat{g}(T)\cdot 0 = 0.
\end{aligned}$$

Therefore, also f is balanced on U. ∎

V. General codes

In this section, we prove lower bounds for the class of general randomized INDEX codes. Let us first formally define these codes.

Definition 22 (Randomized INDEX codes). Let $0 \le \delta < 1$. A δ-error randomized INDEX code $\mathcal{C}$ for $\{0,1\}^n$ with side information graph G on n nodes is a set of codewords in $\{0,1\}^\ell$ together with:

1) A public random string R for both encoding and decoding.
2) A private random string R_E for encoding.
3) n private random strings $R_{D_1}, \ldots, R_{D_n}$ for decoding.
4) An encoding function E that given a source input x in $\{0,1\}^n$ maps the triple (x, R, R_E) into a codeword.
5) A set of decoding functions $D_1, D_2, \ldots D_n$. For each i, D_i maps the quadruple $(E(x, R, R_E), x[N(i)], R, R_{D_i})$ into a bit, satisfying the following:

$$\Pr(D_i(E(x,R,R_E), x[N(i)], R, R_{D_i}) \neq x_i) \le \delta.$$

The probability is over the three random strings R, R_E, and R_{D_i}.

The $n+2$ random strings $R, R_E, R_{D_1}, \ldots, R_{D_n}$ have finite domains and are mutually independent of each other. Usually these are uniformly distributed strings of some fixed length.

The distributions of the private random strings are known in advance to all parties (the sender and the receivers), yet the specific instances chosen are known only to the respective parties. Therefore, the encoding function E may depend on the distributions of $R_{D_1}, \ldots, R_{D_n}$, but not on the specific instances chosen. Similarly, D_i may depend on the distribution of R_E, but not on the specific instance. As usual, the graph G is known in advance to the sender and the receivers and thus the encoding and decoding functions can depend on G. The *length* of $\mathcal{C}$, denoted by $\mathrm{len}(\mathcal{C})$, is defined to be ℓ.

The main technical statement of this section is a direct-sum result for the *information cost* of a randomized INDEX code. A corollary of this result will be the lower bound on the length of randomized INDEX codes. We start with a brief overview of the information theory notions and facts used in this section (See [25] for a more extensive background).

A. Information theory background

In the following $X \sim \mu_X$, $Y \sim \mu_Y$, $Z \sim \mu_Z$ are random variables on domains $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$, respectively.

Entropy and mutual information The *entropy* of X (or, equivalently, of μ_X) is $H(X) = \sum_{x\in\mathcal{X}} \mu_X(x) \log \frac{1}{\mu_X(x)}$. The *binary entropy function* $H_2(p)$ is the entropy of a Bernoulli random variable with probability of success p.

The *joint entropy* of X and Y is the entropy of the joint distribution (μ_X, μ_Y). The *conditional entropy* of X given an event A, denoted $H(X|A)$, is the entropy of the conditional distribution of μ_X given A. The *conditional entropy* of X given Y is $H(X|Y) = \sum_{y\in\mathcal{Y}} \mu_Y(y) H(X|Y=y)$.

The *mutual information* between X and Y is $I(X\,;Y) = H(X) - H(X|Y)$. The *conditional mutual information* between X and Y given Z is $I(X\,;Y|Z) = H(X|Z) - H(X|Y,Z)$.

The following are basic facts about entropy and mutual information.

Proposition 23 (Entropy bound). *Let X be any random variable and let $supp(X)$ be the support of X. Then, $H(X) \le \log|supp(X)|$. Equality iff X is uniform on $supp(X)$.*

Proposition 24 (Chain rule conditional mutual information). *For any sequence of random variables $X_1, \ldots, X_n$,*

$$I(X_1,\ldots,X_n\,;Y) = \sum_{i=1}^{n} I(X_i\,;Y|X_1,\ldots,X_{i-1}).$$

Fano's inequality Fano's inequality [26] gives a lower bound on the error probability of predicting the value of a random variable X from the observation of another random variable Y. We consider a special case where X is uniformly distributed over a binary domain.

Theorem 25 (Fano's inequality). *Let Y be a random variable and let X be uniformly distributed over $\{0,1\}$. Let $g(Y) \in \{0,1\}$ be a function whose prediction error probability $\Pr(g(Y) \neq X) \le \delta \le 1/2$. Then, $I(X\,;Y) \ge 1 - H_2(\delta)$.*

B. Direct sum for information cost

Suppose G' is a vertex-induced subgraph of G. An INDEX code for G easily yields an INDEX code for G' of the same length by arbitrarily fixing the bits of $V(G) \setminus V(G')$. Thus,

Proposition 26. *If G' is a vertex-induced subgraph of G, then the optimal length of an* INDEX *code for G' is no more than that of G.*

What about the other direction? Suppose we can split G into k mutually disjoint vertex-induced subgraphs $G_1, \ldots, G_k$ and suppose we have k INDEX codes $\mathcal{C}_1, \ldots, \mathcal{C}_k$ for these subgraphs. Clearly, by concatenating $\mathcal{C}_1, \ldots, \mathcal{C}_k$ we can obtain an INDEX code for G whose length is $\sum_{i=1}^{k} \operatorname{len}(\mathcal{C}_i)$. But is this always the optimal length code for G?

In general, it looks like one could obtain shorter INDEX codes for G, by exploiting the edges connecting the different subgraphs $G_1, \ldots, G_k$. But what if these graphs are disconnected from each other? In this case, it seems that the optimal length of the INDEX code for G must equal the sum of the optimal lengths of the INDEX codes for $G_1, \ldots, G_k$. In other words, the optimal length of INDEX codes should admit a *direct sum property*. Nevertheless, proving this property for the measure of code length is elusive. The techniques of Feder *et al.* [15] yield a weaker result, which incurs a loss of an additive term that depends linearly on k. We are able to prove the direct sum property not directly for code length, but rather for the "information cost" of codes:

Definition 27 (Information Cost). Let $\mathcal{C}$ be a randomized index code for G. Let R denote the public random string of $\mathcal{C}$, and let $E(x, R)$ denote the encoding of x in $\mathcal{C}$.[4] Let X be uniformly distributed in $\{0,1\}^n$. The *information cost* of $\mathcal{C}$, denoted by $\operatorname{icost}(\mathcal{C})$, equals $I(X\,;\,E(X,R) \mid R)$.

As the information cost of a code is always at most the entropy of the codewords, the entropy bound (Proposition 23) implies that information cost is a lower bound on the code length.

We prove that the information cost of an INDEX code admits a direct sum property. The property holds not only when $G_1, \ldots, G_k$ are totally disconnected from each other; it suffices that there are no edges directed from G_i to G_j for all $i < j$:

Theorem 28. *Let $G_1, G_2, \ldots, G_k$ be vertex-induced subgraphs of a directed graph G such that:*

1) *The vertices of $G_1, G_2, \ldots, G_k$ partition the vertices of G.*
2) *For any $i < j$ and vertices $v_i \in V(G_i)$ and $v_j \in V(G_j)$, there is no directed edge in G from v_i to v_j.*

Let $\mathcal{C}$ be a δ-error randomized INDEX *code for G. Then, there exist δ-error randomized* INDEX *codes $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ for $G_1, G_2, \ldots, G_k$ such that $\operatorname{icost}(\mathcal{C}) \ge \sum_j \operatorname{icost}(\mathcal{C}_j)$.*

Proof: For $j = 1 \ldots k$, define $V_j = V(G_j)$ and $U_j = \bigcup_{i=1}^{j} V_i$.

Let $E(x, R)$ be the encoding function of the INDEX code $\mathcal{C}$ and let X be uniformly distributed on $\{0,1\}^n$. By definition, $\operatorname{icost}(\mathcal{C}) = I(X\,;\,E(X,R) \mid R)$. Using the chain rule for conditional mutual information,

$$I(X\,;\,E(X,R) \mid R) = \sum_{j=1}^{k} I(X[V_j]\,;\,E(X,R) \mid X[U_{j-1}], R).$$

(Slightly abusing notation, $U_0 = \emptyset$ and $x[U_0]$ is an empty string.) To complete the proof of the theorem, it suffices to show the following claim:

Claim 29. *For every j, there is a δ-error randomized* INDEX *code $\mathcal{C}_j$ for G_j such that*

$$\operatorname{icost}(\mathcal{C}_j) = I(X[V_j]\,;\,E(X,R) \mid X[U_{j-1}], R).$$

Proof: The proof is based on a reduction lemma proven in [17]. Fix a value of j and we will construct $\mathcal{C}_j$ using $\mathcal{C}$ as follows. Let $a \in \{0,1\}^{|V_j|}$ denote the source input. As we want to use $\mathcal{C}$, we need to transform a into some input x for $\mathcal{C}$. The transformation will be randomized. That is, x will be a random string, created from a, from the public random string, and from the private random string of the encoder.

x will be equal to a in the coordinates corresponding to vertices in G_j. The other coordinates of x will be filled randomly as follows. Let B have the same distribution as $X[V \setminus U_j]$ and let C be independent of B and have the same distribution as $X[U_{j-1}]$. The public random string for $\mathcal{C}_j$ consists of (R, C) while B will be part of the private randomness of the sender. The random input x will be defined to be the tuple $\langle C, a, B\rangle$. The encoding of a is then $E(\langle C, a, B\rangle, R)$. Note that this encoding is a function of a, of the sender's private random string, and of the public random string.

Let $i \in V_j$ be any coordinate. When applying the decoding function D_i of $\mathcal{C}$ in order to recover a_i, the receiver needs to know the bits of x corresponding to neighbors of i in the graph G. By the property of G_j, it can be seen that the neighbors of i in G are either among the neighbors of i in V_j or belong to U_{j-1}. Now, the values for the former are part of the side information for coordinate i while the values for the latter can be found in the public random string C.

For any instantiation of B and C, the decoding error is simply the error of $\mathcal{C}$ on the input x obtained from a and from the instantiations of B and C. As this error is at most δ, then also averaging over all choices of B and C, the error of $\mathcal{C}_j$ on a is at most δ.

Next, we calculate the information cost of $\mathcal{C}_j$ as follows. Let A be uniformly distributed over $\{0,1\}^{|V_j|}$ and be independent of B and C. Then,

$$\begin{aligned} \operatorname{icost}(\mathcal{C}_j) &= I(A\,;\,E(\langle C, A, B\rangle, R) \mid C, R) \\ &= I(X[V_j]\,;\,E(X,R) \mid X[U_{j-1}], R), \end{aligned}$$

completing the proof of the claim. ■

Applying the above claim for all j completes the proof of the theorem. ■

[4] The dependence of E on the sender's private randomness is suppressed for ease of presentation.

C. Lower bound for randomized codes

Theorem 7 can now be shown as a simple application of the above Theorem 28.

Theorem 7 (restated) *The length of any δ-error randomized* INDEX *code for G is at least* $\text{MAIS}(G) \cdot (1 - H_2(\delta))$, *where* $\text{MAIS}(G)$ *is the size of the maximum acyclic induced subgraph of G and $H_2(\cdot)$ is the binary entropy function.*

Proof: Let G' be a maximal acyclic induced subgraph of G. By Proposition 26, it suffices to consider any INDEX code $\mathcal{C}$ for G'. Let $u_1, u_2, \ldots, u_k$ denote the vertices of G' such that there is no edge from u_i to u_j whenever $i < j$. Apply Theorem 28 with $G = G'$ and where G_j is a graph with a single vertex u_j. We have $\text{icost}(\mathcal{C}) \geq \sum_j \text{icost}(\mathcal{C}_j)$. Now, since $\mathcal{C}_j$ is a INDEX code for a single vertex graph, therefore, it encodes just a single bit that can be decoded with probability of error at most δ. By Fano's inequality, it must have at least $1 - H_2(\delta)$ bits of information. ■

VI. Lower bounds for restricted graphs

In this section we show that for certain natural classes of graphs, the minrank bound is tight w.r.t. *arbitrary* INDEX codes.

Theorem 8 (restated) *Let G be any graph, which is either a DAG, a perfect graph, an odd hole, or an odd anti-hole. Then, the length of any* INDEX *code for G is at least* $\text{minrk}_2(G)$.

A. Directed acyclic graphs

A *directed acyclic graph* (DAG) is one without directed cycles.

Proposition 30. *Let G be any DAG on n nodes. Then, the length of any* INDEX *code for G is at least* $\text{minrk}_2(G)$.

Proof: Let $\mathcal{C}$ be any INDEX code for G. Since G is a DAG, then $\text{MAIS}(G) = n$. Hence, by Theorem 7, $\text{len}(\mathcal{C}) \geq n$. Clearly, $\text{minrk}_2(G) \leq n$, and thus $\text{len}(\mathcal{C}) \geq \text{minrk}_2(G)$. ■

B. Perfect graphs

An undirected graph G is called *perfect*, if for any induced subgraph G' of G, $\omega(G') = \chi(G')$.

Proposition 31. *Let G be any perfect graph on n nodes. Then, the length of any* INDEX *code for G is at least* $\text{minrk}_2(G)$.

Proof: Let $\mathcal{C}$ be any INDEX code for G. By Theorem 7, $\text{len}(\mathcal{C}) \geq \text{MAIS}(G)$. Since G is undirected, then $\text{MAIS}(G) = \alpha(G)$, i.e., the independence number of G. Clearly, $\alpha(G) = \omega(\overline{G})$, implying that $\text{len}(\mathcal{C}) \geq \omega(\overline{G})$.

Lovász [27] proved in 1972 the "Perfect Graph Theorem", stating that a graph G is perfect if and only if its complement is perfect. Now, since G is perfect, then by this theorem also $\overline{G}$ is perfect, implying that in particular $\omega(\overline{G}) = \chi(\overline{G})$. Hence, $\text{len}(\mathcal{C}) \geq \chi(\overline{G})$. However, by the sandwich property of minrank (Proposition 4), $\text{minrk}_2(G) \leq \chi(\overline{G})$ and thus $\text{len}(\mathcal{C}) \geq \text{minrk}_2(G)$. ■

C. Odd holes

Before we prove the lower bound for odd holes, we first characterize their minrank:

Theorem 32. *Let G be an odd hole of length $2n+1$ ($n \geq 2$). Then,* $\text{minrk}_2(G) = n + 1$.

Note that since for an odd hole G, $\omega(\overline{G}) = n$, odd holes are examples of graphs for which $\omega(\overline{G}) < \text{minrk}_2(G)$.

Proof: As $\chi(\overline{G}) = n+1$ for an odd hole of length $2n+1$ and as $\text{minrk}_2(G) \leq \chi(\overline{G})$ (Proposition 4), it suffices to prove that $\text{minrk}(G) \geq n + 1$.

Fix any matrix A that fits G. For convenience, we number the rows and columns of A as $0, 1, \ldots, 2n$ and make all the index arithmetic below modulo $2n$. Let $A_0, \ldots, A_{2n}$ be the $2n + 1$ rows of A. A has the following three properties, for every i,

1) $A_i[i] = 1$.
2) $A_i[i - 1], A_i[i + 1] \in \{0, 1\}$.
3) $A_i[j] = 0$, for $j \notin \{i - 1, i, i + 1\}$.

For a row A_i, we call the rows $A_0, \ldots, A_{i-1}$ the "predecessors of A_i". Note that A_0 has no predecessors. We next prove the following two claims:

Claim 33. *For $i = 1, \ldots, 2n - 2$, either A_i is linearly independent of its predecessors or A_{i+1} is linearly independent of its predecessors.*

Proof: Suppose, to reach a contradiction, that the claim is false. Hence, there exists some $i \in \{1, \ldots, 2n - 2\}$ s.t. both A_i and A_{i+1} linearly depend on their predecessors. It follows that both A_i and A_{i+1} linearly depend on $A_0, \ldots, A_{i-1}$. Since $A_0[i+1] = \cdots = A_{i-1}[i+1] = 0$ (using Property 3 of A and the fact $i+1 < 2n$), then also $A_i[i+1] = 0$ and $A_{i+1}[i+1] = 0$. This contradicts the fact $A_{i+1}[i + 1] = 1$ (Property 1 of A). ■

Claim 34. *At least one among A_1, A_{2n-1}, A_{2n} is linearly independent of its predecessors.*

Proof: If at least one of A_{2n-1}, A_{2n} is independent of its predecessors, then we are done. So suppose both depend on their predecessors. As argued above, this means that A_{2n-1}, A_{2n} both depend on $A_0, \ldots, A_{2n-2}$.

By Property 1 of A, $A_{2n}[2n] = 1$. The only vector among $A_0, \ldots, A_{2n-2}$ that can have a 1 at the $2n$-th coordinate is A_0. Thus, we must have: $A_0[2n] = 1$. By Property 3 of A, $A_1[2n] = 0$. Hence, A_1 cannot depend on its sole predecessor, A_0. We thus obtained that A_1 is linearly independent of its predecessors. ■

Note that in Claim 34 we implicitly use the fact $n \geq 2$, because we assume the indices $1, 2n - 1, 2n$ are distinct.

We next use the above two claims to count the number of rows of A that must be linearly independent of their predecessors. For each i, let $Z_i = 1$ if the i-th row is independent of its predecessors and $Z_i = 0$ otherwise. The number of rows that are linearly independent of their predecessors is $\sum_{i=0}^{2n} Z_i$. Note that this number is exactly the 2-rank of the matrix A.

We know the following three facts about the sequence $Z_0, \ldots, Z_{2n}$:

1) $Z_0 = 1$, because A_0 simply does not have any predecessors.
2) For each $i = 1, \ldots, n-3$, $Z_i + Z_{i+1} \geq 1$ (Claim 33).
3) $Z_1 + Z_{2n-1} + Z_{2n} \geq 1$ (Claim 34).

We now write the sum $2\sum_{i=0}^{2n} Z_i$ as follows:

$$2\sum_{i=0}^{2n} Z_i = 2Z_0 + Z_1 + \sum_{i=1}^{2n-2}(Z_i + Z_{i+1}) + Z_{2n-1} + 2Z_{2n}.$$

Using the above three facts, we have:

$$2\sum_{i=0}^{2n} Z_i \geq 2 + 2n - 2 + 1 + Z_{2n} \geq 2n + 1.$$

Therefore, $\sum_{i=0}^{2n} Z_i \geq (2n+1)/2$. However, since $\sum_{i=0}^{2n} Z_i$ is an integer we have the stronger bound:

$$\sum_{i=0}^{2n} Z_i \geq \lceil (2n+1)/2 \rceil = n + 1.$$

Hence, $\mathrm{rk}_2(A) = \sum_{i=0}^{2n} Z_i \geq n+1$. As this holds for any A that fits G, also $\mathrm{minrk}_2(G) \geq n+1$. ■

The lower bound for odd holes is then the following:

Theorem 35. *Let G be an odd hole on $2n+1$ nodes ($n \geq 2$). Then, the length of any* INDEX *code for G is at least* $\mathrm{minrk}_2(G) = n + 1$.

As for an odd hole G, $\mathrm{MAIS}(G) = \omega(\overline{G}) = n < \mathrm{minrk}_2(G)$, this theorem implies that our lower bound for general INDEX codes (Theorem 7) is not tight.

The proof of this lower bound is considerably harder than the proofs for DAGs and perfect graphs. To this end, we need to study some combinatorial properties of the *confusion graph* associated with INDEX coding.

Definition 36 (Confusion graph). The *confusion graph* $C(G)$ associated with INDEX coding for a directed graph G (abbreviated "confusion graph for G") is an *undirected* graph on $\{0,1\}^n$ such that x and x' are connected by an edge if for some i, we have $x[N(i)] = x'[N(i)]$ but $x_i \neq x'_i$.

If x and x' are connected by an edge in $C(G)$, then no INDEX code $\mathcal{C}$ for G can map x and x' to the same codeword, implying $\log\chi(C(G))$ is a lower bound on $\mathrm{len}(\mathcal{C})$.

Notation. Let $\mathbf{0}$ and $\mathbf{1}$ denote, respectively, the all-zero and the all-one vectors. Let $\mathbf{1}_S$ denotes the characteristic vector of a set $S \subseteq [n]$.

Lemma 37. *Let G be an undirected graph on n nodes and let $C(G)$ be the confusion graph corresponding to* INDEX *coding for G. Then,*

1) *If S is a vertex cover of G, then any two inputs $x, x' \in \{0,1\}^n$ that agree on S (i.e., $x[S] = x'[S]$) are connected by an edge in $C(G)$.*
2) *If S is an independent set in G, then the set $X_S = \{\mathbf{1}_T \mid T \subseteq S\}$ forms a clique in $C(G)$.*
3) *If S, T are two disjoint and independent sets in G, and there exists some $i \in S$ that has no neighbors in T or some $j \in T$ that has no neighbors in S, then the inputs $\mathbf{1}_S$ and $\mathbf{1}_T$ are connected by an edge in $C(G)$.*

Proof of Part 1:: Since $x \neq x'$, there exists some index $i \in [n]$ s.t. $x_i \neq x'_i$. This means that $i \notin S$. If a node does not belong to a vertex cover, then all its neighbors must belong to the vertex cover. We conclude that $N(i) \subseteq S$ and thus $x[N(i)] = x'[N(i)]$. This implies that x and x' are connected by an edge in the confusion graph. ■

Proof of Part 2:: Define $U = [n] \setminus S$. Since S is an independent set in G, then U is a vertex cover. Note that any two input $x, x' \in X_S$ agree on U, and thus by Part 1 must be connected by an edge in the confusion graph. ■

Proof of Part 3:: Suppose, for example, there is $i \in S$ that has no neighbors in T. Since S, T are disjoint, $\mathbf{1}_S$ and $\mathbf{1}_T$ disagree on the i-th coordinate. Since S is independent, $N(i) \subseteq [n] \setminus S$, and thus $\mathbf{1}_S[N(i)] = \mathbf{0}$. Since $N(i) \cap T = \emptyset$, then also $\mathbf{1}_T[N(i)] = \mathbf{0}$. This implies that $\mathbf{1}_S[N(i)] = \mathbf{1}_T[N(i)]$ and therefore $\mathbf{1}_S, \mathbf{1}_T$ must be connected by an edge in the confusion graph. ■

We can now prove Theorem 35:

Proof of Theorem 35: Let G be an odd hole on $2n+1$ nodes ($n \geq 2$). Let $\mathcal{C}$ be any INDEX code for G. We will prove that $|\mathcal{C}|$, the number of codewords in $\mathcal{C}$, is greater than 2^n, implying that $\mathrm{len}(\mathcal{C}) \geq n+1 = \mathrm{minrk}_2(G)$.

Consider the following coloring of G: $S_1 = \{1, 3, \ldots, 2n-1\}$, $S_2 = \{2, 4, \ldots, 2n\}$ and $S_3 = \{2n+1\}$. For each $i \in \{1,2,3\}$, since S_i is an independent set, then by Part 2 of Lemma 37, $\mathcal{C}$ must use $2^{|S_i|}$ different codewords to encode inputs in X_{S_i}. Since $|S_1| = |S_2| = n$, this already implies $|\mathcal{C}| \geq 2^n$. Assume, to the contradiction, that $|\mathcal{C}| = 2^n$.

Since S_1, S_2, S_3 are pairwise disjoint, then the sets $X_{S_1}, X_{S_2}, X_{S_3}$ have only $\mathbf{0}$ as a common input and are otherwise pairwise disjoint. Since $|\mathcal{C}| = 2^n$, and no codeword can encode two different inputs in X_{S_i} ($i = 1,2,3$), then there must be at least one codeword encoding a nonzero input from X_{S_1}, a nonzero input from X_{S_2}, and a nonzero input from X_{S_3}. We call these inputs x_1, x_2, x_3.

We view x_1, x_2, x_3 as characteristic vectors of sets $T_1, T_2, T_3 \subseteq [n]$. Since $x_1, x_2, x_3 \neq \mathbf{0}$, then $T_1, T_2, T_3 \neq \emptyset$. Furthermore, they are all independent and pairwise disjoint. Since the only nonzero vector in X_{S_3} is e_{2n+1}, $T_3 = \{2n+1\}$.

Since x_1, x_2, x_3 are encoded by the same codeword, no two of them can be connected by an edge in the confusion graph. Consider any $i \in T_1$. By Part 3 of Lemma 37, i must have a neighbor $j \in T_2$. Similarly, both i and j must have neighbors in T_3. Since $T_3 = \{2n+1\}$, both are neighbors of $2n+1$. We conclude that $(i, j, 2n+1)$ forms a triangle in G. However, all odd holes are triangle-free. This is a contradiction, and thus $|\mathcal{C}| > 2^n$. ■

The above theorem provides a tight lower bound on the *length* of INDEX codes for odd holes, but not on their *size*. Our upper bound (Theorem 5) gives a code whose size is 2^{n+1}, while the above proof only shows a lower bound of $|\mathcal{C}| > 2^n$. Optimal code size lower bounds are important for deriving lower bounds on the average encoding length and on the information cost. Resorting to a more involved combinatorial argument, we are able to prove tight bounds (i.e., 2^{n+1}) on the size of INDEX codes for odd holes of length at least 7:

Theorem 38. *Let G be an odd hole on $2n+1$ nodes ($n \geq 3$).*

Then, the size of any INDEX *code for G is at least* $2^{\mathrm{minrk}(G)} = 2^{n+1}$.

The proof of this theorem appears in Appendix A.

Dealing with the pentagon (a hole of length 5) turns out to be very tricky. The difficulty of handling the pentagon stems from the fact that the corresponding confusion graph has a rather peculiar property. In most cases, one can obtain tight lower bounds on the chromatic number of the confusion graph by obtaining tight upper bounds on the graph's independence number. It turns out that this approach fails for the pentagon. The size of the pentagon's confusion graph is 32 and its chromatic number is 8. Yet, the code we show below for the pentagon demonstrates that the independence number of its confusion graph is 5, implying that $32/5 < 8$ is not a tight lower bound on the chromatic number.

Codeword	Inputs
C_1	00000, 00110, 10001, 11101, 11110
C_2	11111, 11001, 01110, 00010, 00001
C_3	01010, 01100, 11011, 10111, 10100
C_4	00100, 01011, 10010, 10101
C_5	00111, 01001, 10110, 11010
C_6	01000, 01111, 10000, 10011
C_7	00011, 00101, 11000
C_8	01101, 11100

By applying arguments from the proof of theorem 38 we can obtain a lower bound of 7 on the size of codes for the pentagon, one short of the upper bound of 8. By the same arguments, any INDEX code of size 7 for the pentagon must adhere to certain structural constraints. By a brute force exhaustive search over such codes, we verified that 8 is the tight lower bound.

D. *Odd anti-holes*

Recall that an odd anti-hole is the complement graph of an odd hole. We prove a tight lower bound on the minimum length of codes for odd anti-holes. This bound does not give a tight lower bound on the size of codes for odd anti-holes. Unfortunately, we could not prove tight bounds on the size.

Theorem 39. *Let G be an odd anti-hole on $2n + 1$ nodes ($n \geq 2$). Then, the length of any* INDEX *code for G is at least* $\mathrm{minrk}(G) = 3$.

Proof: We use the same notation and propositions as in the proof for odd holes. Let $\mathcal{C}$ be any INDEX code for G. We would like to show that $|\mathcal{C}| \geq 5$. That would imply that $\mathrm{len}(\mathcal{C}) \geq 3$.

An odd anti-hole of length $2n + 1$ is $(n + 1)$-colorable. Consider the following coloring of G: $S_1 = \{1\}, S_2 = \{2, 3\}, \ldots, S_{n+1} = \{2n, 2n + 1\}$. Let $X_{S_1}, \ldots, X_{S_{n+1}}$ be the input sets corresponding to $S_1, \ldots, S_{n+1}$. Note that these sets share a single input (the all-zero input) and are otherwise pairwise disjoint.

For $i = 2, \ldots, n + 1$, $|X_{S_i}| = 4$, and thus by Part 2 of Lemma 37, $\mathcal{C}$ must use 4 different codewords for each of these sets. Assume, to reach a contradiction, that $|\mathcal{C}| = 4$. Therefore, there must be a single codeword that encodes a nonzero input from X_{S_i}, for each $i = 1, \ldots, n + 1$. Let us denote these inputs by $x_1, \ldots, x_{n+1}$. We view $x_1, \ldots, x_{n+1}$ as characteristic vectors of sets $T_1, \ldots, T_{n+1}$. These sets are all independent and pairwise disjoint. Furthermore, $T_1 = \{1\}$, because the only nonzero input in X_{S_1} is e_1.

Since $x_1, \ldots, x_{n+1}$ are all encoded by a single input, they must form an independent set in the confusion graph $C(G)$. We next prove by induction that for every $i = 1, \ldots, n + 1$, T_i must be the set $\{2i - 1\}$.

For $i = 1$, we already know that $T_1 = \{1\}$. Assume correctness for i. We will show correctness for $i + 1$. Since x_i and x_{i+1} are not connected by an edge in the confusion graph, then by Part 3 of Lemma 37, every node in T_i must have a neighbor in T_{i+1} and vice versa. Since $T_i = \{2i-1\}$ and since the only neighbor of $2i - 1$ in the set $S_{i+1} = \{2i, 2i + 1\}$ is $2i + 1$, then T_{i+1} must be $\{2i + 1\}$.

It follows that $T_{n+1} = \{2n + 1\}$. However, since nodes 1 and $2n+1$ are not neighbors in G, it follows that no node in T_1 has neighbors in T_{n+1}. Thus, by Part 3 of Lemma 37, x_1 and x_{n+1} must be connected by an edge in the confusion graph, in contradiction to the fact $x_1, \ldots, x_{n+1}$ is an independent set in the confusion graph. Therefore, $|\mathcal{C}| \geq 5$. ■

VII. Conclusions

In this paper, we explored upper and lower bounds on the length of INDEX codes for $\{0, 1\}^n$ with side information graph G. We identified a measure on graphs, the *minrank*, which we showed to characterize the length of INDEX codes for natural classes of graphs (DAGs, perfect graphs, odd holes, and odd anti-holes). We also proved that minrank characterizes the minimum length of natural types of INDEX codes (linear, linearly-decodable, and semi-linearly-decodable) for *arbitrary* graphs. For general codes and general graphs, we were able to obtain a weaker bound in terms of the maximum acyclic induced subgraph. Finally, we proved a direct sum theorem for the information cost of INDEX codes with side information.

As Lubetzky and Stav [24] have recently shown, the minrank is not a tight lower bound on the length of a general INDEX code for arbitrary graphs. Characterizing the optimal length of INDEX codes for arbitrary graphs therefore remains an open problem. It is nonetheless important to note that virtually all codes presently in use are linear, and for those our bounds are tight.

The minrank by itself is an interesting subject of study. We know that for undirected graphs, it is bounded from below by the Shannon capacity and from above by the chromatic number of the complement graph. It would be interesting to explore further properties of minrank with respect to other graph measures such as the Lovász Theta function.

Finally, a practical conclusion is that keeping "junk" (unneeded information) may be beneficial, as it can serve as side information and save communication. This is particularly true in view of the declining cost of storage space.

References

[1] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. IT-19, pp. 471–480, 1973.

[2] A. Wyner and J. Ziv, “A theorem on the entropy of certain binary sequences and applications I,” *IEEE Transactions on Information Theory*, vol. IT-19, pp. 769–771, 1973.

[3] A. Wyner, “A theorem on the entropy of certain binary sequences and applications II,” *IEEE Transactions on Information Theory*, vol. IT-19, pp. 772–777, 1973.

[4] H. S. Witsenhausen, “The zero-error side information problem and chromatic numbers,” *IEEE Transactions on Information Theory*, vol. 22, no. 5, pp. 592–593, 1976.

[5] N. Alon and A. Orlitsky, “Source coding and graph entropies,” *IEEE Transactions on Information Theory*, vol. 42, no. 5, pp. 1329–1339, 1996.

[6] P. Koulgi, E. Tuncel, S. L. Regunathan, and K. Rose, “On zero-error source coding with decoder side information,” *IEEE Transactions on Information Theory*, vol. 49, no. 1, pp. 99–111, 2003.

[7] Y. Birk and T. Kol, “Coding-On-Demand by an Informed Source (ISCOD) for Efficient Broadcast of Different Supplemental Data to Caching Clients,” *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2825–2830, 2006, early version appeared in INFOCOM ’98.

[8] W. H. Haemers, “On some problems of Lovász concerning the shannon capacity of a graph,” *IEEE Transactions on Information Theory*, vol. 25, no. 2, pp. 231–232, 1979.

[9] A. E. Brouwer and C. A. van Eijl, “On the p-ranks of the adjacency matrices of strongly regular graphs,” *Journal of Algebraic Combinatorics*, vol. 1, pp. 329–346, 1992.

[10] R. Peeters, “On the p-ranks of the adjacency matrices of distance-regular graphs,” *Journal of Algebraic Combinatorics*, vol. 15, no. 2, pp. 127–149, 2002.

[11] ——, “Orthogonal representations over finite fields and the chromatic number of graphs,” *Combinatorica*, vol. 16, no. 3, pp. 417–431, 1996.

[12] W. H. Haemers, “An upper bound for the Shannon capacity of a graph,” *Algebraic methods in Graph Theory*, vol. 25, pp. 267–272, 1978.

[13] C. Berge, “Färbung von graphen, deren sämtliche bzw. deren ungerade kreise starr sind,” *Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math.-Natur. Reihe*, vol. 10, 1961.

[14] M. Chudnovsky, N. Robertson, P. Seymour, and R. Thomas, “The strong perfect graph theorem,” *Annals of Mathematics*, vol. 164, pp. 51–229, 2006.

[15] T. Feder, E. Kushilevitz, M. Naor, and N. Nisan, “Amortized communication complexity,” *SIAM Journal on Computing*, vol. 24, no. 4, pp. 736–750, 1995.

[16] A. Chakrabarti, Y. Shi, A. Wirth, and A. C.-C. Yao, “Informational complexity and the direct sum problem for simultaneous message complexity,” in *Proceedings of the 42nd IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, 2001, pp. 270–278.

[17] Z. Bar-Yossef, T. S. Jayram, R. Kumar, and D. Sivakumar, “An information statistics approach to data stream and communication complexity,” *J. Computer and System Sciences*, vol. 68, no. 4, pp. 702–732, 2004.

[18] Z. Bar-Yossef, T. S. Jayram, R. Krauthgamer, and R. Kumar, “The sketching complexity of pattern matching,” in *Proceedings of the 8th International Workshop on Randomization and Computation (RANDOM)*, 2004, pp. 261–272.

[19] A. Ambainis, A. Nayak, A. Ta-Shma, and U. Vazirani, “Dense quantum coding and quantum finite automata,” *J. ACM*, vol. 49, no. 4, pp. 496–511, 2002.

[20] I. Kremer, N. Nisan, and D. Ron, “On randomized one-round communication complexity,” *Computational Complexity*, vol. 8, no. 1, pp. 21–49, 1999.

[21] R. W. Yeung and Z. Zhang, “Distributed source coding for satellite communications,” *IEEE Transactions on Information Theory*, vol. 45, pp. 1111–1120, 1999.

[22] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” *IEEE Trans. Inform. Theory*, vol. 46, pp. 1204–1216, 2000.

[23] Z. Bar-Yossef, Y. Birk, T. S. Jayram, and T. Kol, “Index coding with side information,” in *Proceedings of the 47th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, 2006, pp. 197–206.

[24] E. Lubetzky and U. Stav, “Non-linear index coding outperforming the linear optimum,” Preprint, 2007.

[25] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.

[26] R. M. Fano, Class Notes for Transmission of Information, 1952, course 6.574, MIT, Cambridge, MA.

[27] L. Lovász, “A characterization of perfect graphs,” *Journal of Combinatorial Theory, Series B*, vol. 13, pp. 95–98, 1972.

Appendix A
Size lower bound for odd holes

Theorem 38 (restated) *Let G be an odd hole on $2n+1$ nodes ($n \geq 3$). Then, the size of any* INDEX *code for G is at least $2^{\mathrm{minrk}(G)} = 2^{n+1}$.*

Proof: Our strategy for proving the theorem will be by showing that the independence number of the confusion graph $C(G)$ is at most 2^n. Since $|C(G)| = 2^{2n+1}$, it will immediately follow that $\chi(C(G)) \geq 2^{n+1}$, giving us the desired lower bound on the size of the INDEX code.

In the derivations below, we assume the nodes of G are numbered $0, 1, 2, \ldots, 2n$. All the index arithmetics are done modulo $2n$.

Let X be any independent set of inputs in the confusion graph. We would like to prove that $|X| \leq 2^n$. For any set $S \subseteq [n]$ of coordinates, define $X[S]$ to be the set obtained by projecting all the inputs in X on the coordinates in S: $X[S] = \{x[S] \mid x \in X\}$. (When S is a sequence of indices $\{i_1, \ldots, i_k\}$, we write $X[i_1, \ldots, i_k]$ as a shorthand for $X[\{i_1, \ldots, i_k\}]$.)

As an immediate corollary of Part 1 of Lemma 37, we obtain the following:

Proposition 40. *Let G be an undirected graph on n nodes, and let $X \subseteq \{0,1\}^n$ be an independent set in the confusion graph $C(G)$ for G. Then, for any vertex cover $S \subseteq [n]$ of G, $|X| \leq |X[S]|$.*

It is possible to strengthen Proposition 40. Given an undirected graph G on n nodes and a subset T of its vertices, we say that a subset S of T is a *vertex cover of T*, if for every node $v \in T$, either $v \in S$ or $N(v) \subseteq S$ (where $N(v)$ is the set of all the neighbors of v in G). For example, if G is a cycle, and $T = \{i, i+1, i+2\}$, then $S = \{i, i+2\}$ is a vertex cover of T. The following is a straightforward generalization of Proposition 40:

Proposition 41. *Let G be an undirected graph on n nodes, and let $X \subseteq \{0,1\}^n$ be an independent set in the confusion graph $C(G)$ for G. Then, for any subset $T \subseteq [n]$, and for any vertex cover S of T, $|X[T]| \leq |X[S]|$.*

Proof: Suppose, to reach a contradiction, $|X[T]| > |X[S]|$. Then, by the Pigeonhole Principle, there exist two inputs $x, x' \in X$ s.t. $x[S] = x'[S]$ but $x[T] \neq x'[T]$. That is, there is some $j \in T \setminus S$, s.t. $x[j] \neq x'[j]$. Yet, since $j \notin S$, $N(j) \subseteq S$, and thus x and x' agree on all the neighbors of j but disagree on j. This means that x and x' are connected by an edge in the confusion graph, in contradiction to the assumption they are both members of an independent set. ■

Now, in order to bound $|X|$, we consider three cases.

Case 1: $\exists i \in [0, 2n]$ s.t. $|X[i, i+1]| \leq 2$.

Without loss of generality, assume $i = 0$. We construct a vertex cover S of G as follows: $S = \{0, 1, 3, 5, 7, \ldots, 2n-1\}$. Note that S is indeed a vertex cover of G and that $|S| = n+1$. We split it into two parts: $S_1 = \{0, 1\}$ and $S_2 = \{3, 5, \ldots, 2n-1\}$. Clearly, $|X[S]| \leq |X[S_1]| \cdot |X[S_2]|$. Since $|S_2| = n-1$, then $|X[S_2]| \leq 2^{n-1}$. Therefore, $|X[S]| \leq$

$|X[S_1]| \cdot |X[S_2]| \leq 2 \cdot 2^{n-1} \leq 2^n$. Using Proposition 40 we have in this case: $|X| \leq 2^n$.

Case 2: $\exists i \in [0, 2n]$ s.t. $|X[i, i+1]| = 4$.

WLOG, assume $i = 1$. Hence, $|X[1, 2]| = 4$. Since $\{1, 3\}$ is a vertex cover of the set $\{1, 2, 3\}$, then by Proposition 41, $|X[1, 2, 3]| \leq |X[1, 3]| \leq 4$. On the other hand, since $\{1, 2\}$ is a subset of $\{1, 2, 3\}$, $4 = |X[1, 2]| \leq |X[1, 2, 3]| \leq 4$. Hence, $|X[1, 2, 3]| = |X[1, 2]| = 4$, and thus if any two inputs $x, x' \in X$ agree on positions 1 and 2, they also must agree on position 3. Analogously, we have $|X[0, 1, 2]| = |X[1, 2]| = 4$, and thus any two inputs that agree on positions 1 and 2 also agree on position 0. Now, since the bits at positions 1 and 2 completely determine the bits at positions 0 and 3, we have: $|X[0, 1, 2, 3]| = 4$.

Consider now the following vertex cover of G: $S = \{0, 1, 3, 5, 7, \ldots, 2n-1\}$. We split it into two parts: $S_1 = \{0, 1, 3\}$ and $S_2 = \{5, 7, \ldots, 2n-1\}$. We obtain: $|X[S]| \leq |X[S_1]| \cdot |X[S_2]|$. Since $S_1 \subseteq \{0, 1, 2, 3\}$, $|X[S_1]| \leq |X[0, 1, 2, 3]| = 4$. Since $|S_2| = n - 2$, $|X[S_2]| \leq 2^{n-2}$. Therefore, $|X[S]| \leq 4 \cdot 2^{n-2} = 2^n$. Applying now Proposition 40, we have: $|X| \leq |X[S]| \leq 2^n$.

Case 3: $\forall i \in [0, 2n]$, $|X[i, i+1]| = 3$. We split the analysis of this case into two sub-cases.

Sub-case 3.1: $n \geq 4$. That is, G is an odd hole of length at least 9. Our goal in this case is to show that $|X[0, 1, \ldots, 7]| \leq 16$. If we do that, then we can construct a vertex cover of G as follows: $S = S_1 \cup S_2$, where $S_1 = \{0, 1, 3, 5, 7\}$ and $S_2 = \{9, 11, \ldots, 2n-1\}$. We obtain: $|X| \leq |X[S]| \leq |X[S_1]| \cdot |X[S_2]| \leq |X[0, 1, \ldots, 7]| \cdot 2^{|S_2|} \leq 16 \cdot 2^{n-4} = 2^n$.

In order to bound $|X[0, 1, \ldots, 7]|$, we develop a recursive expression for $|X[0, 1, \ldots, i+1]|$, for any $i = 0, \ldots, 2n-1$.

Fix some $i \in [0, 2n-1]$ and denote the three bitstrings in $X[i, i+1]$ by $a_i b_i$, $a_i \bar{b}_i$, and $\bar{a}_i b_i$, where $a_i, b_i \in \{0, 1\}$. Note that exactly two of these bitstrings agree on position i and exactly two agree on position $i+1$. a_i is the majority bit at position i and b_i is the majority bit at position $i+1$. We split the set $X[0, 1, \ldots, i+1]$ into three parts accordingly:

$$\begin{aligned}
\mathcal{A}_i &= \{x[0, 1, \ldots, i+1] \mid x \in X, x[i, i+1] = a_i b_i\}, \\
\mathcal{B}_i &= \{x[0, 1, \ldots, i+1] \mid x \in X, x[i, i+1] = a_i \bar{b}_i\}, \text{ and} \\
\mathcal{C}_i &= \{x[0, 1, \ldots, i+1] \mid x \in X, x[i, i+1] = \bar{a}_i b_i\}.
\end{aligned}$$

Clearly, $|X[0, 1, \ldots, i+1]| = |\mathcal{A}_i| + |\mathcal{B}_i| + |\mathcal{C}_i|$. We will develop recursive expressions for $|\mathcal{A}_i|, |\mathcal{B}_i|, |\mathcal{C}_i|$ and use them to bound $|X[0, 1, \ldots, i+1]|$.

Recall that $a_i b_i$, $a_i \bar{b}_i$, and $\bar{a}_i b_i$ are the three bitstrings constituting $X[i, i+1]$. We would like to explore next how these bitstrings can be extended into bitstrings in $X[i, i+1, i+2]$.

We next argue that $a_i b_i$ and $a_i \bar{b}_i$ cannot be extended using the same bit into bitstrings in $X[i, i+1, i+2]$. If there exists a bit c_i s.t. both $a_i b_i c_i$ and $a_i \bar{b}_i c_i$ belong to $X[i, i+1, i+2]$, then there exist two inputs $x, x' \in X$ s.t. $x[i, i+1, i+2] = a_i b_i c_i$ and $x'[i, i+1, i+2] = a_i \bar{b}_i c_i$. That is, $x[i+1] \neq x'[i+1]$, but x, x' agree on the two neighbors of $i+1$. Therefore, x, x' are connected by an edge in the confusion graph $C(G)$, in contradiction to the assumption both belong to an independent set. We conclude that $a_i b_i$ and $a_i \bar{b}_i$ must be extended by complementary bits into bitstrings in $X[i, i+1, i+2]$.

The above implies there exists some bit c_i s.t. $X[i, 1+1, i+2]$ consists of the following bitstrings:

$$\{\ a_i b_i c_i,\ \ a_i \bar{b}_i \bar{c}_i,\ \ \bar{a}_i b_i \bar{c}_i\ \}.$$

It may optionally contain also the following bitstring:

$$\bar{a}_i b_i c_i.$$

(Note that it cannot consist only of the bitstrings $\{a_i b_i c_i, a_i \bar{b}_i \bar{c}_i, \bar{a}_i b_i c_i\}$, because then $|X[i+1, i+2]| = 2$.)

Since there is a single bitstrings in $X[i, i+1, i+2]$ in which bit $i+1$ is $\bar{b}_i$, b_i must be the majority bit at position $i+1$ w.r.t. the bitstrings in $X[i+1, i+2]$. In other words, $a_{i+1} = b_i$. Similarly, it can be seen that $\bar{c}_i$ must be the majority bit at position $i+2$ w.r.t. the bitstrings in $X[i+1, i+2]$. Hence, $b_{i+1} = \bar{c}_i$.

We conclude the following:

1) If $x \in X$ and $x[i+1, i+2] = a_{i+1} b_{i+1} = b_i \bar{c}_i$, then necessarily $x[i, i+1] = \bar{a}_i b_i$. Therefore, $|\mathcal{A}_{i+1}| \leq |\mathcal{C}_i|$.
2) If $x \in X$ and $x[i+1, i+2] = a_{i+1} \bar{b}_{i+1} = b_i c_i$, then $x[i, i+1] = a_i b_i$ or $x[i, i+1] = \bar{a}_i b_i$. Therefore, $|\mathcal{B}_{i+1}| \leq |\mathcal{A}_i| + |\mathcal{C}_i|$.
3) If $x \in X$ and $x[i+1, i+2] = \bar{a}_{i+1} b_{i+1} = \bar{b}_i \bar{c}_i$, then necessarily $x[i, i+1] = a_i \bar{b}_i$. Therefore, $|\mathcal{C}_{i+1}| \leq |\mathcal{B}_i|$.

For $i = 0$, we have $|\mathcal{A}_0| = |\mathcal{B}_0| = |\mathcal{C}_0| = 1$. Hence, $\mathcal{A}_i, \mathcal{B}_i, \mathcal{C}_i$ form Fibonacci-like series. Since we need the value of the series only at $i = 6$, we expand their prefixes explicitly in Table I.

i	$\lvert\mathcal{A}_i\rvert$	$\lvert\mathcal{B}_i\rvert$	$\lvert\mathcal{C}_i\rvert$	$\lvert X[0, 1, \ldots, i+1]\rvert$
0	1	1	1	3
1	1	2	1	4
2	1	2	2	5
3	2	3	2	7
4	2	4	3	9
5	3	5	4	12
6	4	7	5	16

TABLE I
UPPER BOUNDS OBTAINED USING THE RECURSION.

The last row of the table gives our desired upper bound: $|X[0, 1, \ldots, 7]| \leq 16$.

Sub-case 3.2: $n = 3$. That is, G is a hole of length 7. We show that in this case $|X[0, 1, \ldots, 5]| \leq 8$. This would imply that $|X| \leq |X[0, 1, 3, 5]| \leq |X[0, 1, \ldots, 5]| \leq 8 = 2^n$.

By what we proved in the previous sub-case, we know $|X[0, 1, \ldots, 5]| \leq 9$. We assume then, to reach a contradiction, that $|X[0, 1, \ldots, 5] = 9$. We partition X into three sets as follows:

$$\begin{aligned}
X_{\mathcal{A}} &= \{x \in X \mid x[0, 1] = a_0 b_0\}, \\
X_{\mathcal{B}} &= \{x \in X \mid x[0, 1] = a_0 \bar{b}_0\}, \text{ and} \\
X_{\mathcal{C}} &= \{x \in X \mid x[0, 1] = \bar{a}_0 b_0\}.
\end{aligned}$$

We will prove that if $|X[0, 1, \ldots, 5]| = 9$, then $|X_{\mathcal{A}}[5]| = |X_{\mathcal{B}}[5]| = |X_{\mathcal{C}}[5]| = 2$. That would imply that $|X[5, 6, 0, 1]| = |X_{\mathcal{A}}[5, 6, 0, 1]| + |X_{\mathcal{B}}[5, 6, 0, 1]| + |X_{\mathcal{C}}[5, 6, 0, 1]| \geq 2+2+2 = 6$. By symmetry, the upper bound we proved in the previous sub-case on $|X[0, 1, 2, 3]|$ holds for any sequence of four consecutive positions. Therefore, $|X[5,6,0,1]| \leq 5$, contradicting $|X[5,6,0,1]| \geq 6$.

So how do we prove $|X_{\mathcal{A}}[5]| = |X_{\mathcal{B}}[5]| = |X_{\mathcal{C}}[5]| = 2$? Define, for any $i \in [0, 2n-1]$, $\mathcal{A}_i^R = X_{\mathcal{A}}[0,1,\ldots,i+1]$, $\mathcal{B}_i^R = X_{\mathcal{B}}[0,1,\ldots,i+1]$, and $\mathcal{C}_i^R = X_{\mathcal{C}}[0,1,\ldots,i+1]$. Note that:

$$\begin{aligned}
\mathcal{A}_i^R &= \{x[0,1,\ldots,i+1] \mid x \in X, x[0,1] = a_0 b_0\}, \\
\mathcal{B}_i^R &= \{x[0,1,\ldots,i+1] \mid x \in X, x[0,1] = a_0 \overline{b}_0\}, \text{ and} \\
\mathcal{C}_i^R &= \{x[0,1,\ldots,i+1] \mid x \in X, x[0,1] = \overline{a}_0 b_0\}.
\end{aligned}$$

By reversing the order of indices and then applying the same argument as in the proof of the previous sub-case, we can obtain that:

1) The upper bound on $|\mathcal{A}_i|$ applies to $|\mathcal{A}_i^R|$.
2) The upper bound on $|\mathcal{B}_i|$ applies to $|\mathcal{C}_i^R|$.
3) The upper bound on $|\mathcal{C}_i|$ applies to $|\mathcal{B}_i^R|$.

Since $|X[0,1,\ldots,5]| = 9$ meets its upper bound, then also $|\mathcal{A}_i^R|, |\mathcal{B}_i^R|, |\mathcal{C}_i^R|$ must meet their upper bounds for $i = 0,1,\ldots,4$. We now show separately that $|X_{\mathcal{A}}[5]| = |X_{\mathcal{B}}[5]| = |X_{\mathcal{C}}[5]| = 2$:

1) Using Table I, we have $|\mathcal{A}_2^R| = 1$, while $|\mathcal{A}_4^R| = 2$. Therefore, $|X_{\mathcal{A}}[3]| = 1$ while $|X_{\mathcal{A}}[3,4,5]| = 2$. Since $\{3,5\}$ is a vertex cover of $\{3,4,5\}$, then $|X_{\mathcal{A}}[3,5]| \geq |X_{\mathcal{A}}[3,4,5]| = 2$. However, since $|X_{\mathcal{A}}[3]| = 1$, then $|X_{\mathcal{A}}[5]| = 2$.
2) Using Table I, we have $|\mathcal{B}_3^R| = 3$ while $|\mathcal{B}_4^R| = 4$. Again, this must imply that $|X_{\mathcal{B}}[5]| = 2$.
3) Using Table I, we have $|\mathcal{C}_3^R| = 2$ while $|\mathcal{C}_4^R| = 3$. That is, $|X_{\mathcal{C}}[0,1,2,3,4]| = 2$ while $|X_{\mathcal{C}}[0,1,2,3,4,5]| = 3$. This can happen only if $|X_{\mathcal{C}}[5]| = 2$.

■