Using the Planted Clique Conjecture for Cryptography:

Public-Key Encryption from Planted Clique and Noisy k-LIN Over Expanders

Riddhi Ghosal [*] UCLA
Isaac M. Hair [†] UCLA, UCSB
Aayush Jain [‡] CMU
Amit Sahai [§] UCLA

Abstract

We give a public key encryption scheme that is provably secure against poly-size adversaries, assuming $n^{\log^{\alpha} n}$ hardness of the standard planted clique conjecture, for any $\alpha \in (0, 1)$, and a relatively mild hardness conjecture about noisy k-LIN over expanders that is not known to imply public-key encryption on its own. Both of our conjectures correspond to natural average-case variants of NP-complete problems and have been studied for multiple decades, with unconditional lower bounds supporting them in a variety of restricted models of computation. Our encryption scheme answers an open question in a seminal work by Applebaum, Barak, and Wigderson [STOC'10].

[*] riddhi@cs.ucla.edu
[†] hair@ucsb.edu
[‡] aayushja@andrew.cmu.edu
[§] sahai@cs.ucla.edu

1 Introduction

Public-key encryption (PKE) [DH76, RSA78] is a fundamental primitive enabling secure communication via open and untrusted channels. Despite the immense success of public-key cryptography as a field, and despite intense interest and effort over nearly five decades, we know of surprisingly few sources of hardness that can be nontrivially utlized in building PKE[1].

Furthermore, almost all of these rely on structured hardness rooted in domains such as number theory, algebra, coding theory or lattices (see, for example [DH76, RSA78, Reg05, BFKL94, Ale03]). This is in stark contrast with private-key encryption and one-way functions, where any hard-on-average problem satisfying very mild structural requirements suffices (e.g. planted clique problem, random constraint satisfaction problems, and sparse PCA).

The overarching goal of this research is to devise new ways of nontrivially leveraging other sources of hardness to enable public-key encryption (and beyond). One such untapped source of hardness is the domain of statistical inference and average-case complexity. For decades, these areas have excelled at identifying and studying natural average-case problems in domains such as graph theory, combinatorics and machine learning from both algorithmic and hardness perspectives. There are also systematic means to give evidence of hardness in restricted algorithmic models such as sum-of-squares [Sho85, Nes00, Par00, Las01], statistical-query model [Kea98], low-degree tests framework [BHK$^+$19, Hop18] and many others. Unfortunately, despite an abundant supply of well-studied average-case problems, what we seem to lack are *algorithimic reduction techniques* that can nontrivially leverage these problems to create the kinds of "trapdoor" structures that are needed to create public-key encryption. Can we develop new techniques that would allow us to add to the short list of hardness conjectures that can be nontrivially used to build PKE?

In this work, we humbly add one more source of hardness to this list, namely $n^{\log^\alpha n}$ hardness of the classic planted clique conjecture, for any $\alpha \in (0, 1)$. We do by developing a new quasi-polynomial-time sparsification technique, that can "simulate" a random sparse graph from the point of view of achieving a desired level of expansion. We now elaborate.

Can the planted clique conjecture help us build PKE? Let us examine the classic planted-clique conjecture which roughly states that no efficient algorithm can determine, with constant advantage, whether a random graph contains a planted clique of an appropriate size. This conjecture is the oldest [AKS98, Jer92, Kuč95, Kar76] and arguably the most well-studied conjecture in average-case complexity. Moreover, its worst-case counterpart, the max clique problem, is a classic NP-complete problem. Despite decades of research on the planted clique conjecture [Jer92, Kuč95, ERSY22, JP00, ABW10, ABI$^+$23, MRS20, HK11, AAK$^+$07, ABBG11, BR13, BBH18, BB20, HS24], constructing a PKE nontrivially utilizing this conjecture has remained elusive.

A seminal work that kick-started the search of PKE from such average-case complexity problems was that of Applebaum, Barak and Wigderson [ABW10] which showed several PKE schemes from various statistical inference problems. Of particular relevance to this work is a construction recipe that shows how to design a PKE from any local CSP and a *variant* planted graph conjecture that, roughly speaking, posits the hardness of finding a small subgraph planted in a random sparse k-regular bipartite graph. Indeed, the work of [ABW10] leaves finding a way to leverage the classic planted clique conjecture directly to build PKE as an open question.

We make significant progress towards this question. In particular, our main result builds a PKE scheme from two very well-studied [AKS98, Jer92, Kuč95, Kar76, ERSY22, JP00, ABW10, ABI$^+$23, MRS20,

[1] We say that a hardness conjecture X can be *nontrivially* utilized in building PKE if there exists some other hardness conjecture Y that is not known to imply PKE on its own, and yet we can show that the combination of hardness conjectures X and Y yield PKE.

HK11, AAK+07, ABBG11, BR13, BBH18, BB20, HS24, FK03, FR10, DM15, DGGP14, BHK+19, MW15, KZ14, HWX15, AOW15, BM16, RRS17, FPV15, App16, BSV19, KMOW17] average-case complexity problems, namely: $n^{\log^\alpha n}$ hardness of the standard planted clique conjecture on graphs with n vertices, for any constant $\alpha \in (0,1)$, together with a mild conjecture on the hardness of the noisy k-LIN problem[2] over expanding graphs. Our k-LIN conjecture when interpreted in the special case of equations over $\mathbb{F}_2$ also gives rise to one of the most studied hard distributions for MAX-k-SAT (e.g. [Gri01, Sch08, FKO06, KMOW17]).

Indeed, neither of these well-studied conjectures (with parameters that we use) were previously known to imply PKE. To the best of our knowledge, ours is the first work that constructs a PKE making critical use of the standard planted clique conjecture[3]. At the heart of our work is a new algorithmic technique to leverage this conjecture to create "trapdoor-friendly" structures.

We now introduce our hardness conjectures, and go through a brief discussion of their history.

Conjecture 1: The Planted Clique Problem. Let $\mathcal{G}(n, \frac{1}{2})$ denote the standard Erdős–Rényi distribution[4] over random graphs. Let $\mathcal{G}(n, \frac{1}{2}, n^\delta)$ denote the distribution that first samples a graph from $\mathcal{G}(n, \frac{1}{2})$ and then plants a clique on n^δ randomly selected nodes.

Conjecture 1.1 ($n^{\log^\alpha n}$ hardness[5] of the Planted Clique Conjecture)**.** *There exist constants $0 < \alpha < 1$ and $\delta \in (0, \frac{1}{2})$ such that, for all $n^{O(\log^\alpha n)}$-size algorithms $\mathcal{A}$,*

$$\left| \Pr\left[\mathcal{A}(G) = 1 \mid G \xleftarrow{\$} \mathcal{G}(n, \frac{1}{2})\right] - \Pr\left[\mathcal{A}(G) = 1 \mid G \xleftarrow{\$} \mathcal{G}(n, \frac{1}{2}, n^\delta)\right] \right| \leq \frac{1}{3}.$$

Remark 1.2. For conciseness, we refer to the above conjecture simply as "the planted clique conjecture" in the body of this paper.

Note that we only require *the existence* of some constants; for example, $\alpha = 10^{-3}$ and $\delta = 10^{-3}$ would suffice. Even stronger versions have been considered, for example the "strongish planted clique conjecture" of Manurangsi, Rubinstein, and Schramm [MRS20], which posits $n^{o(\log n)}$ hardness for the same problem.

Planted clique has been rigorously studied in statistical inference and average-case complexity ever since its introduction [AKS98, Jer92, Kuč95, Kar76, ERSY22, JP00, ABW10, ABI+23, MRS20, HK11, AAK+07, ABBG11, BR13, BBH18, BB20, HS24, MW15, KZ14, HWX15]. Many different algorithms have been proposed (e.g. [AKS98, FK03, FR10, DM15, DGGP14]), but none of the currently known algrorithms succeed in determining, with constant advantage, whether there exists a planted clique of size smaller than $o(\sqrt{n})$ in polynomial time. Furthermore, the problem has been very well analyzed in restricted algorithmic models, and is subject to strong lower bounds in these models, including sum-of-squares [BHK+19] and other frameworks [Jer92, CMZ23, Ros08, FGR+17, FK03], providing rigorous evidence for hardness. The recent work of Hirahara and Shimizu also showed a search-to-decision reduction for planted clique [HS24], so our conjecture is equivalent to assuming only the hardness of the search version of the problem.

Conjecture 2: Noisy k-LIN Problem over Expanders. We now describe the noisy k-LIN problem over expander graphs. This is a particular well-studied instantiation of local CSPs over expander graphs where

[2] In cryptographic contexts this conjecture is sometimes called Sparse LPN [Ale03, IKOS08, BSV19, DIJL23], but we will use the terminology of noisy k-LIN. In some references, only the special case of k-LIN over $\mathbb{F}_2$, i.e. k-XOR, is considered.

[3] That is, the planted clique conjecture on an Erdős–Rényi graph with edge probability $\frac{1}{2}$

[4] That is: in a graph with n vertices, include an edge between every pair of vertices independently with probability $\frac{1}{2}$.

[5] Technically, the conjecture is defined to assert hardness against all $n^{O(\log^\alpha n)}$-size algorithms, but note that hardness against $n^{\log^\alpha n}$-size algorithms would guarantee hardness against all $n^{O(\log^\beta n)}$-size algorithms for any constant β satisfying $0 < \beta < \alpha$. So for notational simplicity we slightly abuse notation and use the name "$n^{\log^\alpha n}$-hardness of Planted Clique Conjecture" to mean hardness against all $n^{O(\log^\alpha n)}$-size algorithms, for some $\alpha \in (0,1)$.

the predicate is just the linear function. As in standard planted CSPs over expander graphs, the problem is defined with respect to a graph represented by a collection of sets $(S_1, \ldots, S_m)$. Here, n is the number of variables and $m = m(n)$ is the number of clauses (each clause being a sparse linear equation over a subset of the variables), and each set S_i is of size exactly $k = k(n)$ and supported over $\{1, \ldots n\}$. The set system satisfies a property that is called (γ, k, t)-expansion, which requires that for any collection $\{S_{i \in I}\}$ of size at most t, the union of S_i has at least $\gamma \cdot k \cdot |I|$ many distinct variables, where γ is a constant bigger than $\frac{1}{2}$. See Definition 3.1 for formal details.

In the context of the noisy k-LIN problem, one is asked to distinguish between the joint distributions $(\mathbf{A}, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e})$ and $(\mathbf{A}, \mathbf{b} = \mathbf{r})$, where $\mathbf{A}$ is a sparse matrix in $\mathbb{F}_p^{m \times n}$ produced by taking an indicator matrix $\mathbf{I} \in \{0,1\}^{m \times n}$ representing the sets $\{S_1, \ldots, S_m\}$ and replacing each nonzero entry in $\mathbf{I}$ with a random element of $\mathbb{F}_p^*$. Specifically, for all i, the ith row of $\mathbf{A}$ encodes the set S_i: we have that $\mathbf{A}_{i,j} \neq 0$ if $j \in S_i$ and 0 otherwise. The secret $\mathbf{s} \in \mathbb{F}_p^n$ and $\mathbf{r} \in \mathbb{F}_p^m$ are uniformly chosen random vectors of appropriate dimensions, and every entry of $\mathbf{e}$ is sampled independently from a Bernoulli distribution with probability η. Backed by systematic lower bounds over many years [Gri01, FKO06, Sch08, BM16, RRS17, FPV15, App16, BSV19, KMOW17], we formalize the following widely believed conjecture:

Conjecture 1.3 (Noisy k-LIN Conjecture)**.** *There exists a constant $\gamma \in (\frac{1}{2}, 1)$ such that the following holds. Let α, ς be any constants with $\alpha \in (0,1)$ and $\varsigma > 0$. Then, for n large enough, prime $p = n^{\Omega(1)}$, $k = \Omega(\log(n))$, we have the following: Let $\mathbf{I} \in \{0,1\}^{m \times n}$ correspond to any $(\gamma, k, 2^{(\log n)^\alpha})$-expanding set system, and let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be sampled by replacing each nonzero entry in $\mathbf{I}$ with a random element in $\mathbb{F}_p^*$. It holds that*

$$(\mathbf{A}, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}),$$

is computationally indistinguishable from

$$(\mathbf{A}, \mathbf{b} = \mathbf{r}).$$

We only require indistinguishability with respect to a small constant distinguishing advantage, specifically $\frac{1}{16}$. Above, $\mathbf{s} \leftarrow \mathbb{F}_p^n$ and $\mathbf{r} \leftarrow \mathbb{F}_p^m$ are uniformly randomly chosen and $\mathbf{e} \in \mathbb{F}_p^m$ sampled as:

$$\mathbf{e}_i = \begin{cases} \textit{Random element of } \mathbb{F}_p, \textit{ with probability } \eta = (\log n)^{-\varsigma}. \\ 0, \textit{ otherwise.} \end{cases}$$

Remark 1.4. In fact, we can use the search version of the above conjecture, and the matrix $\mathbf{A}$ for our PKE scheme will satisfy additional desirable properties[6]. See Section 4 for details.

Intuitively, the conjecture becomes more "secure" for larger values of k, and if the expansion threshold t and the error probability η are large. In particular, it is desirable that t is at least $\Omega(\log n)$, and in our case, it is actually much larger, namely $t = 2^{\log^\alpha n}$. Keeping this value of t in mind, we can mentally classify the problem in two regimes when it comes to using noisy k-LIN for designing PKE, based on the noise probability η.

- Noise probability $\eta < \frac{O(\log n)}{t}$: In this regime, Applebaum et. al. [ABW10] showed how to design a PKE only leveraging the noisy k-LIN problem for any constant $k \geq 3$. Here, the noise probability increases as the number of clauses m gets closer to $n^{k/2}$, but this causes the expansion parameter t to decrease as well, and in all cases, the noise probability is $\eta < \frac{O(\log n)}{t}$. Recall that noisy k-LIN is broken when given approximately $m = n^{k/2}$ clauses [AOW15, BM16, FPV15], in which case no nontrivial expansion is possible. Any small non-expanding collection of sets, in some sense, can serve as a secret key for the scheme.

[6]Most notably, for *every* non-empty set of $\{S_{i \in I}\}$, the union of S_i has at least $|I|^{1-\varepsilon}$ many distinct variables, where ε here is an arbitrarily small positive constant chosen at time of setup.

Indeed, the k-LIN problem in this regime can be efficiently attacked if given unbounded preprocessing on the graph[7]. The ABW scheme from noisy k-LIN problem alone can be broken given a short advice on the matrix. This is also related to the so-called "Alekhnovich" barrier in cryptography for design of encryption schemes from (variants of) the Learning Parity with Noise problem [Ale03].

- In the other regime when $\eta > \frac{\omega(\log n)}{t}$, which is *our* regime, we do not know of any PKE scheme from the noisy k-LIN conjecture alone. This regime differs greatly from the previous one: No attacks are known on noisy k-LIN in the second regime, even with unbounded preprocessing on the expander graph. This is also connected to the notion of efficient non-deterministic refutation for certain CSPs [FKO06] which only applies to the first regime, and where there has been no progress in the second regime.

  In our setting, we mandate expansion on all sets of size at most $t = 2^{(\log n)^\alpha}$, and the error rate is $\eta = (\log n)^{-O(1)} = (\log t)^{-O(1)} \gg \frac{\omega(\log n)}{t}$. At the same time, our construction satisfies the following property with regard to subsets of the clauses: For any set S of x variables, at most $x^{1+\varepsilon} \ll x^{k/2}$ clauses exist that use only variables contained in S, where ε is an arbitrarily small positive constant chosen at time of setup. In particular, this means that the total number of clauses is at most $n^{1+\varepsilon} \ll n^{k/2}$.

In Section 7, we import some existing results that show unconditional lower bounds supporting Conjecture 1.3 in the following restricted models of computation: (1) local tests, (2) linear tests, (3) low degree polynomials, (4) sum of squares algorithms, and (5) AC^0 circuits.

Remark 1.5. An important remark is that there are contrived PAC learning based counterexamples to noisy k-LIN in settings where the entire expander graph can be computed by a circuit in a shallow complexity class such as $\mathsf{AC}^0[\oplus]$ [OST19]. This is due the fact that the resulting k-LIN is approximable by a $\mathsf{AC}^0[\oplus]$ circuit that can be PAC-learned. The work also showed existence of such expander graphs relying on Nisan-Wigderson pseudorandom generator applied to some polynomial sized seed [Nis91].

We note that in our case the expander for k-LIN is sampled from a natural distribution induced by a random $G(n, \frac{1}{2})$, and furthermore multiple steps in our construction will introduce independent randomness for the neighbors of each edge in the expander graph. So the neighbor function simply has too much entropy to be computed by any nontrivial $\mathsf{AC}^0[\oplus]$ circuit. In fact the current counterexamples [OST19] only apply to the setting where the number of samples is superpolynomial with respect to the length of the secret, which is far outside of our setting. Finally, in the setting where $p > 2$, and the coefficients for the k-LIN constraints are chosen independently and randomly from $\mathbb{F}_p^*$ (as in our construction), there are no known counterexamples even where the number of samples can be superpolynomial.

Our Main Result. We establish the following theorem:

Theorem 1.6 (Informal)**.** *If Conjecture 1.1 and Conjecture 1.3 both hold, then there exists a semantically secure public key encryption scheme.*

1.1 Related Work

Relationship with [ABW10]. Applebaum, Barak and Wigderson designed several PKE schemes from related assumptions. They have a construction from random noisy k-LIN for any $k \geq 3$ that is in the regime $\eta < O(\frac{\log n}{t})$. Most relevant to our work is their PKE scheme in the regime when $\eta > \frac{\omega(\log n)}{t}$, which is

[7] It suffices to use a t sparse vector $\mathbf{x}$ such that $\mathbf{xA} = 0$ to distinguish the samples from random.

based on nosiy k-LIN[8] for any k and a hidden subgraph assumption over random sparse k regular bipartite graphs. Recently, this assumption was proven to be secure against low degree polynomial algorithms, thanks to the work of Bogdanov, Kothari and Rosen [BKR23], however we don't know yet how to reduce the standard planted clique problem to their assumption. Applebaum, Barak, and Wigderson do prove an interesting theorem (see Theorem B.13 in the appendix of [ABW10]) that connects the planted clique problem to a variant of their bipartite graph assumption via a simple reduction. We now discuss two of the main reasons why Theorem B.13 in [ABW10] leads to strictly (much) weaker conclusions than ours.

- Firstly, the theorem only works with planted clique over significantly sparser graphs, specifically those with edge probability $2^{-\ell}$ where ℓ is roughly of the order $\log^{(1-\varepsilon)} n$ for some small constant $\varepsilon > 0$. The standard planted clique problem, on the other hand, sets the edge probability to $\frac{1}{2}$, and is the most widely studied form of the problem algorithmically.
- Also, their reduction for very low edge probability works only when considering a weakening of their bipartite subgraph assumption: namely where the associated graph satisfies a weaker property known as unique-neighbor expansion. While it turns out that unique neighbor expansion is enough to establish a few lower bounds for noisy k-LIN, it is much weaker than the notion of (γ, k, t)-expanding set-systems for $\gamma > \frac{1}{2}$, which is what our result uses.

Other Schemes. There are only a few other works that give PKE from graph-based assumptions. Bogdanov, Kothari and Rosen [BKR23] give a variant of one scheme by Applebaum, Barak, and Wigderson [ABW10]. In particular, they replace the planted dense subgraph assumption with another non-standard assumption about planted even covers, for which they show hardness against low degree polynomial algorithms.

Abram, Beimel, Ishai, Kushilevitz, and Narayanan explore using a more aggressive assumption to build PKE with logarithmic-size messages [ABI+23]. Hudoba proposed a scheme inspired by the problem of finding a planted clique in a modified random graph [Hud16], but this scheme was broken by a polynomial-time key recovery attack [CJ24].

Goldreich PRGs. Inspired by constraint satisfaction over expander graphs, in 2000 Goldreich proposed low-complexity local candidate one-way functions [Gol00] that were later imagined as candidate low-complexity cryptographic pseudorandom generators [CM01, IKOS08]. The complexity theory and cryptography communities have jointly developed a rich body of literature on the cryptanalysis and theory of constant-locality Boolean PRGs [Gol00, CM01, MST03, CEMT09, BQ09, ABW10, ABR12, App12, OW14, AL16, KMOW17, CDM+18, AK19]. Ever since its introduction Goldreich's PRGs have been conjectured [Gol00, ABW10, ABR12, AL16] to be secure on arbitrary expander graphs. The cryptanalysis of such PRG candidates bears uncanny resemblance to the algorithmic study of the noisy k-LIN conjecture (in the regime $p = 2$) over expander graphs due to similarity in structure of the two problems. A typical way to instantiate Goldreich PRG predicates is to consider a function that is a sparse linear function which is XORed with an appropriately chosen non-linear predicate to mimic the noise from the noisy k-XOR problem [AL16]. A 20 year study of Goldreich PRGs in cryptography once again solidifies our confidence in the noisy k-LIN conjecture.

1.2 Open Problems

Of course, the most natural open problem left open by our work is to build PKE assuming *only* some reasonable hardness of the planted clique conjecture. However, we believe that the most pressing open

[8] This PKE is actually quite general, and could work with other local CSPs as well.

problems are:

- Can we *further populate the list* of explicit average-case hardness conjectures that can be nontrivially used in building PKE?
- Can we further populate the list of explicit hardness conjectures that suffice on their own to yield PKE (without needing any "helper" hardness conjectures)?
- What level of sparse error is needed in the context of "noisy linear algebra"-type hardness conjectures, like k-LIN, to yield (plausibly sub-exponentially secure) PKE without any additional hardness conjectures?
- More broadly, *what are the minimal levels of mathematical structure* that suffice to yield PKE?

2 Technical Overview

The technical core of our new PKE construction is the instantiation of an *expanding set system* from the planted clique conjecture. Concretely, the set system consists of a set of "variables" and a set of "clauses" over those variables. This set system is used to sample ciphertexts that are instances of k-LIN on expanders.

In this overview, we will focus on discussing how we start with a graph G that is either sampled from $\mathcal{G}(n, \frac{1}{2})$ or from $\mathcal{G}(n, \frac{1}{2}, n^\delta)$, and then make a sequence of transformations to it that finally yields this desired set system.

Desiderata. We begin by stating the properties that we need in order to simultaneously achieve security and decryption correctness:

Goal 1. If we start with a graph G sampled from $\mathcal{G}(n, \frac{1}{2})$, then we want the sets of variables and clauses to satisfy strong expansion properties. In particular, every collection I of up to $t = 2^{\log^\alpha n}$ clauses must include at least $\gamma \cdot k \cdot |I|$ variables, where $\gamma \in (\frac{1}{2}, 1)$ is any fixed constant chosen at the time of construction.

Goal 2. If we start with a graph G sampled from $\mathcal{G}(n, \frac{1}{2}, n^\delta)$, then there must exist a set of $x = \mathsf{polylog}(n)$ clauses that include less than x variables.

Specifically, Goal 1 above is needed for security of our PKE scheme; Goal 2 allows us to have efficient decryption for our PKE scheme (see Section 5 for details).

An initial naive approach. The most obvious idea is to use the incidence graph between edges and vertices to define our sets of variables and clauses. More specifically, we could use (i) the set of vertices to represent the variables, along with (ii) the set of edges to represent each clause, where each edge represents a clause depending on the variables for the edge's endpoints. Indeed such a construction would only allow us to achieve an arity of $k = 2$ if applied to a standard graph[9], which doesn't work for us[10].

A natural generalization of the above idea is to look at subgraphs of G with larger size. In particular, to obtain a set of clauses with arity $k > 2$, we can start by assigning a variable to each $(k-1)$-clique in G. Each clause would then correspond to a k-clique, with the variables being those for each $(k-1)$-clique inside the k-clique.

[9] This corresponds to noisy 2-XOR, which is easy to solve.

[10] In fact, one of the appendices of Applebaum et. al. [ABW10] used this naive vertex-edge incidence graph in a reduction.

If we think of constant k, then the number of variables and clauses is polynomial in n; however, we do not achieve expansion as per Goal 1. Indeed, in $\mathcal{G}(n, \frac{1}{2})$, we expect there to be cliques of size up to $\Theta(\log n)$, and for each clique of size x in our graph, we will have roughly $\binom{x}{k}$ clauses on only $\binom{x}{k-1}$ variables, which is (very) contracting. In fact, it turns out that if one picks a random set of x vertices, for *any* $x \leq n$, we expect to find $\Theta(\binom{x}{k})$ clauses that contain $\Theta(\binom{x}{k-1})$ variables. The problem here is that the density of small regions in the resulting variable-clause bipartite graph when starting from $\mathcal{G}(n, \frac{1}{2})$ is not substantially different than the density of such regions when starting from $\mathcal{G}(n, \frac{1}{2}, n^{\delta})$. We therefore ask the following question:

Can we modify G in a useful way to amplify the difference in local density between the planted clique and the rest of the graph?

Sparsification. Intuitively, our aim is to amplify the density disparity between the portion of the graph inside and outside the planted clique, thereby enabling us to achieve Goals 1 and 2. We do so by means of a *vertex*-deletion based sparsification technique. We transform G into a larger and sparser graph H, and we will now identify variables with $(k-1)$-cliques in H, and clauses with k-cliques in H. We do so as follows:

1. Create a set of labels, each of which corresponds to an ℓ-sized subset of vertices in G.
2. Include each label as a vertex of H with probability $p = n^{-c\ell}$ for some small positive constant $c < \delta$.
3. Connect two nodes in H with an edge iff their respective labels correspond to two subsets whose union forms a clique (of size at most 2ℓ) in G.

This method is called the "randomized graph product" and has been used previously to study the hardness of various graph problems [BS92, MRS20, HS24]. Perhaps the most interesting aspect of this randomized graph product is Step 2, where *almost all* of the labels are discarded. To gain intuition as to why this is helpful to us, let's denote by $\hat{H}$ the graph obtained via the product above, but without Step 2 – namely, we create a vertex in $\hat{H}$ for every label created in Step 1. Now, observe that corresponding to the clique of size $\Theta(\log n)$ in G, we can (efficiently) find a clique in $\hat{H}$ of size roughly $\binom{\log n}{\ell}$. The fact that this "huge" clique in $\hat{H}$ is of size $\omega(\log n)$ turns out to be highly problematic for us: this is because the "huge" clique will induce a very high level of compression. Because of the deletion (Step 2), our sparsified graph H does *not* have such a large clique.

Going super-polynomial in n — the need for large graph H. We pause now to make a crucial observation. Note that if we pick $\ell = O(1)$, then the edge probability in H is still a constant. This implies that even after sparsification, for all x, there will exist many subgraphs with x vertices and $\Theta(x^2)$ edges. In other words, small structures in H are still quite dense, and the difference in density as compared to the planted clique is only a constant factor.

To overcome this hurdle, it is imperative that we pick ℓ to be superconstant. In particular, we choose $\ell = \log^{\alpha} n$, for any constant $\alpha \in (0, \frac{1}{2})$. While this does yield an H whose size in slightly superpolynomial in the size of G, we now have a very useful property: H has an edge probability of $(\frac{1}{2})^{(\log n)^{\Theta(1)}}$. In fact, it turns out that with these parameters, there exists a cutoff value $t = 2^{(\log n)^{\Theta(1)}}$ such that any subgraph of H with $x \leq t$ vertices will have at most only $x \cdot \mathsf{polylog}(n)$ edges! At the same time, if G had a planted clique, then H still would contain a large planted clique, in which subgraphs of the same size will have $\Theta(x^2)$ edges.

Since the size of H is superpolynomial in n, we need the planted clique conjecture to hold against adversaries running in time
$n^{O(\log^{\alpha} n)}$ as stated in Conjecture 1.1.

Constructing the Incidence Graph. Our remaining goal is to create a mapping from H to variables and clauses. Drawing on the example from the start of this section, we could have a variable for each vertex and a clause for each edge. Obviously, as observed before, such a construction does not work because the arity is $k = 2$, but it also fails for a more interesting reason: large star graphs[11] always exist in H, and they correspond to sets of clauses with poor expansion properties. Since every edge in the star graph shares a vertex, this means that each clause corresponding to an edge of the star graph has only one unique variable, and the other variable is shared by all of the other clauses. So the best expansion factor we can hope for is $\gamma \leq \frac{1}{2}$, no matter how aggressive our sparsification is. This violates Goal 1 where we need $\gamma > \frac{1}{2}$.

Looking at k-cliques. To get some intuition, let us extend the above idea and consider having one variable for each edge in H, and a clause for each *triangle* in H, where the three variables for each clause are given by the edges inside each triangle. We can again try to find a compressing subset by forcing all triangles to share as many edges as possible. Observe that, because any sufficiently small set of vertices has a near linear number of edges, we can only force all triangles to share a single edge. Nearly all of the other edges will be unique to their specific triangle. In this way, we get an upper bound on the expansion factor of $\gamma \leq \frac{2}{3}$. Extending this to having one variable for each $(k-1)$-clique and a clause of arity k for each k-clique, the same worst-case construction gives an upper bound of $\gamma \leq \frac{k-1}{k}$. In fact, this this naive upper bound is close to the lower bound in the following sense. We show that for any constant $\varepsilon > 0$ there is a constant integer k such that,

For any small enough set I of clauses, nearly all clauses in I have that a $(1-\varepsilon)$ fraction of their variables appear in at most $\mathsf{polylog}(n)$ *many other clauses of I.*

On the other hand, when G contains a planted clique, we get a tiny subset T of clauses where every variable is shared by a much higher number of clauses in T. Thus the planted clique gives a trapdoor "highly compressing" set of clauses, but when G doesn't contain a planted clique, any small set of clauses can only be at worst mildly compressing. We refer readers to Lemma 6.7 for the formal claim and other details.

The above property takes us very close to our goal. However so far we cannot rule out the case that, for some small set of clauses I, the clauses together only touch $|I|/\mathsf{polylog}(n)$ many variables.

Augmentation with Random Vectors. To ensure the stronger notion of expansion that we desire, we augment the clauses with random sparse vectors. In more detail, instead of each $(k-1)$-clique corresponding to just one variable, we will introduce a new constant parameter ζ, and have each $(\zeta-1)$-clique correspond to $\log^C n$ many variables for some sufficiently large constant C. For every ζ-clique, we will form a clause of arity $k = \Theta(\log |H|)$. To get the set of variables for each clause, we randomly subsample k/ζ variables from the set of $\log^C n$ variables corresponding to each $(\zeta-1)$-size sub-clique of the clause's ζ-clique.

We show in Lemma 5.4 that a set of random sparse vectors has exemplary small-set-expansion properties. Recall that, without augmentation, any small set of clauses is only mildly compressing – nearly all of the clauses have a $(1-\varepsilon)$ fraction of their variables appearing in at most $\mathrm{polylog}(n)$ other clauses. In Lemma 5.5, we layer these two observations to show that augmentation gives the necessary expansion when G is sampled from $\mathcal{G}(n, \frac{1}{2})$.

At the same time, when G is sampled from $\mathcal{G}(n, \frac{1}{2}, n^\delta)$, our hidden "highly compressing" set of clauses is so dense that it remains dense even after this augmentation step, thereby enabling decryption (see proof of Lemma 5.3).

[11] Star graphs of size $k+1$ are complete bipartite graphs whose node sets have size 1 and k.

Notational Guidance. What we called n here in this overview, we call $\bar{n}$ in the technical sections. We will use n to refer to the total number of variables resulting from our final transformation as sketched above.

3 Preliminaries

Notations. A vector is said to be k-sparse if it has exactly k many non-zero entries. Similarly a matrix is called k-sparse if each of its rows forms a k-sparse vector. Let $\mathsf{hw}(\mathbf{v})$ be the Hamming weight of $\mathbf{v}$, so $\mathsf{hw}(\mathbf{v}) = k$ iff $\mathbf{v}$ is a k-sparse vector. Let $\bigvee_{\mathbf{v}\in S} \mathbf{v}$ denote the bitwise-OR of all vectors $\mathbf{v} \in S$. We use $[x]$ to denote the set $\{1, \dots x\}$. We use $N_G(v)$ to denote the open neighborhood of vertex v with respect to the graph G, i.e. all nodes (barring v) in G that are connected to v with an edge. Throughout, the reader may assume that quantities are rounded to integer values as needed. All logarithms are taken base 2.

Definition 3.1 ((γ, k, t)-Expanding Set System)**.** *Let $\mathcal{S}$ be a collection of sets. We say that $\mathcal{S}$ is a (γ, k, t)-expanding set system iff:*

1. *$\forall S \in \mathcal{S}, |S| = k$, and*
2. *$\forall \mathcal{T} \subseteq \mathcal{S}$ such that $|\mathcal{T}| \leq t$:*

$$\left| \cup_{S\in\mathcal{T}} S \right| \geq \gamma \cdot k|\mathcal{T}|.$$

Definition 3.2 ((γ, k, t)-Expanding Matrix)**.** *We say that $\mathbf{S} \in \{0,1\}^{m\times n}$ corresponds to a (γ, k, t)-expanding set system iff:*

1. *For all rows $\mathbf{S}_i$ of $\mathbf{S}$, we have that $\mathsf{hw}(\mathbf{S}_i) = k$, and*
2. *$\forall T \subseteq \{1, \dots m\}$ such that $|T| \leq t$:*

$$\mathsf{hw}\left(\bigvee_{i\in T} \mathbf{S}_i\right) \geq \gamma \cdot k|T|.$$

Weak Public Key Encryption.

Definition 3.3 (Adapted from Definition 4.1 in [ABW10])**.** *An $(\psi(n), \phi(n))$-secure public-key bit encryption scheme is a triple (Gen, Enc, Dec) of probabilistic polynomial time algorithms such that*

- *Algorithm Gen, on input 1^n, produces a pair (pk, sk).*
- *($(1-\psi)$-correctness) For a random bit $b \leftarrow \{0,1\}$,*

$$\Pr\left[Dec_{sk}(Enc_{pk}(b)) = b\right] > 1 - \psi(n),$$

  where $(pk, sk) \xleftarrow{\$} Gen(1^n)$ and the probability is over the randomness of Gen, Enc, Dec, and the choice of b.

- *(ϕ-privacy) The distributions $(pk, Enc_{pk}(0))$ and $(pk, Enc_{pk}(1))$ are $\phi(n)$-indistinguishable, where $(pk, sk) \xleftarrow{\$} Gen(1^n)$.*

If $\psi(n)$ and $\phi(n)$ satisfy $\psi < (1 - \sqrt{\phi})/2$ and ϕ is a constant, we say that the scheme is a weak PKE.

Lemma 3.4 (See Theorem 6 in [HR05])**.** *Every weak PKE can be transformed into a semantically secure PKE that supports arbitrarily (polynomially) long messages.*

Randomized Graph Product. Our version of the randomized graph product is a slight variation of the original definition introduced in [BS92, MRS20]. This will be a crucial tool for the construction of the PKE.

Definition 3.5 (Randomized Graph Product)**.** *Given a graph $G = (V_G, E_G)$, integer ℓ, and probability p, we define the randomized graph product $H := (V_H, E_H) \xleftarrow{\$} \mathrm{RP}_p^\ell(G)$ as follows:*

1. *Let $V' = (v_1, v_2, \ldots, v_{\binom{V_G}{\ell}})$ denote an ordered sequence of labels. Every label $v \in V'$ corresponds to a subset $U_v \subset V_G$ of size ℓ ordered in some fixed "canonical" sense.*
2. *Let V_H be the node set of H created in the following manner: for every $v \in V'$, add v to V_H independently with probability p.*
3. *For each $u, v \in V_H$ with $u \neq v$, let $(u, v) \in E_H$ iff $U_u \cup U_v$ induces a clique in G.*

4 Our Hardness Conjectures

In this section we redefine the conjectures and all of their variants formally as we use them in the paper.

4.1 The Planted Clique Conjecture

Conjecture 4.1 ($n^{\log^\alpha n}$ hardness of the Planted Clique Conjecture)**.** *There exist constants $0 < \alpha < 1$ and $\delta \in (0, \frac{1}{2})$ such that, for all $n^{O((\log n)^\alpha)}$-size algorithms $\mathcal{A}$,*

$$\left| \Pr\left[\mathcal{A}(G) = 1 | G \xleftarrow{\$} \mathcal{G}(n, \frac{1}{2})\right] - \Pr\left[\mathcal{A}(G) = 1 | G \xleftarrow{\$} \mathcal{G}(n, \frac{1}{2}, n^\delta)\right] \right| \leq \frac{1}{3}.$$

Remark 4.2. Recall that, for conciseness, we simply refer to the above as "the planted clique conjecture."

As discussed earlier, this conjecture is based on a long line of work in the average-case complexity community [AKS98, Jer92, Kuč95, Kar76, ERSY22, JP00, ABW10, ABI+23, MRS20, HK11, AAK+07, ABBG11, BR13, BBH18, BB20, HS24, MW15, KZ14, HWX15].

4.2 Noisy k-LIN Conjecture

We first restate the standard noisy k-LIN conjecture from the introduction. Recall that this conjecture is substantiated by a long series of works giving lower bounds on the k-LIN problem [Gri01, FKO06, Sch08, BM16, RRS17, FPV15, App16, BSV19, KMOW17].

Conjecture 4.3 (Noisy k-LIN Conjecture)**.** *There exists a constant $\gamma \in (\frac{1}{2}, 1)$ such that the following holds. Let α, ς be any constants with $\alpha \in (0, 1)$ and $\varsigma > 0$. Then, for n large enough, prime $p = n^{\Omega(1)}$, $k = \Omega(\log(n))$, we have the following: For all $(\gamma, k, 2^{(\log n)^\alpha})$-expanding matrices $\overline{\mathbf{S}} \in \{0,1\}^{m \times n}$ and for all polynomial sized adversaries $\mathcal{A}$,*

$$\left| \Pr_{\mathbf{A},\mathbf{s},\mathbf{e}}[\mathcal{A}(\mathbf{A}, \mathbf{b}) = 1 \mid \mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}] - \Pr_{\mathbf{A},\mathbf{b}}[\mathcal{A}(\mathbf{A}, \mathbf{b}) = 1 \mid \mathbf{b} \xleftarrow{\$} \mathbb{F}_p^m] \right| \leq \frac{1}{4}.$$

In the above, $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ is the matrix produce by replacing each nonzero entry in $\overline{\mathbf{S}}$ with a random element in $\mathbb{F}_p^*$, and $\mathbf{e}$ is sampled as:

$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\varsigma}. \\ 0, \text{ otherwise.} \end{cases}$$

Below we define the analogous search version of the noisy k-XOR conjecture which we use to construct another PKE in Section 8.

Conjecture 4.4 (Search Noisy k-LIN Conjecture)**.** *There exists a constant $\gamma \in (\frac{1}{2}, 1)$ such that the following holds. Let α, ς be any constants with $\alpha \in (0,1)$ and $\varsigma > 0$. Then, for n large enough, prime $p = n^{\Omega(1)}$, $k = \Omega(\log(n))$, we have the following: For all $(\gamma, k, 2^{(\log n)^\alpha})$-expanding matrices $\overline{\mathbf{A}} \in \{0,1\}^{m \times n}$ and for all polynomial sized adversaries $\mathcal{A}$,*

$$\Pr_{\mathbf{A},\mathbf{s},\mathbf{e}}[\mathcal{A}(\mathbf{A}, \mathbf{b}) = \mathbf{s} \mid \mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}] \leq \frac{1}{4}.$$

In the above, $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ is the matrix produce by replacing each nonzero entry in $\overline{\mathbf{A}}$ with a random element in $\mathbb{F}_p^*$, and $\mathbf{e}$ is sampled as:

$$\mathbf{e}_i = \begin{cases} \textit{Random element of } \mathbb{F}_p^*, \textit{ with probability } \eta = (\log n)^{-\varsigma}. \\ 0, \textit{ otherwise.} \end{cases}$$

In other words, it should be hard to recover $\mathbf{s}$ given $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e})$. In Section 8 we present a PKE construction from Conjecture 1.1 and Conjecture 4.4. While a reduction from search to decision noisy k-LIN exists for random expander matrices $\mathbf{A}$ [App16, BSV19], the reduction suffers from a polynomial loss in the sample complexity, i.e. the number of clauses. Thus our construction in Section 8 relies on a potentially weaker[12] assumption that the one in Section 5 for the following two reasons: (1) the semantic security reduction to the search version does not suffer from a polynomial loss, and (2) it is unclear if the search-decision reduction mentioned above extends to arbitrary expander matrices $\mathbf{A}$.

Non-Compression. In fact, the matrix $\mathbf{A}$ from both of our PKE schemes satisfies an additional property: The matrix $\mathbf{S} \in \{0,1\}^{m \times n}$ indicating the nonzero entries in $\mathbf{A}$ is a $(\gamma, \Theta(\log n))$-non-compressing matrix, for any constant $0 < \gamma < 1$ fixed at the beginning of key generation. We define this property below.

Definition 4.5 ((γ, k)-Non-Compressing Matrix)**.** *We say that $\mathbf{S} \in \mathbb{F}_2^{m \times n}$ is a (γ, k)-non-compressing matrix iff:*

1. *For all rows $\mathbf{S}_i$ of $\mathbf{S}$, we have that $\mathsf{hw}(\mathbf{S}_i) = k$, and*
2. *$\forall T \subseteq \{1, \ldots m\}$:*

$$\mathsf{hw}\left(\bigvee_{i \in T} \mathbf{S}_i\right) \geq |T|^\gamma.$$

In other words, for *every arbitrarily large* set of rows, the sum of hamming weights for those rows should not be much greater than the hamming weight when we bitwise-OR all of the rows together. This property is desirable in the context of Goldreich's candidate one-way function [Gol11, App12, AL16], which is highly related to noisy k-LIN. (In this context "non-compression" is known as "low stretch.")

5 Public key encryption from Planted Clique and Noisy k-LIN

In this section, we give a PKE scheme from the *decisional* noisy k-LIN conjecture. For a description of how to use the search version, see Section 8.

[12] The assumption is weaker in the sense that the search noisy k-LIN is at least as hard as decision noisy k-LIN.

Suggested Parameters. We suggest the following choices of parameters. The constants are solely optimized for exposition, and we believe they can be improved significantly while still maintaining the integrity of the scheme. Let $\alpha, \delta \in (0, \frac{1}{2})$ be constants such that Conjecture 1.1 holds, and let $\gamma \in (\frac{1}{2}, 1)$ be a constant such that Conjecture 4.3 holds. (If Conjecture 1.1 holds for some $\alpha \geq \frac{1}{2}$, this automatically implies that the conjecture holds for all $\alpha \in (0, \frac{1}{2})$.) Assuming the security parameter is λ, we will set the following parameters:

- p is any prime.
- Size of initial Erdos-Renyi random graph: $\bar{n} = 2^{(\log \lambda)^{1/(1+\alpha)}}$.
- Planted clique size: $\bar{k} = \bar{n}^{\delta}$.
- Graph product parameter for RP: $\ell = (\log \bar{n})^{\alpha}$.
- Node deletion probability for RP: $q = \bar{n}^{-(\delta \ell)/2}$.
- Clique size (constant integer) for constructing k-LIN matrix: $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$.
- Exponent (constant) for block size in k-LIN matrix: $\beta = \zeta \cdot (\zeta(1-\alpha) + 2)$.
- Exponent (constant) for Bernoulli noise for k-LIN samples: $\rho = 2 + \beta\zeta$.

Remark 5.1. The final three parameters are balanced so that, if our final k-LIN matrix $\mathbf{A} \in \mathbb{F}_2^{m \times n}$ is constructed from an Erdős–Rényi graph without a planted clique, we achieve arbitrarily high expansion on all sets of at most $t = 2^{(\log n)^{\alpha}}$ many rows, while maintaining an error rate of $(\log t)^{-O(1)}$. At the same time, we achieve arbitrarily high non-compression over the entire matrix.

Notations. Let $H \xleftarrow{\$} \mathrm{RP}_q^{\ell}(G)$. Then, by construction, every node $v \in V_H$ corresponds to a set $U_v \subset V_G$ of size ℓ in some "canonical" order. Let $c_{H,\zeta}$ denote the total number of ζ-cliques in a graph H. Let $\mathbf{A}_i$ denote the ith row of a matrix $\mathbf{A}$. Allow $\|$ to denote the concatenation of two vectors in the standard sense. We abuse notation and allow $(G, K_G) \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2}, \bar{k})$ to denote a graph G drawn from the planted Erdős–Rényi distribution along with the set of vertices K_G for the planted clique.

Encoding Process. First we describe a randomized encoding of any graph H with n nodes into a sparse matrix $\mathbf{S} \in \{0,1\}^{m \times n}$, where $m = c_{H,\zeta}$ and $n = c_{H,\zeta-1} \cdot \log^{\beta} m$. We use this algorithm in the key generation step of the PKE.

$\star$ $\mathbf{A} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$:

1. Let $m = c_{H,\zeta}$. Enumerate over all ζ-cliques of H in a fixed canonical order and create labels $(R_1, R_2, \ldots, R_m)$.
2. Let $n' = c_{H,\zeta-1}$. Enumerate over all $(\zeta-1)$-cliques of H in a fixed canonical order and create labels $(C_1, C_2, \ldots, C_{n'})$.
3. For all $i \in [m], j \in [n']$ do the following:
   - Set an indicator variable $I_{i,j} = 1$ if the $(\zeta-1)$-clique labeled C_j is a subgraph of the ζ-clique labeled R_i, and set $I_{i,j} = 0$ otherwise.
4. For all $i \in [m], j \in [n']$ do the following:
   - Set $\mathbf{b}_{i,j} \in \{0,1\}^{(\log m)^{\beta}}$ to be a random $\frac{\log m}{\zeta}$-sparse vector if $I_{i,j} = 1$, and set $\mathbf{b}_{ij} =$

$\mathbf{0}^{\log^\beta m}$ otherwise. We refer to each $\mathbf{b}_{i,j}$ as a *block*.

5. For all $i \in [m]$, set $\mathbf{S}_i := (\mathbf{b}_{i,1} \| \mathbf{b}_{i,2} \| \cdots \| \mathbf{b}_{i,n'})$.
6. Output $\mathbf{S}$.

Key Generation.

$\star$ $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$:

1. $(G, K_G) \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2}, \bar{k})$.
2. $H \xleftarrow{\$} \mathrm{RP}^\ell_q(G)$.
3. Define $K_H = \{v \in V_H : U_v \subset K_G\}$.
4. $\overline{\mathbf{A}} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{S}$ have m rows and n columns. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\overline{\mathbf{A}}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} := \mathbf{A}$.
6. $\mathsf{sk} := (H, K_H)$.
7. Output $(\mathsf{pk}, \mathsf{sk})$.

Encryption.

$\star$ $\mathsf{ct} \leftarrow \mathsf{Enc}(1^\lambda, \mathsf{pk}, b \in \{0,1\})$:

1. Parse pk as $\mathbf{A}$. Let $\mathbf{A}$ have m rows and n columns.
2. Sample $\mathbf{e}$ as:

$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\varsigma}. \\ 0, \text{ otherwise.} \end{cases}$$

3. Sample $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$.
4. If $b = 0$, then $\mathsf{ct} \leftarrow \mathbf{A}\mathbf{s} + \mathbf{e}$.
5. If $b = 1$, then $\mathsf{ct} \xleftarrow{\$} \mathbb{F}_2^m$.
6. Output ct.

Decryption.

$\star$ $b' \leftarrow \mathsf{Dec}(1^\lambda, \mathsf{pk}, \mathsf{sk}, \mathsf{ct})$:

1. Parse pk as $\mathbf{A}$. Let $\mathbf{A}$ have m rows and n columns.
2. Parse sk as (H, K_H).
3. Parse ct as $\mathbf{b}$.
4. Let η be any (superconstant) integer satisfying $(\log n)^{\rho-1} \leq \binom{\eta}{\zeta} \leq 2(\log n)^{\rho-1}$.
5. Uniformly randomly select a subset $T \subset K_H$ among all subsets of size η.

6. Recompute all canonical labels R_i and C_j. Recompute the indicator variables $I_{i,j}$.
7. Define $X_i^{(\zeta)}$ as the subset of V_H corresponding to the ζ-clique labeled as R_i. Define a set S of row labels of $\mathbf{A}$ as
$$S = \left\{i : X_i^{(\zeta)} \subset T\right\}$$
8. Split every row of $\mathbf{A}$ into blocks of length $\log^\beta m$, i.e., $\mathbf{A}_i = (\mathbf{b}_{i,1} \| \cdots \| \mathbf{b}_{i,c_{H,\zeta-1}})$ where every $\mathbf{b}_{i,j}$ has width $\log^\beta m$. Define a set of block labels $S' \subset [c_{H,\zeta-1}]$ such that
$$S' = \{j : \exists i \in S \text{ such that } I_{i,j} = 1\}.$$
9. Let $\tilde{\mathbf{A}}$ be the submatrix of $\mathbf{A}$ with rows labeled by S and columns made by concatenating the blocks labeled by S'.
10. Solve the following system of linear equations using Gaussian elimination:
$$\tilde{\mathbf{A}}\mathbf{x} = \mathbf{b}|_S,$$
where $\mathbf{b}|_S$ is the vector $\mathbf{b}$ restricted to indices in S.
11. If the above step returns a valid solution, then output $b' = 0$. Otherwise output $b' = 1$.

5.1 Correctness

First, we show that K_H is almost surely very large.

Lemma 5.2. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$, for all large enough λ, the following holds. Let $\bar{n} = 2^{(\log \lambda)^{1/(1+\alpha)}}$, $\bar{k} = \bar{n}^\delta$, $\ell = (\log \bar{n})^\alpha$, and $q = \bar{n}^{-(\delta\ell)/2}$. Then the set K_H produced in algorithm* Gen *satisfies*
$$|K_H| > |V_H|^{\delta/2}$$
with all but $o(1)$ probability over the randomness of RP_p^ℓ.

Proof. The planted clique contains $\binom{\bar{k}}{\ell}$ many ℓ-tuples of vertices, and the entire graph G contains $\binom{\bar{n}}{\ell}$ many ℓ-tuples. Each ℓ-tuple becomes a vertex in H independently with probability $\bar{n}^{-\delta\ell/2}$, so a Chernoff argument shows that $|V_H| = \bar{n}^{(1-\delta/2 \pm o(1))\ell}$ with all but $o(1)$ probability. Since $\bar{k} = \bar{n}^\delta$, another Chernoff argument shows that the number of ℓ-tuples inherited from the planted clique is $\bar{n}^{(\delta/2 \pm o(1))\ell} = |V_H|^{\delta/(2-\delta) \pm o(1)} > |V_H|^{\delta/2}$ with all but $o(1)$ probability, with the last inequality holding for all large enough λ. □

We use the above to argue decryption correctness.

Lemma 5.3. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$, for all large enough λ, the following holds. Let $\bar{n} = 2^{(\log \lambda)^{1/(1+\alpha)}}$, $\bar{k} = \bar{n}^\delta$, $\ell = (\log \bar{n})^\alpha$, $q = \bar{n}^{-(\delta\ell)/2}$, $\zeta > 1$ be a constant integer, $\beta = \Theta(1)$, and $\rho = 2 + \beta\zeta$. Then for all $b \in \{0, 1\}$*
$$\Pr_{(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{Gen}(1^{\bar{n}})}[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, b)) = b)] \geq 1 - o(1),$$
where the probability is over the coins of Encode, Gen, Enc, *and* Dec.

Proof. Let $\mathbf{A}$ have m rows and n columns. First observe that K_H is made of nodes that form a subgraph of K in G, and therefore K_H must form a clique. Moreover by Lemma 5.2, the size of K_H is more than $|V_H|^{\delta/2}$ with all but $o(1)$ probability.

Since ρ is a constant, we can always find an integer η satisfying the requirement for decryption. By construction, $|T| = \eta$, and T induces a clique in H. Moreover, S contains precisely all of the row labels for ζ-cliques that are subgraphs of T, and $|S| = \binom{\eta}{\zeta}$.

The encoding of H to $\mathbf{A}$ is such that for any row $\mathbf{A}_i$, the jth block $\mathbf{b}_{i,j}$ is non-zero (i.e. $I_{i,j} = 1$) iff the $(\zeta - 1)$-clique labeled as C_j is a subgraph of the ζ-clique labeled as R_i. As S contains all ζ sized cliques that are a subgraph of T, and T is a clique, the set S corresponds to row labels of all possible ζ-cliques that are subgraphs of T. This count is exactly $\binom{\eta}{\zeta}$. A similar argument shows that the size of S' is exactly $\binom{\eta}{\zeta-1}$. Thus, the total number of columns in $\tilde{\mathbf{A}}$ is $\binom{\eta}{\zeta-1} \cdot \log^\beta m$, and the total number of rows is $\binom{\eta}{\zeta}$.

Note that for every choice of $\mathbf{x} \in \mathbb{F}_p^{\binom{\eta}{\zeta-1} \cdot \log^\beta m}$, we have that $\tilde{\mathbf{A}}\mathbf{x}$ can map to at most one unique vector in $\mathbb{F}_p^{\binom{\eta}{\zeta}}$. Also, since ζ is a constant and $\binom{\eta}{\zeta} = \Theta((\log n)^{\rho-1})$, we have that $\eta = \Theta((\log n)^{(\rho-1)/\zeta})$ by the Stirling approximation. Thus if we encrypted a 1, i.e. $\mathbf{b}|_S$ is a random vector in $\mathbb{F}_p^{|S|}$, then the probability that Gaussian elimination will return a solution is upper bounded as follows.

$$
\begin{aligned}
\Pr[\text{returns solution}] \leq & \frac{p^{\binom{\eta}{\zeta-1} \cdot \log^\beta m}}{p^{\binom{\eta}{\zeta}}} \\
\leq & \frac{p^{\binom{\eta}{\zeta-1} \cdot \log^\beta m}}{p^{\binom{\eta}{\zeta}}} \\
\leq & 2^{\binom{\eta}{\zeta-1} \cdot \log^\beta m - \binom{\eta}{\zeta}} \\
< & 2^{\eta^{\zeta-1} \cdot \log^\beta m - \Omega(\eta^\zeta)} && \text{(Stirling approximation)} \\
\leq & 2^{O((\log n)^{(\rho-1)\cdot(\zeta-1)/\zeta} \cdot \log^\beta m) - \Omega((\log n)^{\rho-1})} \\
< & 2^{O((\log n)^{(\rho-1)\cdot(\zeta-1)/\zeta} \cdot (\log n)^\beta) - \Omega((\log n)^{\rho-1})} \\
& && (m \text{ and } n \text{ are polynomially related w.h.p., and } \beta \text{ is constant}) \\
\leq & 2^{O((\log n)^{(1+\beta\zeta)\cdot(\zeta-1)/\zeta+\beta)} - \Omega((\log n)^{1+\beta\zeta})} && (\rho = 2 + \beta\zeta) \\
\leq & 2^{O((\log n)^{1+\beta\zeta-1/\zeta}) - \Omega((\log n)^{1+\beta\zeta})} \\
\leq & 2^{-\Omega((\log n)^{1+\beta\zeta})} && (\zeta \text{ is a constant})
\end{aligned}
$$

With (very) high probability, n is superconstant in λ. Therefore,

$$\Pr[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^n) \wedge \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, 1)) = 1)] \geq 1 - o(1).$$

Let us look at the case that a 0 was encrypted. Here $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$. Let S_{col} be the set of columns for the blocks in S'. The linear system solved by Gaussian elimination becomes

$$\tilde{\mathbf{A}}\mathbf{x} = \tilde{\mathbf{A}} \cdot (\mathbf{s}|_{S_{\text{col}}}) + \mathbf{e}|_S.$$

Now, every entry of $\mathbf{e}$ has is nonzero independently with probability at most $(\log n)^{-\rho}$. So the probability that $\mathbf{e}|_S = \mathbf{0}$ is at most

$$(1 - (\log n)^{-\rho})^{|S|} = (1 - (\log n)^{-\rho})^{\binom{\eta}{\zeta}} \geq (1 - (\log n)^{-\rho})^{\Theta((\log n)^{(\rho-1)})} \geq 1 - o(1),$$

again using that n is superconstant in λ with (very) high probability.

If this happens, then Gaussian elimination must return a valid solution. If there is at least one error, it is still possible that we get a valid solution, but this only increases the decryption correctness. Therefore

$$\Pr[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^n) \wedge \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, 0)) = 0)] \geq 1 - o(1).$$

We conclude that for all $b \in \{0, 1\}$,

$$\Pr[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^n) \wedge \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, b)) = b)] \geq 1 - o(1).$$

□

5.2 Security

The proof of security will follow from standard hybrid arguments by reducing to the hardness of planted clique (Conjecture 1.1) and noisy k-LIN (Conjecture 4.3). To use the latter, we first need to show that the indicator matrix $\mathbf{S}$ for $\mathbf{A}$ indeed satisfies the pre-conditions of Conjecture 4.3. Namely, $\mathbf{S}$ should be an expanding matrix when $\mathbf{S}$ is constructed from a graph G without a planted clique. We will also show that $\mathbf{S}$ is a non-compressing matrix when constructed in this manner. We present the proofs below.

5.2.1 Proof of Expansion

Before showing the proof of expansion, we prove the following lemma regarding sparse vectors.

Lemma 5.4. *Let $c_1, c_2 > 0$ be any constants. Let $n \in \mathbb{N}$ and $\Delta = (\log n)^{c_2}$. Let T be a set of $n^{O(1)}$ many uniformly sampled t-sparse vectors in $\mathbb{F}_2^{(\log n)^{c_1}}$, where $t = \Theta(\log n)$. Then, for all subsets $S \subset T$ of size at most Δ,*

$$\mathsf{hw}(\bigvee_{\mathbf{s} \in S} \mathbf{s}) > \frac{c_1 - c_2 - 2}{c_1} \cdot \sum_{\mathbf{s} \in S} \mathsf{hw}(\mathbf{s})$$

with all but $o(1)$ probability over the choice of the vectors.

Proof. Clearly the lemma holds if $c_1 - c_2 - 2 \leq 0$. So consider the case that $c_1 - c_2 - 2 > 0$.

We use linearity of expectation to calculate the number of sets of vectors S with $|S| \leq \Delta$ such that

$$\mathsf{hw}(\bigvee_{\mathbf{s} \in S} \mathbf{s}) \leq \frac{c_1 - c_2 - 2}{c_1} \cdot \sum_{\mathbf{s} \in S} \mathsf{hw}(\mathbf{s}) = \frac{c_1 - c_2 - 2}{c_1} t|S|.$$

Let $x := |S|$ and the number of sets of size x that contradict the lemma be N_x. We want all non-zero entries of each $\mathbf{s} \in S$ to be restricted to a subset of $[(\log n)^{c_1}]$ that has size at most $\frac{c_1 - c_2 - 2}{c_1} tx$.

Now, the total number of ways to choose a set of x many vectors is $\binom{n^{O(1)}}{x} \leq n^{O(x)}$, and the total number of ways to select $\frac{c_1 - c_2 - 2}{c_1} tx$ many positions out of a total of $(\log n)^{c_1}$ positions is

$$\binom{(\log n)^{c_1}}{\frac{c_1 - c_2 - 2}{c_1} tx} < ((\log n)^{c_1})^{\frac{c_1 - c_2 - 2}{c_1} tx}.$$

The probability that all the non-zero entries for all vectors $\mathbf{x} \in S$ lie within a given set of $\frac{c_1-c_2-2}{c_1}tx$ many positions is

$$\left(\frac{\binom{\frac{c_1-c_2-2}{c_1}tx}{t}}{\binom{(\log n)^{c_1}}{t}}\right)^x < \left(\frac{\frac{c_1-c_2-2}{c_1}tx}{(\log n)^{c_1}}\right)^{tx} < \left(\frac{tx}{(\log n)^{c_1}}\right)^{tx} < \left(\frac{(\log n)^{3/2}\Delta}{(\log n)^{c_1}}\right)^{tx} = ((\log n)^{3/2+c_2-c_1})^{tx}.$$

Putting all of this together, we have:

$$\begin{aligned}
\mathbb{E}[N_x] \leq & n^{O(x)} \cdot ((\log n)^{c_1})^{\frac{c_1-c_2-2}{c_1}tx} \cdot ((\log n)^{3/2+c_2-c_1})^{tx} \\
\leq & 2^{O(x\log n)} \cdot 2^{(c_1-c_2-2)tx\log\log n} \cdot 2^{(3/2+c_2-c_1)tx\log\log n} \\
\leq & 2^{O(x\log n)} \cdot 2^{((c_1-c_2-2)+(3/2+c_2-c_1))tx\log\log n} \\
\leq & 2^{O(x\log n)} \cdot 2^{-\frac{1}{2}tx\log\log n} \\
\leq & 2^{O(x\log n)} \cdot 2^{-\omega(x\log n)} \\
\leq & 2^{-\omega(x\log n)}
\end{aligned}$$

Finally taking a union bound of all $x \leq \Delta$ and applying Markov's inequality, we get that

$$\Pr\left[N \coloneqq \sum_x N_x \geq 1\right] \leq 2^{-\omega(\log n)}.$$

□

Next, we state a useful lemma about ζ-cliques and $(\zeta-1)$-cliques, the proof of which is deferred to Section 6. To state the lemma, we begin by defining a notion of "high" and "low" degree that applies to cliques.

Definition 6.6 ((Δ, S)-High Degree)**.** *A $(\zeta-1)$-clique is said to be* (Δ, S)-high degree *if it is contained in at least Δ many distinct ζ-cliques of S, where S is a set of ζ-cliques.*

Lemma 6.7. *Let $\bar{n} \in \mathbb{N}$ be any parameter, $\zeta > 1$ be any (not necessarily constant) integer, and $\alpha, \delta \in (0, \frac{1}{2})$ be any constants. Set $\ell = (\log \bar{n})^\alpha$, $q = \bar{n}^{-(\delta\ell)/2}$, and $\Delta = (\log \bar{n})^{\zeta(1-\alpha)}$. Sample $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$, and sample $H \xleftarrow{\$} \mathrm{RP}_q^\ell(G)$. With all but $o(1)$ probability over the random choice of G and randomness of RP_q^ℓ, the following occurs. For any set S of $2^{o((\log \bar{n})^{2\alpha})}$ many ζ-cliques of H, define $T \subset S$ as*

$$T = \{K \in S : K \text{ has as a subgraph more than one } (\Delta, S)\text{-high degree } (\zeta-1)\text{-clique}\}.$$

Then $|T| = o(1) \cdot |S|$.

In other words, the above lemma says that for any set S consisting of ζ-cliques, as long as the size of S is not too high, nearly all of the ζ-cliques in S contain at most one high degree $(\zeta-1)$-clique.

Using the above lemma, we can argue that $\mathbf{A}$ is a good small-set-expander.

Lemma 5.5. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$ and $\gamma \in (\frac{1}{2}, 1)$, for all large enough λ, the following holds. Set $\bar{n} = 2^{(\log\lambda)^{1/(1+\alpha)}}$, $\ell = (\log \bar{n})^\alpha$, $q = \bar{n}^{-(\delta\ell)/2}$, $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$, and $\beta = \zeta \cdot (\zeta(1-\alpha)+2)$. Let $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$, $H \xleftarrow{\$} \mathrm{RP}_q^\ell(G)$, and $\mathbf{A} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Then $\mathbf{A} \in \mathbb{F}_2^{m\times n}$ is a $(\gamma, \Theta(\log n), 2^{(\log n)^\alpha})$-expanding matrix with $1 - o(1)$ probability over the random choice of G, the randomness of RP_q^ℓ, and the coins of* Encode.

Proof. First note that, by construction, each row of $\mathbf{A}$ is exactly k-sparse, where $k = \log m = \Theta(\log n)$. ($m$ and n are polynomially related with probability $1 - o(1)$ over the random choice of G and the randomness of RP_p^ℓ.)

Recall that our small-set-expansion requirement is the following, where $t = 2^{(\log n)^\alpha}$.

$$\forall S \subseteq \{1, \dots m\} \text{ such that } |S| \leq t, \text{ we have that } \mathsf{hw}(\vee_{i \in S} \mathbf{S}_i) \geq \gamma \cdot k|S|.$$

Consider any set of indices S with $|S| \leq t$. Define B to be the set of all blocks $\mathbf{b}_{i,j}$ such that $i \in S$ and $I_{i,j} = 1$. We now lower bound $\mathsf{hw}(\vee_{i \in S} \mathbf{S}_i)$ by analyzing the blocks in B.

Fix $\Delta = (\log \bar{n})^{\zeta(1-\alpha)}$. Let $K(S) = \{\zeta\text{-clique of } H \text{ with label } R_i : i \in S\}$. Let $S_{\text{Bad}} \subset S$ be the set of all $i \in S$ such that the ζ-clique labeled R_i contains at least two $(\Delta, K(S))$-high degree $(\zeta - 1)$-cliques.

Observe that, since $n = |V_H|^{O(1)} = \bar{n}^{O(\ell)}$ and $\alpha \in (0, \frac{1}{2})$ is a constant, we have that $\log^\alpha n = o(\log^{2\alpha} \bar{n})$. So $|K(S)| \leq 2^{o(\log^{2\alpha} \bar{n})}$. Therefore by Lemma 6.7,

$$|S_{\text{Bad}}| \leq o(1) \cdot |S|$$

Define B_{Good} to be the set of all blocks $\mathbf{b} \in B$ that are contained in a row $\mathbf{S}_i$ with $i \in S \backslash S_{\text{Bad}}$. Since every row $\mathbf{S}_i$ contains exactly ζ many blocks $\mathbf{b}_{i,j}$ with $I_{i,j} = 1$:

$$|B_{\text{Good}}| \geq (1 - o(1)) \cdot |B|.$$

Let $B_{\text{Good, Low}}$ be the set of all blocks $\mathbf{b}_{i,j} \in B_{\text{Good}}$ such that C_j is the label of a $(\zeta - 1)$-clique that is *not* $(\Delta, K(S))$-high degree. Since each ζ-clique with label R_i for some $i \in S \backslash S_{\text{Bad}}$ has that all but one of its $(\zeta - 1)$-cliques are not $(\Delta, K(S))$-high degree,

$$|B_{\text{Good, Low}}| \geq \frac{\zeta - 1}{\zeta} \cdot |B_{\text{Good}}| \geq \frac{\zeta - 1}{\zeta} \cdot (1 - o(1)) \cdot |B|.$$

Since each block has exactly the same hamming weight, it must be the case that

$$\sum_{\mathbf{b} \in B_{\text{Good, Low}}} \mathsf{hw}(\mathbf{b}) = \frac{|B_{\text{Good, Low}}|}{|B|} \cdot \sum_{\mathbf{b} \in B} \mathsf{hw}(\mathbf{b}).$$

Furthermore

$$\sum_{\mathbf{b} \in B} \mathsf{hw}(\mathbf{b}) = \sum_{i \in S} \mathsf{hw}(\mathbf{A}_i) = k|S|$$

since all of the nonzero entries in each row appear in a block $\mathbf{b}_{i,j}$ with $I_{i,j} = 1$. Composing these results, we have

$$\sum_{\mathbf{b} \in B_{\text{Good, Low}}} \mathsf{hw}(\mathbf{b}) \geq \frac{\zeta - 1}{\zeta} \cdot (1 - o(1)) \cdot k|S|.$$

Let $B_{\text{Good, Low},j}$ be the set of all $\mathbf{b}_{i,j} \in B_{\text{Good, Low}}$ with a fixed j value. Recall that each block in $B_{\text{Good, Low},j}$ is a $(\log m)$-sparse vector in $\mathbb{F}_2^{\log^\beta m}$. Furthermore, since $m > \bar{n}$ w.h.p, $B_{\text{Good, Low},j}$ contains at most $\Delta = (\log \bar{n})^{\zeta(1-\alpha)} < (\log m)^{\zeta(1-\alpha)}$ many vectors selected out of a total of $m^{O(1)}$ many vectors.

(The set of vectors is taken to be the set of all blocks $\mathbf{b}_{i,j}$ in $\mathbf{S}$.) We now apply Lemma 5.4 with $c_1 = \beta = \zeta \cdot (\zeta(1-\alpha)+2)$ and $c_2 = \zeta(1-\alpha)$. For each j, we have that

$$\begin{aligned}
\mathsf{hw}(\bigvee_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathbf{b}) &> \frac{\zeta\cdot(\zeta(1-\alpha)+2)-\zeta(1-\alpha)-2}{\zeta\cdot(\zeta(1-\alpha)+2)} \cdot \sum_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathsf{hw}(\mathbf{b}) \\
&= \frac{\zeta\cdot(\zeta(1-\alpha)+2)-1\cdot(\zeta(1-\alpha)+2)}{\zeta\cdot(\zeta(1-\alpha)+2)} \cdot \sum_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathsf{hw}(\mathbf{b}) \\
&= \frac{(\zeta-1)\cdot(\zeta(1-\alpha)+2)}{\zeta\cdot(\zeta(1-\alpha)+2)} \cdot \sum_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathsf{hw}(\mathbf{b}) \\
&= \frac{\zeta-1}{\zeta} \cdot \sum_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathsf{hw}(\mathbf{b})
\end{aligned}$$

To determine $\mathsf{hw}(\vee_{i\in S}\mathbf{S}_i)$, we can sum the hamming weights for each section in the final vector. Considering only the blocks in $B_{\text{Good, Low}}$, we can write this as:

$$\mathsf{hw}(\vee_{i\in S}\mathbf{S}_i) \geq \sum_j \left(\mathsf{hw}(\bigvee_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathbf{b}) \right)$$

Now composing all of our results and using that $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$:

$$\begin{aligned}
\sum_j \left(\mathsf{hw}(\bigvee_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathbf{b}) \right) &> \sum_j \left(\frac{\zeta-1}{\zeta} \cdot \sum_{\mathbf{b}\in B_{\text{Good, Low},j}} \mathsf{hw}(\mathbf{b}) \right) \\
&= \frac{\zeta-1}{\zeta} \cdot \sum_{\mathbf{b}\in B_{\text{Good, Low}}} \mathsf{hw}(\mathbf{b}) \\
&\geq \frac{\zeta-1}{\zeta} \cdot \frac{\zeta-1}{\zeta} \cdot (1-o(1)) \cdot k|S| \\
&= \left(\frac{\lceil \frac{1}{1-\sqrt{\gamma}} \rceil}{\lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1} \right)^2 \cdot (1-o(1)) \cdot k|S| \\
&> \left(\frac{\frac{1}{1-\sqrt{\gamma}} - 1}{\frac{1}{1-\sqrt{\gamma}}} \right)^2 \cdot k|S| \\
&= \left(\frac{1-(1-\sqrt{\gamma})}{1} \right)^2 \cdot k|S| \\
&= \gamma \cdot k|S|.
\end{aligned}$$

Thus, we achieve the desired lower bound with all but $o(1)$ probability. □

5.2.2 Proof of Non-Compression

As stated in Section 4, $\mathbf{S}$ will also be a $(\gamma, \Theta(\log n))$-non-compressing matrix with high probability. We begin our proof by stating the Kruskal-Katona theorem [Kru63, Kat87]. At a high level, this relates the number unique sets of size $\zeta - 1$ contained in a collection of sets, each of which has size ζ, for arbitrary values of ζ.

Lemma 5.6 (Simplified version of the Kruskal-Katona Theorem [Kru63, Kat87])**.** *Consider any set system $\mathcal{T}$ where every $T \in \mathcal{T}$ has cardinality ζ, for any constant integer $\zeta > 1$. Define*

$$\mathcal{Q} = \{C : |C| = (\zeta - 1) \wedge (\exists T \in \mathcal{T} \text{ such that } C \subset T)\}$$

Then $|\mathcal{Q}| \geq \Omega(|\mathcal{T}|^{(\zeta-1)/\zeta})$.

Lemma 5.7. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$ and $\gamma \in (\frac{1}{2}, 1)$ and $\beta > 0$, for all large enough λ, the following holds. Let $\bar{n} = 2^{(\log \lambda)^{1/(1+\alpha)}}$, $\bar{k} = \bar{n}^{\delta}$, $\ell = (\log \bar{n})^{\alpha}$, $q = \bar{n}^{-(\delta\ell)/2}$, $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$, and $\rho = 2 + \beta\zeta$. Let $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$, $H \xleftarrow{\$} \mathrm{RP}^{\ell}_{q}(G)$, and $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Then $\mathbf{S} \in \mathbb{F}_2^{m \times n}$ is a $(\gamma, \Theta(\log n))$-non-compressing matrix with probability $1 - o(1)$ over the randomness of* Gen.

Proof. As argued in Lemma 5.5, each row of $\mathbf{S}$ is exactly k-sparse, where $k = \Theta(\log n)$ with $1 - o(1)$ probability.

Consider any set system $\mathcal{K} = \{X_i^{(\zeta)} : i \in P\}$, where $P \subseteq \{1, \ldots m\}$ and each $X_i^{(\zeta)}$ is the set of vertices in H for the ζ-clique of H with label R_i. Define

$$\mathcal{L} = \{L : |L| = (\zeta - 1) \wedge (\exists X \in \mathcal{K} \text{ such that } L \subset X)\}.$$

In other words, $\mathcal{L}$ is the set of all $(\zeta - 1)$-cliques contained in some ζ-clique of $\mathcal{K}$. Since $\gamma \in (\frac{1}{2}, 1)$ is a constant and $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$, we have that $\zeta > 4$. Therefore by Lemma 5.6, we have that $|\mathcal{L}| \geq \Omega(|\mathcal{K}|^{(\zeta-1)/\zeta})$. Expanding this expression, we get:

$$\begin{aligned}
|\mathcal{L}| \geq& \Omega(|\mathcal{K}|^{\lceil \frac{1}{1-\sqrt{\gamma}} \rceil / (\lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1)}) \\
>& \Omega(|\mathcal{K}|^{(\lceil \frac{1}{1-\sqrt{\gamma}} \rceil - 1)/\lceil \frac{1}{1-\sqrt{\gamma}} \rceil}) \\
=& \Omega(|\mathcal{K}|^{1 - 1/\lceil \frac{1}{1-\sqrt{\gamma}} \rceil}) \\
\geq& \Omega(|\mathcal{K}|^{1 - 1/(\frac{1}{1-\sqrt{\gamma}})}) \\
=& \Omega(|\mathcal{K}|^{\sqrt{\gamma}})
\end{aligned}$$

Now consider the set of row vectors $\mathcal{S}$ for $\mathbf{S}$. Any subset $\mathcal{T} \subset \mathcal{S}$ corresponds to exactly $|\mathcal{T}|$ many distinct ζ-cliques in H, and (by the above) at least $\Omega(|\mathcal{T}|^{\sqrt{\gamma}})$ many $(\zeta - 1)$-cliques of H. By construction, each of these $(\zeta - 1)$-cliques corresponds to at least one distinct "1" entry in the bitwise-OR of all vectors $\bigvee_{\mathbf{v} \in \mathcal{T}} \mathbf{v}$. Therefore we have

$$\mathsf{hw}\left(\bigvee_{\mathbf{v} \in \mathcal{T}} \mathbf{v}\right) \geq \Omega(|\mathcal{T}|^{\sqrt{\gamma}}).$$

Since each row vector is exactly k-sparse, we also have that any subset $\mathcal{T} \subset \mathcal{S}$ with $|\mathcal{T}| \leq k$ satisfies

$$\mathsf{hw}\left(\bigvee_{\mathbf{v} \in \mathcal{T}} \mathbf{v}\right) \geq k \geq |\mathcal{T}|.$$

Combining these two observations and noting that k is superconstant implies that

$$\mathsf{hw}\left(\bigvee_{\mathbf{v} \in \mathcal{T}} \mathbf{v}\right) \geq |\mathcal{T}|^{\gamma}.$$

holds for sufficiently large λ. □

5.2.3 Semantic Security

We first show that the scheme is a weak PKE.

Lemma 5.8 (Weak Security)**.** *Assume that Conjecture 1.1 holds for some constants $\alpha, \delta \in (0, \frac{1}{2})$, and assume that Conjecture 4.3 holds for some constant $\gamma \in (\frac{1}{2}, 1)$. Then for all large enough λ, the following is true. Set $\bar{n} = 2^{(\log \lambda)^{1/(1+\alpha)}}$, $\bar{k} = \bar{n}^{\delta}$, $\ell = (\log \bar{n})^{\alpha}$, $q = \bar{n}^{-(\delta\ell)/2}$, $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$, $\beta = \zeta \cdot (\zeta(1-\alpha)+2)$, and $\rho = 2 + \beta\zeta$. For all poly(λ)-size algorithms $\mathcal{A}$,*

$$\Big| \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^{\lambda}); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 0)) = 1, \right] - \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^{\lambda}); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 1)) = 1\right] \Big| \leq \frac{12}{13},$$

where the probability is taken over the coins of Encode*,* Gen*, and* Enc*.*

Proof. We will use standard hybrid arguments.

Hybrid $\mathcal{H}_0(1^{\lambda})$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2}, \bar{k})$.
2. $H \xleftarrow{\$} \mathrm{RP}_p^{\ell}(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{S}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$ where
$$\mathsf{ct} \leftarrow \mathbf{A}\mathbf{s} + \mathbf{e}.$$
Here $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$ and we sample $\mathbf{e}$ as:
$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\varsigma}. \\ 0, \text{ otherwise.} \end{cases}$$
7. Output $(\mathsf{pk}, \mathsf{ct})$.

Hybrid $\mathcal{H}_{0,\$}(1^{\lambda})$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$.
2. $H \xleftarrow{\$} \mathrm{RP}_p^{\ell}(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{S}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.

5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$ where
$$\mathsf{ct} \leftarrow \mathbf{As} + \mathbf{e}.$$
Here $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$ and we sample $\mathbf{e}$ as:
$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\rho}. \\ 0, \text{ otherwise.} \end{cases}$$
7. Output $(\mathsf{pk}, \mathsf{ct})$.

Hybrid $\mathcal{H}_{1,\$}(1^\lambda)$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$.
2. $H \xleftarrow{\$} \mathrm{RP}_p^\ell(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{S}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 1)$ where
$$\mathsf{ct} \leftarrow \mathbf{b}$$
where $\mathbf{b} \xleftarrow{\$} \mathbb{F}_p^m$.
7. Output $(\mathsf{pk}, \mathsf{ct})$.

Hybrid $\mathcal{H}_1(1^\lambda)$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2}, \bar{k})$.
2. $H \xleftarrow{\$} \mathrm{RP}_p^\ell(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{S}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 1)$ where
$$\mathsf{ct} \leftarrow \mathbf{b}$$
where $\mathbf{b} \xleftarrow{\$} \mathbb{F}_p^m$.

7. Output $(\mathsf{pk}, \mathsf{ct})$.

Before analyzing the hybrids, note that $\lambda = \bar{n}^{\Theta(\ell)}$ and $\lambda = n^{\Theta(1)}$ with very high probability.

Claim 5.9 ($\mathcal{H}_0 \approx_{\frac{1}{3}} \mathcal{H}_{0,\$}$)**.** *Assume that Conjecture 1.1 holds for some constants $\alpha, \delta \in (0, \frac{1}{2})$. For all sufficiently large λ, and for all constant $\gamma \in (\frac{1}{2}, 1)$, the following is true. Set the parameters as in the lemma statement. For all poly(λ)-size algorithms $\mathcal{A}$,*

$$\left|\Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_0(1^\lambda)) = 1\right] - \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{0,\$}(1^\lambda)) = 1\right]\right| \leq \frac{1}{3}.$$

Proof. The only difference between $\mathcal{H}_0$ and $\mathcal{H}_{0,\$}$ is the presence or absence of a planted clique. Therefore, a poly(λ)-size algorithm $\mathcal{A}$ for differentiating between the two hybrids implies a poly(λ)-size algorithm $\mathcal{B}$ for the planted clique problem.

Specifically, $\mathcal{B}$ uses its input graph G^* in place of graph G from Step 1, then follows steps 2 through 7 (which are the same in both hybrids and can be completed in poly(λ) time), and finally gives the result as input to $\mathcal{A}$. The output from $\mathcal{A}$ is used as the output for $\mathcal{B}$. Clearly by our choice of parameters, the distinguishing advantage of $\mathcal{B}$ is identical to that of $\mathcal{A}$.

□

Claim 5.10 ($\mathcal{H}_{0,\$} \approx_{\frac{1}{4}+o(1)} \mathcal{H}_{1,\$}$)**.** *Assume that Conjecture 4.3 holds for some constant $\gamma \in (\frac{1}{2}, 1)$. For all sufficiently large λ, and for all constants $\alpha, \delta \in (0, \frac{1}{2})$, the following is true. Set the parameters as in the lemma statement. For all poly(λ)-size algorithms $\mathcal{A}$,*

$$\left|\Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{0,\$}(1^\lambda)) = 1\right] - \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{1,\$}(1^\lambda)) = 1\right]\right| \leq \frac{1}{4} + o(1).$$

Proof. We have already proved that when we start with a graph $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$, then the matrix $\mathbf{A}$ generated as per our encoding procedure above satisfies the preconditions of Conjecture 4.3, namely Lemma 5.5, with $1 - o(1)$ probability. Assuming this event occurs, it is easy to see that any adversary that can distinguish between Hybrids $\mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$}$ with advantage greater than $\frac{1}{4}$ can be transformed into one that can distinguish between $(\mathbf{A}, \mathbf{As} + \mathbf{e})$ and $(\mathbf{A}, \$)$ with advantage better than $\frac{1}{4}$. This will contradict Conjecture 4.3.

□

Claim 5.11 ($\mathcal{H}_{1,\$} \approx_{\frac{1}{3}} \mathcal{H}_1$)**.** *Assume that Conjecture 1.1 holds for some constants $\alpha, \delta \in (0, \frac{1}{2})$. For all sufficiently large λ, and for all constant $\gamma \in (\frac{1}{2}, 1)$, the following is true. Set the parameters as in the lemma statement. For all poly(λ)-size algorithms $\mathcal{A}$,*

$$\left|\Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{1,\$}(1^\lambda)) = 1\right] - \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_1(1^\lambda)) = 1\right]\right| \leq \frac{1}{3}.$$

Proof. This follows identically from the proof of Claim 5.9. □

Combining the above three claims, and accounting for the fact that $\lambda = \bar{n}^{\Theta(\ell)}$ and $\lambda = n^{\Theta(1)}$ with very high probability, we get

$$\Big|\Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 0)) = 1,\right] - \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 1)) = 1\right]\Big| \leq \frac{11}{12} + o(1) < \frac{12}{13}.$$

□

We are now ready to prove our main theorem.

Theorem 5.12. *Assuming Conjecture 1.1 and Conjecture 4.3 both hold, there exists a semantically secure public key encryption scheme.*

Proof. Clearly $o(1) < \frac{1-\sqrt{12/13}}{2}$. Thus, by Lemma 5.8, the encryption scheme constructed above is an $\left(o(1), \frac{12}{13}\right)$ weak PKE. The theorem then follows from Lemma 3.4. □

6 Useful Properties of Random Graphs

In this section, we will prove certain properties of random graphs that will eventually be helpful towards proving Lemma 6.7.

Throughout, we will consider $G := (V_G, E_G) \xleftarrow{\$} \mathcal{G}(n, \frac{1}{2})$ and $H := (V_H, E_H) \xleftarrow{\$} \mathrm{RP}_q^\ell(G)$. Let $\bar{n} := |V_G|$, $q := \bar{n}^{-(\delta\ell)/2}$, and $\ell := (\log \bar{n})^\alpha$, for any positive constants $\alpha, \delta \in (0, \frac{1}{2})$. Recall that we can interpret every node $v \in V_H$ as corresponding to a set U_v of ℓ many nodes in G, and interpret any edge $e \in E_H$ connecting nodes $(v, w) \in V_H$ as corresponding to a set of edges $W_e = K(U_v \cup U_w)$. Here $K(X)$ denotes the clique induces by vertices in the set X.

Definition 6.1 (Vertex and Edge Mappings)**.** *For a subset of vertices $S \subseteq V_H$, define*

$$G^V(S) := \bigcup_{v \in S} U_v,$$

where U_i is as defined above, and for a subset of edges $T \subseteq E_H$, define

$$G^E(T) := \bigcup_{e \in T} W_e,$$

where W_e is as defined above.

In other words $G^V(S)$ is the union of vertices in G for each ℓ-subset corresponding to a vertex of S and $G^E(T)$ is the union of edges in G for each clique in G corresponding to an edge of T.

Lemma 6.2. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$, there exists a constant $c = c(\delta)$ such that the following is true. Let $\bar{n}$ be any parameter. Set $\ell = (\log \bar{n})^\alpha$ and $q = \bar{n}^{-(\delta\ell)/2}$. Let G be any graph with $\bar{n}$ many vertices, and sample $H \xleftarrow{\$} \mathrm{RP}_q^\ell(G)$. With all but $o(1)$ probability over the randomness of RP_q^ℓ we have that, for all $S \subset V_H$ with $|S| \leq \bar{n}^{o(1)}$, $|G^V(S)| > c\ell|S|$.*

Proof. Suppose for contradiction that there exists some $S \subset V_H$ with $|S| = \bar{n}^{o(1)}$ and such that $|G^V(S)| \leq c\ell|S|$. This implies that there exists an $x = \bar{n}^{o(1)}$ and some subset $S' \subset V_G$ of size $y \leq c\ell x$ with the property that at least x many vertices v in V_H have $U_v \subset S'$. If this event occurs for some $y < c\ell x$, this automatically implies that the event also occurs for all larger values of y, so throughout the proof we will assume $y = c\ell x$.

Let N_x denote the expected number of such subsets S' for a fixed choice of x, and $N = \sum_{x \leq \bar{n}^{o(1)}} N_x$.

Observe that there are $\binom{\bar{n}}{y}$ many ways to choose a set S' of y many nodes in G. Furthermore, $\binom{y}{\ell}$ is the total number of ℓ-sized subsets of a given S', so there are $\binom{\binom{y}{\ell}}{x}$ many ways to choose x many ℓ-sized subsets. The probability that all x of these were actually sampled by RP_q^ℓ (in other words, the probability that they all exist as vertices of H) is q^x.

Take $c = \delta/5$. We show by linearity of expectation that

$$\begin{aligned}
\mathbb{E}[N_x] \leq & \binom{\bar{n}}{y} \cdot \binom{\binom{y}{\ell}}{x} \cdot q^x \\
< & \bar{n}^y \cdot y^{x\ell} \cdot p^q \\
\leq & 2^{y\log(\bar{n}) + x\ell\log(y) - \frac{1}{2}x\ell\delta\log(\bar{n})} \\
\leq & 2^{y\log(\bar{n}) + x\ell\cdot(\log(y) - \frac{1}{2}\delta\log(\bar{n}))} \\
\leq & 2^{y\log(\bar{n}) + x\ell\cdot(\log x + \log\ell + \log c - \frac{1}{2}\delta\log(\bar{n}))} && (y = cx\ell) \\
\leq & 2^{y\log(\bar{n}) + x\ell\cdot(o(1) - \frac{1}{2}\delta)\log(\bar{n})} && (\ell \leq \bar{n}^{o(1)}, x \leq \bar{n}^{o(1)}) \\
\leq & 2^{-x\ell\cdot\frac{\delta}{4}\log(\bar{n})} = 2^{-x\cdot\omega(\log\bar{n})} && (y = cx\ell, \text{ and } c < \delta/4)
\end{aligned}$$

Taking a union bound over all possible values of x gives

$$\mathbb{E}[N] = \sum_{x \leq n^{o(1)}} \mathbb{E}[N_x] = \sum_{x \leq n^{o(1)}} 2^{-x\omega(\log\bar{n})} \leq 2^{-\omega(\log\bar{n})}.$$

Finally using Markov's inequality,

$$\Pr[N \geq 1] \leq \mathbb{E}[N],$$

and therefore $N = 0$ with all but $o(1)$ probability.

□

Lemma 6.3. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$, there exists a constant $c^* = c^*(\delta)$ such that the following is true. Let $\bar{n}$ be any parameter. Set $\ell = (\log\bar{n})^\alpha$ and $q = \bar{n}^{-(\delta\ell)/2}$. Sample $G \xleftarrow{\$} \mathcal{G}(n, \frac{1}{2})$ and $H \xleftarrow{\$} \mathrm{RP}^\ell_q(G)$. With all but $o(1)$ probability over the random choice of G and the randomness of RP^ℓ_q we have that, for all subgraphs $S = (V_S, E_S)$ of H with $|V_S| = 2^{o((\log\bar{n})^{2\alpha})}$, $|E_S| < c^*(|V_S|(\log\bar{n})^{1-\alpha})$.*

Proof. Assume that the event in Lemma 6.2 occurs. Now suppose for the sake of contradiction that there exists a subgraph S of H with $|V_S| = 2^{o((\log\bar{n})^{2\alpha})}$ and $|E_S| \geq c^*|V_S|(\log\bar{n})^{1-\alpha}$.

We can actually argue the contradiction for just $|E_S| = c^*|V_S|(\log\bar{n})^{1-\alpha}$. This is because, if there exists a subgraph S with $|E_S| > c^*|V_S|(\log\bar{n})^{1-\alpha}$, we can arbitrarily delete edges of S to get a subgraph S' with $|E_{S'}| = c^*|V_{S'}|(\log\bar{n})^{1-\alpha}$ and $|V_{S'}| = |V_S|$.

Let $x \coloneqq |V_S|$ and $yx \coloneqq |E_S|$ (which implies that $y = c^*(\log\bar{n})^{1-\alpha}$). Let $N_S(v_S)$ denote the open neighborhood of $v_S \in V_S$ within the subgraph S. Without loss of generality, we may assume that the minimum degree of each vertex in S is $\lfloor y \rfloor = \lfloor c^*(\log\bar{n})^{1-\alpha} \rfloor$. If not, since y is independent of $|S|$, we can repeatedly delete each vertex of degree less than $\lfloor y \rfloor$ to get a smaller subgraph S' with minimum degree $\lfloor y \rfloor$ and $x' < x$ many vertices. For ease of exposition, we will actually just assume a minimum degree of $\frac{y}{2}$, since $\frac{y}{2} \leq \lfloor y \rfloor$ for any $y \geq 1$.

By Lemma 6.2, there exists some constant $c = f(\delta)$ such that

$$|G^V(N_S(v_S))| > c\ell|N_S(v_S)| \geq c\ell\frac{y}{2}$$

for all $v_S \in V_S$.

Define a mapping $\phi : G^V(V_S) \to V_S$ such that, for all $v_G \in G^V(V_S)$, we have that $\phi(v_G) = v_S$ for some $v_S \in V_S$ such that $v_G \in U_{v_S}$. There may be multiple choices of v_S for each v_G; just choose from these arbitrarily. By the definition of $G^V(\cdot)$, such a mapping always exists.

Let $E_\phi(v_G) \subset G^E(E_S)$ be the set of edges between a vertex $v_G \in G^V(V_S)$ and the vertices in $G^V(N_S(\phi(v_G)))$. Note that v_G must have an edge to every vertex in $G^V(N_S(\phi(v_G)))\backslash v_G$ by the definition of the randomized graph product. So $(|G^V(N_S(\phi(v_G)))| - 1)$ is a lower bound on the degree of $v_G \in G^V(V_S)$ with respect to the edges in $G^E(E_S)$.

We already showed that $|G^V(N_S(\phi(v_G)))| > c\ell\frac{y}{2}$ for all possible $\phi(v_G)$, and therefore this statement holds for all v_G. Thus $c\ell\frac{y}{2} - 1$ is a strict lower bound for the degree of each v_G with respect to $G^E(E_S)$. (In other words the min degree is at least $c\ell\frac{y}{2}$.)

Applying Lemma 6.2 again, and using that having constant $\alpha \in (0, \frac{1}{2})$ implies that $|V_S| = 2^{o((\log \bar{n})^{2\alpha})} = \bar{n}^{o(1)}$, we know that $|G^V(V_S)| > c\ell|V_S|$.

Let $z \coloneqq |G^E(E_S)|$. Putting all of this together, and observing that the total number of edges in any graph $Z = (V_Z, E_Z)$ is at least $\frac{1}{2} \cdot |V_Z| \cdot \text{MinDegree}(Z)$, we have

$$|G^E(E_S)| \geq \frac{1}{2} \cdot |G^V(V_S)| \cdot c\ell\frac{y}{2} > \frac{1}{4} \cdot c^2\ell^2 \cdot y|V_S| = \frac{1}{4} \cdot c^2\ell^2 \cdot yx.$$

We now apply linearity of expectation to show that the number of subgraphs S of H with $|V_S| = 2^{o((\log \bar{n})^{2\alpha})}$ and $|E_S| = c^*|V_S|(\log \bar{n})^{1-\alpha}$ is vanishing by analyzing the edge probabilities in G. Let N_x be the expected number of subgraphs in H with x nodes and yx edges. Setting $c^* = \frac{15}{c^2}$ and using that the edge probability in G is $\frac{1}{2}$, we have

$$\begin{aligned}
\mathbb{E}[N_x] =& \binom{|V_H|}{x} \cdot \binom{\binom{x}{2}}{xy} \cdot 2^{-z} \\
<& \bar{n}^{x\ell} \cdot x^{2xy} \cdot 2^{-z} && (|V_H| < \bar{n}^\ell) \\
<& 2^{x\ell \log \bar{n} + 2xy \log x - \frac{1}{4}c^2\ell^2 yx} && (|E^G(E_S)| > \tfrac{1}{4}c^2\ell^2 yx) \\
=& 2^{x(\ell \log \bar{n} + 2y \log x - \frac{1}{4}c^2\ell^2 y)} \\
<& 2^{x(\ell \log \bar{n} - \frac{1}{5}c^2\ell^2 y)} && (\log x = o(\ell^2)) \\
=& 2^{x(\ell \log \bar{n} - \frac{1}{5}c^2\ell c^* \log \bar{n})} && (\ell = (\log(\bar{n}))^\alpha \text{ and } y = c^*((\log \bar{n})^{1-\alpha})) \\
=& 2^{x(\ell \log \bar{n} - 3\ell \log \bar{n})} \\
\leq& 2^{-2x\ell \log \bar{n}}
\end{aligned}$$

Finally this gives us

$$\begin{aligned}\mathbb{E}[N] &= \sum_{x \leq 2^{o((\log \bar{n})^{2\alpha})}} \mathbb{E}[N_x] \\ &< \sum_{x \leq 2^{o((\log \bar{n})^{2\alpha})}} 2^{-2x\ell \log \bar{n}} \\ &< o(1)\end{aligned}$$

Using Markov's inequality,

$$\Pr[N \geq 1] \leq \mathbb{E}[N],$$

and therefore $N = 0$ with all but $o(1)$ probability (assuming the event in Lemma 6.2 occurred, which also happens with all but $o(1)$ probability).

□

Lemma 6.4. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$, there exists a constant $\tilde{c} = \tilde{c}(\delta)$ such that the following is true. Let $\bar{n}$ be any parameter and $\zeta > 1$ be any (not necessarily constant) integer. Set $\ell = (\log \bar{n})^{\alpha}$ and $q = \bar{n}^{-(\delta\ell)/2}$. Sample $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$, and sample $H \xleftarrow{\$} \mathrm{RP}_q^{\ell}(G)$. With all but $o(1)$ probability over the random choice of G and the randomness of RP_q^{ℓ} we have that, for all sets S of $2^{o((\log \bar{n})^{2\alpha})}$ many ζ-cliques in H, there exists a set S' of $(\zeta-1)$-cliques such that $|S'| \geq \frac{\tilde{c}\zeta}{(\log \bar{n})^{1-\alpha}}|S|$ and*

$$\forall C \in S', \exists K \in S \text{ such that } C \subset K$$

Proof. We give a proof by induction. Throughout the proof, we will take S' to be the set of all distinct $(\zeta-1)$-cliques contained in some ζ-clique of S.

For the base case of $\zeta = 2$, the lemma corresponds to lower bounding the ratio of vertices to edges, and so the claim holds by Lemma 6.3. For the inductive case, we will count the number of $(\zeta-1)$-cliques and ζ-cliques that each vertex participates in, and then use this to bound the overall ratio between $|S'|$ and $|S|$. All of the inductive cases are deterministic, so they follow immediately if the event in Lemma 6.3 occurs.

Consider some set S satisfying the condition in the lemma, and let V_S be the union of vertices for all cliques in S. We now examine an arbitrary vertex $v \in V_S$. Each distinct ζ-clique $K \in S$ with $v \in K$ contains a unique $(\zeta-1)$-clique, namely $C := K \setminus \{v\}$. Denote the set of all such $(\zeta-1)$-cliques to be S'_v. Note that $|S'_v| \leq |S|$, since each $K \in S$ contributes at most one $(\zeta-1)$-clique to S'_v. This implies that that the inductive hypothesis also bounds the ratio of $(\zeta-1)$-cliques to $(\zeta-2)$-cliques in S'_v.

Observe that v participates in exactly $|S'_v|$ many ζ-cliques. But also, v participates in exactly as many $(\zeta-1)$-cliques as there are $(\zeta-2)$-cliques fully contained in some $(\zeta-1)$-clique $C \in S'_v$. Denote this set of $(\zeta-2)$-cliques to be S''_v. Using the inductive hypothesis, $|S''_v| \geq \tilde{c} \cdot (\zeta-1) \cdot |S'_v|/(\log \bar{n})^{1-\alpha}$.

Now consider the sum over all counts for ζ-cliques. We have that

$$\sum_{v \in V_S} |S'_v| = \zeta \cdot |S|$$

since each ζ-clique is counted exactly ζ many times. By the same argument,

$$\sum_{v \in V_S} |S''_v| = (\zeta-1) \cdot |S'|.$$

Composing these results yields the lemma:

$$\begin{aligned}|S'| =& \frac{1}{\zeta-1}\sum_{v\in V_S}|S''_v| \\ \geq& \frac{1}{\zeta-1}\sum_{v\in V_S}(\tilde{c}\cdot(\zeta-1)\cdot|S'_v|/(\log\bar{n})^{1-\alpha}) \\ =& \frac{\tilde{c}}{(\log\bar{n})^{1-\alpha}}\sum_{v\in V_S}|S'_v| \\ =& \frac{\tilde{c}\zeta}{(\log\bar{n})^{1-\alpha}}|S|\end{aligned}$$

□

Corollary 6.5. *Let $\bar{n} \in \mathbb{N}$ be any parameter, $\zeta > 1$ be any (not necessarily constant) integer, and $\alpha, \delta \in (0, \frac{1}{2})$ be any constants. Set $\ell = (\log\bar{n})^{\alpha}$ and $q = \bar{n}^{-(\delta\ell)/2}$. Sample $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$, and sample $H \xleftarrow{\$} \mathrm{RP}^{\ell}_{q}(G)$. With all but $o(1)$ probability over the random choice of G and the randomness of RP^{ℓ}_{q}, the following occurs. Any subgraph $H' = (V_{H'}, E_{H'})$ of H with $|V_{H'}| = 2^{o((\log\bar{n})^{2\alpha})}$ contains at most $O(\frac{|V_{H'}|}{\zeta}(\log\bar{n})^{(\zeta-1)(1-\alpha)})$ many ζ-cliques.*

Proof. The proof follows by iteratively applying Lemma 6.4. □

Definition 6.6 ((Δ, S)-High Degree)**.** *A $(\zeta-1)$-clique is said to be* (Δ, S)-high degree *if it is contained in at least Δ many distinct ζ-cliques of S, where S is a set of ζ-cliques.*

Lemma 6.7. *Let $\bar{n} \in \mathbb{N}$ be any parameter, $\zeta > 1$ be any (not necessarily constant) integer, and $\alpha, \delta \in (0, \frac{1}{2})$ be any constants. Set $\ell = (\log\bar{n})^{\alpha}$, $q = \bar{n}^{-(\delta\ell)/2}$, and $\Delta = (\log\bar{n})^{\zeta(1-\alpha)}$. Sample $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$, and sample $H \xleftarrow{\$} \mathrm{RP}^{\ell}_{q}(G)$. With all but $o(1)$ probability over the random choice of G and randomness of RP^{ℓ}_{q}, the following occurs. For any set S of $2^{o((\log\bar{n})^{2\alpha})}$ many ζ-cliques of H, define $T \subset S$ as*

$$T = \{K \in S : K \text{ has as a subgraph more than one } (\Delta, S)\text{-high degree } (\zeta-1)\text{-clique}\}.$$

Then $|T| = o(1)\cdot|S|$.

Proof. Consider any set S satisfying the condition in the lemma. By the pigeonhole principle, and the fact that each ζ-clique contains exactly ζ many $(\zeta-1)$-cliques, at most $\zeta|S|/\Delta$ many $(\zeta-1)$-cliques will be (Δ, S)-high degree. Denote this set of $(\zeta-1)$-cliques to be S'_{High}. We now bound the number of ζ-cliques $K \in S$ that contain at least two $(\zeta-1)$-cliques in S'_{High}.

Let $V_{\text{High}} \subset V_H$ be the set of all vertices v such that there exists $C \in S'_{\text{High}}$ with $v \in C$. Applying a naive upper bound, we have that $|V_{\text{High}}| \leq (\zeta-1)|S'_{\text{High}}|$, which is realized when all of the $(\zeta-1)$-cliques in S'_{High} are vertex-disjoint. Applying Corollary 6.5, we see that at most

$$O(\frac{|V_{\text{High}}|}{\zeta}\cdot(\log\bar{n})^{(\zeta-1)(1-\alpha)}) = O((|S|/\Delta)\cdot(\log\bar{n})^{(\zeta-1)(1-\alpha)}) = O(|S|/(\log\bar{n})^{1-\alpha})$$

many ζ-cliques can possibly span V_{High} (even ignoring whether they are in S). The remaining ζ-cliques in S must have at least one vertex outside of V_{High}. But if at least one vertex of a ζ-clique $K \in S$ is outside, then all but one of the $(\zeta-1)$-cliques $C \subset K$ will also have a vertex outside of V_{High}. Since V_{High} completely

covers all high degree $(\zeta-1)$-cliques, this implies that only one $(\zeta-1)$-cliques in K could be high degree.

Therefore the total number of all ζ-cliques in S that have more than one (Δ, S)-high degree $(\zeta-1)$-clique is $O(|S|/(\log \bar{n})^{1-\alpha}) = o(1) \cdot |S|$. □

7 Evidence for the Noisy k-LIN Conjecture

Here we discuss some lower bounds about the special case of k-XOR, i.e., k-LIN over $\mathbb{F}_2$. We refer readers to a recent work by Kocurek and Manohar [KM] for lower bounds for k-Lin over larger fields.

Computational Indistinguishability (Adapted from [ABW10]). We say that a distribution $\mathcal{D}$ over $\{0,1\}^m$ ε-fools a class $\mathcal{F}$ of boolean functions over $\{0,1\}^m$ if for every $f \in \mathcal{F}$ we have:

$$|\Pr[f(\mathcal{D}) = 1] - \Pr[f(U_m) = 1]| \leq \varepsilon.$$

Here U_m is the uniform distribution over $\mathbb{F}_2^m$. Let $\mathcal{D}_{\mathbf{A},(\log n)^{-\rho}}$ be the distribution induced by a noisy k-XOR sample, where $\mathbf{A} \in \mathbb{F}_2^{m \times n}$. We import the following theorem about the unconditional hardness of the noisy k-XOR conjecture from the work of Applebaum, Barak and Wigderson.

Lemma 7.1 (Theorem 9.1 in the full version of [ABW10]). *Let $\mathbf{A} \in \mathbb{F}_2^{m \times n}$ be any $(0.51, k, t)$-expanding matrix. Then $\mathcal{D}_{\mathbf{A},(\log n)^{-\rho}}$ satisfies the following:*

1. *0-fools t-wise tests. ($\mathcal{D}_{\mathbf{A},(\log n)^{-\rho}}$ is t-wise independent.)*
2. *$\delta = \frac{1}{2} \cdot (1 - 2(\log n)^{-\rho})^t$-fools linear tests. ($\mathcal{D}_{\mathbf{A},(\log n)^{-\rho}}$ is δ-biased.)*
3. *$8 \cdot (1 - 2(\log n)^{-\rho})^{t/2^{d-1}}$-fools degree d polynomials over $\mathbb{F}_2$.*

In this paper, we use the noisy k-XOR conjecture with $\rho = O(1)$, $k = \Omega(\log n)$, and $t = 2^{\log^\alpha n}$. Moreover, we achieve an expansion of, say, 0.99, which is significantly higher than the 0.51 threshold in the above theorem. Plugging these in, we can state the following corollary:

Corollary 7.2. *Let $\mathbf{A} \in \mathbb{F}_2^{m \times n}$ be a d-sparse matrix that is a $(0.51, \log n, 2^{\log^\alpha n})$-expander for some constant $\alpha < \frac{1}{2}$. Then $\mathcal{D}_{\mathbf{A},(\log n)^{-\rho}}$ satisfies the following:*

1. *0-fools any $2^{\log^\alpha n}$-wise test.*
2. *$2^{-\Omega(2^{\log^\alpha n})}$-fools linear tests.*
3. *$2^{-\Omega(2^{\log^\alpha n})}$-fools degree d polynomials over $\mathbb{F}_2$, for any $d \leq 0.99 \log^\alpha n$.*

Sum of Squares Lower Bounds. Grigoriev [Gri01] showed that a random instance of 3-XOR over n variables with $O(n)$ clauses cannot be refuted by the degree $o(n)$ sum of squares algorithm. This result was later rediscovered by Schoenebeck [Sch08], who also observed that it can be used to get lower bounds for random instances of k-XOR and k-SAT. In fact, these lower bounds work for any k-XOR instance that is sufficiently expanding. Barak and Steurer [BS] describe how to derive Grigoriev's result in this generalized sense, with their main lemma being the following.

Lemma 7.3 (Paraphrased from Lemma 9 in [BS]). *There exists some constant $\bar{c}$ such that the following is true. Let ϕ be any max 3-XOR instance whose clauses constitute a $(0.51, 3, t)$-expanding set system. Then the degree $\bar{c}t$ sum of squares algorithm fails to refute ϕ.*

In other words, even if only a 0.51 fraction of the clauses can be satisfied by any assignment of the variables, the degree $\bar{c}t$ sum of squares algorithm fails to show that we cannot satisfy all of the clauses simultaneously.

Barak and Steurer's proof proceeds independently of the XOR arity, so it directly translates to k-XOR for any (possibly superconstant) k. We can thus state the following corollary.

Corollary 7.4. *There exists some constant $\bar{c}$ such that the following is true. Let ϕ be any max k-XOR instance whose clauses constitute a $(0.51, k, t)$-expanding set system. Then the degree $\bar{c}t$ sum of squares algorithm fails to refute ϕ.*

While the above corollary only concerns refutation, not the problem of distinguishing between $(\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$ and $(\mathbf{A}, \$)$, it is natural to believe that it gives strong evidence for the security of the noisy k-XOR conjecture.

AC^0 Circuit Lower Bounds. We can also apply a result of Braverman regarding AC^0 circuits [Bra08]. We import the following theorem.

Lemma 7.5 (From [Bra08])**.** *There exists some function $r(m, \varepsilon, d) = (\log(m/\varepsilon))^{O(d^2)}$ such that the following holds. For all $d = O(1)$ and $m = \Omega(1)$, for all AC^0 circuits X of depth d and size m, and for all $r(m, d, \varepsilon)$-wise independent distributions $\mathcal{D}$ over Boolean vectors, we have that X is ε-fooled by $\mathcal{D}$.*

Rearranging values, we get the following corollary.

Corollary 7.6. *Any circuit of depth $d = O(1)$ and size $n^{O(1)}$ is $2^{-2^{\Omega(\log^{\alpha} n)}}$-fooled by $\mathcal{D}_{\mathbf{A},(\log n)^{-\rho}}$.*

8 Using the Search Version of Noisy k-LIN

8.1 PKE Scheme

This construction is very similar to the previous one. The only major change is in the decryption procedure.

Notations. Recall that we can interpret every node $v \in V_H$ as corresponding to a set U_v of ℓ many nodes in G, and interpret any edge $e \in E_H$ connecting nodes $(v, w) \in V_H$ as corresponding to a set of edges $W_e = K(U_v \cup U_w)$. Here $K(X)$ denotes the clique induces by vertices in the set X.

As before, for a subset of vertices $S \subseteq V_H$, define

$$G^V(S) \coloneqq \bigcup_{v \in S} U_v,$$

where U_i is as defined above, and for a subset of edges $T \subseteq E_H$, define

$$G^E(T) \coloneqq \bigcup_{e \in T} W_e,$$

where W_e is as defined above.

In other words $G^V(S)$ is the union of vertices in G for each ℓ-subset corresponding to a vertex of S and $G^E(T)$ is the union of edges in G for each clique in G corresponding to an edge of T.

Suggested Parameters. We use the same parameters as for the first PKE scheme. Let $\alpha, \delta \in (0, \frac{1}{2})$ be constants such that Conjecture 1.1 holds, and let $\gamma \in (\frac{1}{2}, 1)$ be a constant such that Conjecture 4.4 holds. (As noted before, if Conjecture 1.1 holds for some $\alpha \geq \frac{1}{2}$, this automatically implies that the conjecture holds for all $\alpha \in (0, \frac{1}{2})$.) Assuming the security parameter is λ,

- Size of initial Erdos-Renyi random graph: $\bar{n} = 2^{(\log \lambda)^{1/(1+\alpha)}}$.
- Planted clique size: $\bar{k} = \bar{n}^{\delta}$.
- Graph product parameter for RP: $\ell = (\log \bar{n})^{\alpha}$.
- Node deletion probability for RP: $q = \bar{n}^{-(\delta\ell)/2}$.
- Clique size (constant integer) for constructing k-LIN matrix: $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$.
- Exponent (constant) for block size in k-LIN matrix: $\beta = \zeta \cdot (\zeta(1-\alpha) + 2)$.
- Exponent (constant) for Bernoulli noise for k-LIN samples: $\rho = 2 + \beta\zeta$.

Key Generation and Encoding Scheme. The algorithms $\mathsf{Gen}(1^\lambda)$ and $\mathsf{Encode}(H, \zeta, \beta)$ are identical to those for the first PKE scheme.

Encryption.

$\star$ $\mathsf{ct} \leftarrow \mathsf{Enc}(1^\lambda, \mathsf{pk}, b \in \{0,1\})$:

1. Parse pk as $\mathbf{A}$. Let $\mathbf{A}$ have m rows and n columns.
2. Sample $\mathbf{e}$ as:
$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\rho}. \\ 0, \text{ otherwise.} \end{cases}$$
3. Sample $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$, and then sample $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^{n\lceil \log p \rceil}$ conditioned on $\tilde{\mathbf{s}} \cdot \mathbf{y} = b$, where $\tilde{\mathbf{s}}$ is any canonical bit representation of $\mathbf{s}$.
4. Output $\mathsf{ct} \leftarrow (\mathbf{y}, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e})$.

Decryption. Decryption in this case is significantly more involved than it was for the scheme based on decisional k-LIN. This is because to enable decryption, one needs to recover the *entire* secret $\mathbf{s}$ with $1 - o(1)$ probability. This is a strictly stronger requirement than before.

$\star$ $b' \leftarrow \mathsf{Dec}(1^\lambda, \mathsf{pk}, \mathsf{sk}, \mathsf{ct})$:

1. Parse pk as $\mathbf{A}$. Let $\mathbf{A}$ have m rows and n columns.
2. Parse sk as (H, K_H).
3. Parse ct as $(\mathbf{y}, \mathbf{b})$.
4. Let η be any (superconstant) integer satisfying $(\log n)^{\rho-1} \leq \binom{\eta}{\zeta} \leq 2(\log n)^{\rho-1}$.
5. Recompute all canonical labels C_j.
6. We will solve for a vector $\mathbf{s}' \in \mathbb{F}_p^n$ as follows: Interpret the $\mathbf{s}'$ as the concatenation of $c_{H,\zeta-1}$

many blocks, each of size exactly $\log^\beta m$. In other words, $(\mathbf{s}')^T = (\mathbf{b}_1 \| \cdots \| \mathbf{b}_{n/\log^\beta m})$, where $(\mathbf{s}')^T$ is the transpose of $\mathbf{s}'$. For each $j \in [n/\log^\beta m]$, we solve for $\mathbf{b}_j$ in the following manner.

(a) Let $X_j^{(\zeta-1)}$ be the set of vertices in H for the $(\zeta-1)$-clique labeled as C_j.

(b) Let $K^{(j)} = (\cap_{v \in X_j^{(\zeta-1)}} N_H(v)) \cap K_H$.

(c) Let $z \coloneqq \lfloor \frac{|K^{(j)}|}{\eta-(\zeta-1)} \rfloor$. Randomly partition $K^{(j)}$ into subsets $K_1^{(j)}, \ldots K_z^{(j)}$ of size exactly $\eta-(\zeta-1)$ and at most one subset $K_{\text{Leftover}}^{(j)}$ of size less than $\eta-(\zeta-1)$.

(d) For all $a \in [z]$, let $T_a = K_a^{(j)} \cup X_j^{(\zeta-1)}$.

(e) For all $a \in [z]$, let $\mathbf{x}_a \leftarrow \mathsf{Solve}(H, T_a, \mathbf{A}, \mathbf{b})$.

(f) Set $\mathbf{b}_j$ to be the majority vote of the corresponding blocks in the $\mathbf{x}_a$'s. If there is no majority vector for $\mathbf{b}_j$, set $\mathbf{b}_j$ to an arbitrary vector.

7. Output $b' \leftarrow \mathbf{y} \cdot \tilde{\mathbf{s}}'$, where $\tilde{\mathbf{s}}'$ is expressed in the same canonical bit representation as in the algorithm Enc.

Below is the algorithm to solve each system of linear equations.

$\star$ $\mathbf{x} \leftarrow \mathsf{Solve}(H, T, \mathbf{A}, \mathbf{b})$:

1. Let $\mathbf{A}$ have m rows and n columns.
2. Recompute all canonical labels R_i and C_j. Recompute all indicator variables $I_{i,j}$.
3. For all $\mathbf{A}_i$, define $X_i^{(\zeta)}$ as the subset of V_H corresponding to the ζ clique labelled as R_i. Define a set S of row labels of $\mathbf{A}$ as
$$S = \left\{ i : X_i^{(\zeta)} \subset T \right\}$$
4. Split every row of $\mathbf{A}$ into blocks of length $\log^\beta m$, i.e., $\mathbf{A}_i = (\mathbf{b}_{i,1} \| \cdots \| \mathbf{b}_{i,c_{H,\zeta-1}})$ where every $\mathbf{b}_{i,j}$ has width $\log^\beta m$. Define a set of block labels $S' \subset [c_{H,\zeta-1}]$ such that
$$S' = \{ j : \exists i \in S \text{ such that } I_{i,j} = 1 \}.$$
5. Let $\tilde{\mathbf{A}}$ be the submatrix of $\mathbf{A}$ with rows labeled by S and columns made by concatenating the blocks labeled by S'.
6. Solve the following system of linear equations using Gaussian elimination:
$$\tilde{\mathbf{A}}\mathbf{x} = \mathbf{b}|_S,$$
where $\mathbf{b}|_S$ is the vector $\mathbf{b}$ restricted to indices in S.
7. If the above step returns a valid solution, then output $\mathbf{x}$. Otherwise output $\mathbf{0}$.

8.2 Correctness

We first show that for all j, $K^{(j)}$ is very large. In particular, this applies even if the jth block of $\mathbf{s}'$ corresponds to a $(\zeta-1)$-clique that is *entirely outside* of K_H.

Lemma 8.1. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$ and $\beta > 0$, for all constant integers $\zeta > 1$, for all large enough λ, the following is true. Set $\bar{n} = 2^{(\log \lambda)^{1/(1+\alpha)}}$, $\bar{k} = \bar{n}^{\delta}$, $\ell = (\log \bar{n})^{\alpha}$, $q = \bar{n}^{-(\delta\ell)/2}$, and $\rho = 2 + \beta\zeta$. With $1 - 2^{-\bar{n}^{\Omega(1)}}$ probability over the random choice of G and the randomness of RP_q^{ℓ}, we have that $|K^{(j)}| > \bar{n}^{\delta\ell/3}$ for all j.*

Proof. Recall that $K^{(j)} = (\cap_{v \in X_j^{(\zeta-1)}} N_H(v)) \cap K_H$, where $X_j^{(\zeta-1)}$ is the set of vertices for the $(\zeta-1)$-clique in H labeled C_j. We will analyze each $K^{(j)}$ in terms of the original graph G.

Let the set of vertices for the planted clique in G be K_G. The vertices of $X_j^{(\zeta-1)}$ correspond to a set of at most $\ell \cdot (\zeta - 1)$ many vertices in G, namely $Y = G^V(X_j^{(\zeta-1)})$.

By definition of the randomized graph product, we have that $G^V(K^{(j)}) \subseteq K_G^{(j)}$, where $K_G^{(j)} = (\cap_{u \in Y} N_G(u)) \cap K_G$. Furthermore, during RP_q^{ℓ}, every ℓ-sized subset of $K_G^{(j)}$ becomes a vertex of $K^{(j)}$ with probability q.

Our first goal is to give a lower bound on $|K_G^{(j)}|$ that holds for all j with very high probability. To do this, we find it more convenient to consider a random set of vertices as the set Y, and then take a union bound over all possible sets of size up to $\ell \cdot (\zeta - 1)$.

Fix a random set $Y \subset V_G$, conditioned on $|Y| \leq \ell \cdot (\zeta - 1)$, and let $K_G^Y = (\cap_{u \in Y} N_G(u)) \cap K_G$. We have that $|K_G| = \bar{k} = \bar{n}^{\delta}$ by construction, and each vertex of K_G is in the neighborhood of all vertices in Y with probability at least $2^{-|Y|}$, over the random choice of G and the randomness of Y. (The true probability is slightly greater since some vertices of Y may happen to fall inside of the planted clique.) Furthermore, this event occurs independently for each vertex of K_G. We can thus lower bound the probability that $|K_G^Y| \geq \bar{n}^{0.9\delta}$ using a Chernoff bound. Since $\mathbb{E}[|K_G^Y|] \geq \bar{n}^{\delta} 2^{-|Y|} > 2\bar{n}^{0.9\delta}$ when $\bar{n}$ is sufficiently large and $\alpha \in (0, \frac{1}{2})$, we get

$$\begin{aligned}
\Pr\left[|K_G^Y| \geq \bar{n}^{0.9\delta}\right] &= 1 - \Pr\left[|K_G^Y| < \bar{n}^{0.9\delta}\right] \\
&\geq 1 - \Pr\left[|K_G^Y| < \frac{1}{2}\mathbb{E}[|K_G^Y|]\right] \\
&\geq 1 - 2^{-\bar{n}^{\Omega(1)}}
\end{aligned}$$

This lower bound also holds with $1 - 2^{-\bar{n}^{\Omega(1)}}$ probability after taking a union bound over all subsets $Y \subset V_G$ such that $|Y| \leq \ell \cdot (\zeta - 1)$, since there are less than $\bar{n}^{\ell \cdot (\zeta-1)}$ such subsets. So with probability $1 - 2^{-\bar{n}^{\Omega(1)}}$, we have that $|K_G^{(j)}| \geq \bar{n}^{0.9}$ holds for all j.

Fix a random choice of j. We now lower bound the number of ℓ-sized subsets of $K_G^{(j)}$ that were sampled by $\mathrm{RP}_q^{\ell}(G)$. The number of such subsets is $|K^{(j)}|$. We apply a Chernoff bound again, and use that

$\mathbb{E}[|K^{(j)}| \,\Big|\, |K_G^{(j)}| = \bar{n}^{0.9\delta}] > 2\bar{n}^{\ell\delta/3}$.

$$\begin{aligned}
\Pr\Big[|K^{(j)}| \geq \bar{n}^{\ell\delta/3}\Big] &\geq \Pr\Big[|K^{(j)}| \geq \bar{n}^{\ell\delta/3} \,\Big|\, K_G^{(j)} \geq \bar{n}^{0.9\delta}\Big] \cdot (1 - 2^{-\bar{n}^{\Omega(1)}}) \\
&\geq (1 - \Pr\Big[|K^{(j)}| < \frac{1}{2}\mathbb{E}[|K^{(j)}| \,\Big|\, |K_G^{(j)}| \geq \bar{n}^{0.9\delta}]\Big]) \cdot (1 - 2^{-\bar{n}^{\Omega(1)}}) \\
&> (1 - \Pr\Big[|K^{(j)}| < \frac{1}{2}\mathbb{E}[|K^{(j)}| \,\Big|\, |K_G^{(j)}| = \bar{n}^{0.9\delta}]\Big]) \cdot (1 - 2^{-\bar{n}^{\Omega(1)}}) \\
&\geq 1 - 2^{-\bar{n}^{\Omega(1)}}
\end{aligned}$$

Applying a union bound over all choices of j yields the lemma. □

We now show that decryption succeeds with very high probability.

Lemma 8.2. *For all constants $\alpha, \delta \in (0, \frac{1}{2})$ and $\beta > 0$, for all constant integers $\zeta > 1$, for all large enough λ, the following is true. Set $\bar{n} = 2^{(\log\lambda)^{1/(1+\alpha)}}$, $\bar{k} = \bar{n}^{\delta}$, $\ell = (\log\bar{n})^{\alpha}$, $q = \bar{n}^{-(\delta\ell)/2}$, and $\rho = 2 + \beta\zeta$. Then for all $b \in \{0,1\}$,*

$$\Pr_{(\mathsf{pk},\mathsf{sk})\leftarrow\mathsf{Gen}(1^{\bar{n}})}[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, b)) = b)] \geq 1 - o(1),$$

where the probability is over the coins of Encode, Gen, Enc, *and* Dec.

Proof. Assume that the event in Lemma 8.1 occurs. Consider a single block $\mathbf{b}_j$ of $\mathbf{s}'$. By Lemma 8.1, when solving for $\mathbf{b}_j$, we take the majority vote of the solutions to at least $\bar{n}^{\delta\ell/3-o(1)}$ many systems of linear equations. Denote this set of systems to be $\mathcal{E}$.

Each system $E_a \in \mathcal{E}$ is derived from a set $K_a^{(j)} \cup X_j^{(\zeta-1)}$ of vertices in H. Specifically, the equations are defined by the ζ-cliques in $K_a^{(j)} \cup X_j^{(\zeta-1)}$. Because every $K_a^{(j)}$ is vertex-disjoint, we have that every ζ-clique in $K_a^{(j)} \cup X_j^{(\zeta-1)}$ is distinct from every ζ-clique in $K_{a'}^{(j)} \cup X_j^{(\zeta-1)}$, for all $a \neq a'$. So for every $E_a, E_{a'} \in \mathcal{E}$ with $a \neq a'$, every equation in E_a is distinct from every equation in $E_{a'}$. (In fact, all of the $(\zeta-1)$-cliques are also distinct, except for the one labeled C_j, which is common across all $E_a \in \mathcal{E}$.)

Using that $K^{(j)}$ was partitioned randomly, we know that each $E_a \in \mathcal{E}$ will give the correct value for all variables in its system with probability $1 - o(1)$, via the same proof technique as in Lemma 5.3. In fact, the probability is only over the randomness of the error vector restricted the that specific set of equations. This means that every E_a gives the correct value for $\mathbf{b}_j$ independently with probability $1 - o(1)$. By a Chernoff bound, the probability that more than half of the systems give the correct value is more than $1 - 2^{-\bar{n}^{\Omega(1)}}$. If this occurs, then the majority vote must give the correct value of $\mathbf{b}_j$ regardless of the values given by the remaining systems.

The lemma follows by taking a union bound over all blocks of $\mathbf{s}'$, multiplying by the probability that the even in Lemma 8.1 occurs, and noting that recovering $\mathbf{s}' = \mathbf{s}$ implies $b' = b$. □

8.3 Security

Security follows from nearly the same hybrid argument as in Section 5, but with an additional appeal to the Goldreich-Levin Theorem [GL89].

Lemma 8.3 (Weak Security)**.** *Assume that Conjecture 1.1 holds for some constants $\alpha, \delta \in (0, \frac{1}{2})$, and assume that Conjecture 4.4 holds for some constant $\gamma \in (\frac{1}{2}, 1)$. Then for all large enough λ, the following*

is true. Set $\bar{n} = 2^{(\log\lambda)^{1/(1+\alpha)}}$, $\bar{k} = \bar{n}^{\delta}$, $\ell = (\log\bar{n})^{\alpha}$, $q = \bar{n}^{-(\delta\ell)/2}$, $\zeta = \lceil \frac{1}{1-\sqrt{\gamma}} \rceil + 1$, $\beta = \zeta \cdot (\zeta(1-\alpha)+2)$, *and* $\rho = 2 + \beta\zeta$. *For all poly*(λ) *size algorithms* $\mathcal{A}$,

$$\Big| \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 0)) = 1,\right] \\ - \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 1)) = 1\right] \Big| \leq 0.99,$$

where the probability is taken over the coins of Encode, Gen, Enc, *and* $\mathcal{A}$.

Proof. We will use standard hybrid arguments.

Hybrid $\mathcal{H}_0(1^\lambda)$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2}, \bar{k})$.
2. $H \xleftarrow{\$} \mathrm{RP}^{\ell}_{p}(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{A}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$ where
$$\mathsf{ct} \leftarrow (\mathbf{y}, \mathbf{A}\mathbf{s} + \mathbf{e})$$
Here $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$, and then we sample $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^{n\lceil \log p \rceil}$ conditioned on $\tilde{\mathbf{s}} \cdot \mathbf{y} = 0$. We sample $\mathbf{e}$ as:
$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\rho}. \\ 0, \text{ otherwise.} \end{cases}$$
7. Output $(\mathsf{pk}, \mathsf{ct})$.

Hybrid $\mathcal{H}_{0,\$}(1^\lambda)$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$.
2. $H \xleftarrow{\$} \mathrm{RP}^{\ell}_{p}(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{A}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$ where
$$\mathsf{ct} \leftarrow (\mathbf{y}, \mathbf{A}\mathbf{s} + \mathbf{e})$$

Here $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$, and then we sample $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^{n\lceil \log p \rceil}$ conditioned on $\tilde{\mathbf{s}} \cdot \mathbf{y} = 0$. We sample $\mathbf{e}$ as:

$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\rho}. \\ 0, \text{ otherwise.} \end{cases}$$

7. Output $(\mathsf{pk}, \mathsf{ct})$.

Hybrid $\mathcal{H}_{1,\$}(1^\lambda)$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2})$.
2. $H \xleftarrow{\$} \mathrm{RP}_p^\ell(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{A}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$ where
$$\mathsf{ct} \leftarrow (\mathbf{y}, \mathbf{A}\mathbf{s} + \mathbf{e})$$
Here $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$, and then we sample $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^{n\lceil \log p \rceil}$ conditioned on $\tilde{\mathbf{s}} \cdot \mathbf{y} = 1$. We sample $\mathbf{e}$ as:
$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\rho}. \\ 0, \text{ otherwise.} \end{cases}$$
7. Output $(\mathsf{pk}, \mathsf{ct})$.

Hybrid $\mathcal{H}_1(1^\lambda)$:

1. $G \xleftarrow{\$} \mathcal{G}(\bar{n}, \frac{1}{2}, \bar{k})$.
2. $H \xleftarrow{\$} \mathrm{RP}_p^\ell(G)$.
3. $\mathbf{S} \leftarrow \mathsf{Encode}(H, \zeta, \beta)$. Let $\mathbf{A}$ have m rows and n columns.
4. Let $\mathbf{A} \in \mathbb{F}_p^{m \times n}$ be the matrix where we replace each nonzero entry in $\mathbf{S}$ with a random element in $\mathbb{F}_p^*$.
5. $\mathsf{pk} \coloneqq \mathbf{A}$.
6. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$ where
$$\mathsf{ct} \leftarrow (\mathbf{y}, \mathbf{A}\mathbf{s} + \mathbf{e})$$

Here $\mathbf{s} \xleftarrow{\$} \mathbb{F}_p^n$, and then we sample $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^{n\lceil \log p \rceil}$ conditioned on $\tilde{\mathbf{s}} \cdot \mathbf{y} = 1$. We sample $\mathbf{e}$ as:

$$\mathbf{e}_i = \begin{cases} \text{Random element of } \mathbb{F}_p^*, \text{ with probability } \eta = (\log n)^{-\rho}. \\ 0, \text{ otherwise.} \end{cases}$$

7. Output $(\mathsf{pk}, \mathsf{ct})$.

As before, before analyzing the hybrids, note that $\lambda = \bar{n}^{\Theta(\ell)}$ and $\lambda = n^{\Theta(1)}$ with very high probability.

We first argue that $\mathcal{H}_0 \approx_{\frac{1}{3}} \mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$} \approx_{\frac{1}{3}} \mathcal{H}_1$, which follows from the planted clique conjecture.

Claim 8.4 ($\mathcal{H}_0 \approx_{\frac{1}{3}} \mathcal{H}_{0,\$}$)**.** *Assume that Conjecture 1.1 holds for some constants $\alpha, \delta \in (0, \frac{1}{2})$. For all sufficiently large λ, and for all constant $\gamma \in (\frac{1}{2}, 1)$, the following is true. Set the parameters as in the lemma statement. For all poly(λ)-size algorithms $\mathcal{A}$,*

$$\left|\Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_0(1^\lambda)) = 1\right] - \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{0,\$}(1^\lambda)) = 1\right]\right| \leq \frac{1}{3}.$$

Proof. Identical to the proof of Claim 5.9. □

Claim 8.5 ($\mathcal{H}_{1,\$} \approx_{\frac{1}{3}} \mathcal{H}_1$)**.** *Assume that Conjecture 1.1 holds for some constants $\alpha, \delta \in (0, \frac{1}{2})$. For all sufficiently large λ, and for all constant $\gamma \in (\frac{1}{2}, 1)$, the following is true. Set the parameters as in the lemma statement. For all poly(λ)-size algorithms $\mathcal{A}$,*

$$\left|\Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{1,\$}(1^\lambda)) = 1\right] - \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_1(1^\lambda)) = 1\right]\right| \leq \frac{1}{3}.$$

Proof. Identical to the proof of Claim 5.11. □

We now argue that $\mathcal{H}_{0,\$} \approx_{0.30} \mathcal{H}_{1,\$}$ via an appeal to the Goldreich-Levin Theorem [GL89]. We first state a version of this theorem below:

Claim 8.6 (Adapted from Theorem 3 in [Bel99])**.** *Suppose we have an unknown vector $\mathbf{s} \in \mathbb{F}_2^n$, but we know the value of $f(\mathbf{s})$ for some function f. Suppose further that we have a poly(n)-size algorithm $\mathcal{Q}$ satisfying*

$$\Pr_{\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^n}[\mathcal{Q}(\mathbf{y}, f(\mathbf{s})) = \mathbf{y} \cdot \mathbf{s}] \geq 0.51,$$

and a poly(n)-size algorithm $\mathcal{Y}$ satisfying

$$\mathcal{Y}(\mathbf{s}', f(\mathbf{s})) = 1 \text{ if and only if } \mathbf{s}' = \mathbf{s}.$$

Then there exists a poly(n)-size algorithm $\mathcal{Z}$ such that

$$\Pr[\mathcal{Z}(f(\mathbf{s})) = \mathbf{s}] = 1 - o(1).$$

Claim 8.7 ($\mathcal{H}_{0,\$} \approx_{0.30} \mathcal{H}_{1,\$}$)**.** *Assume that Conjecture 4.4 holds for some constant $\gamma \in (\frac{1}{2}, 1)$. For all sufficiently large λ, and for all constants $\alpha, \delta \in (0, \frac{1}{2})$, the following is true. Set the parameters as in the lemma statement. For all poly(λ)-size algorithms $\mathcal{A}$,*

$$\left|\Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{0,\$}(1^\lambda)) = 1\right] - \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{1,\$}(1^\lambda)) = 1\right]\right| \leq 0.30.$$

Proof. Towards a proof by contradiction, suppose there exists a poly(λ)-size algorithm $\mathcal{A}$ to differentiate between $\mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$}$ with advantage more than 0.30. By randomly sampling λ many choices of $(\mathbf{A}, \mathbf{s}, \mathbf{e})$ as generated in hybrids $\mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$}$, and then executing $\mathcal{A}$ on these, we can approximate

$$p_0 \approx \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{0,\$}(1^\lambda)) = 1\right] \text{ and}$$

$$p_1 \approx \Pr\left[\mathcal{A}(1^\lambda, \mathcal{H}_{1,\$}(1^\lambda)) = 1\right]$$

to within an error of $\pm o(1)$, with probability $1 - o(1)$.

Suppose the above procedure indicates that $p_1 > p_0$. Then we define an algorithm $\mathcal{B}$ which (i) takes as input a tuple $(\mathbf{y}, \mathbf{A}, \mathbf{As} + \mathbf{e})$, (ii) gives the tuple as input to algorithm $\mathcal{A}$, and (iii) outputs 1 if $\mathcal{A}$ outputs 1, and outputs 0 otherwise. If instead the procedure indicates that $p_1 < p_0$, we define $\mathcal{B}$ identically, except it outputs 1 if $\mathcal{A}$ does *not* output 1, and it outputs 0 otherwise. Assuming the estimates for p_0 and p_1 are accurate to within $\pm o(1)$, we have that

$$\Pr_{(\mathbf{y},\mathbf{A},\mathbf{s},\mathbf{e}) \text{ is sampled as in } \mathcal{H}_{1,\$}}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{As} + \mathbf{e}) = 1] - \Pr_{(\mathbf{y},\mathbf{A},\mathbf{s},\mathbf{e}) \text{ is sampled as in } \mathcal{H}_{0,\$}}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{As} + \mathbf{e}) = 1] > 0.30. \tag{1}$$

This ensures that, with high probability, $\mathcal{B}$ is more likely to return 1 when the dot product $\mathbf{y} \cdot \tilde{\mathbf{s}}$ does indeed equal 1 (the original algorithm $\mathcal{A}$ may have the reverse property).

We now perform a probabilistic analysis of $\mathcal{B}$. Define

$$f(\mathbf{A}, \mathbf{s}, \mathbf{e}) = \Pr_{\mathbf{y} \xleftarrow{\$} \mathbb{F}_p^n \text{ conditioned on } \tilde{\mathbf{s}} \cdot \mathbf{y} = 1}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{As} + \mathbf{e}) = 1] - \Pr_{\mathbf{y} \xleftarrow{\$} \mathbb{F}_p^n \text{ conditioned on } \tilde{\mathbf{s}} \cdot \mathbf{y} = 0}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{As} + \mathbf{e}) = 1].$$

Note that $f(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in [-1, 1]$ for all $(\mathbf{A}, \mathbf{s}, \mathbf{e})$. The sampling procedure for $(\mathbf{A}, \mathbf{s}, \mathbf{e})$ is the same in both hybrids, so we can rewrite (1) as

$$\sum_{(\mathbf{A},\mathbf{s},\mathbf{e})} \Pr\left[(\mathbf{A}, \mathbf{s}, \mathbf{e}) \text{ is sampled in } \mathcal{H}_{1,\$}\right] \cdot f(\mathbf{A}, \mathbf{s}, \mathbf{e}) > 0.30. \tag{2}$$

Let $\mathcal{T}$ be the set of all possible $(\mathbf{A}, \mathbf{s}, \mathbf{e})$, and let $\mathcal{T}_{\geq \varepsilon} \subseteq \mathcal{T}$ be the set of all $(\mathbf{A}, \mathbf{s}, \mathbf{e})$ such that $f(\mathbf{A}, \mathbf{s}, \mathbf{e}) \geq \varepsilon$. For all $(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in \mathcal{T} \backslash \mathcal{T}_{\geq \varepsilon}$, we have the trivial upper bound that $f(\mathbf{A}, \mathbf{s}, \mathbf{e}) < \varepsilon$. For all $(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in \mathcal{T}_{\geq \varepsilon}$, we have the trivial upper bound that $f(\mathbf{A}, \mathbf{s}, \mathbf{e}) \leq 1$. This combined with (2) implies that

$$\sum_{(\mathbf{A},\mathbf{s},\mathbf{e}) \in \mathcal{T}_{\geq \varepsilon}} \Pr\left[(\mathbf{A}, \mathbf{s}, \mathbf{e}) \text{ is sampled in } \mathcal{H}_{1,\$}\right] + \varepsilon \cdot \sum_{(\mathbf{A},\mathbf{s},\mathbf{e}) \in \mathcal{T} \backslash \mathcal{T}_{\geq \varepsilon}} \Pr\left[(\mathbf{A}, \mathbf{s}, \mathbf{e}) \text{ is sampled in } \mathcal{H}_{1,\$}\right] > 0.30,$$

and hence,

$$\sum_{(\mathbf{A},\mathbf{s},\mathbf{e}) \in \mathcal{T}_{\geq \varepsilon}} \Pr\left[(\mathbf{A}, \mathbf{s}, \mathbf{e}) \text{ is sampled in } \mathcal{H}_{1,\$}\right] > 0.30 - \varepsilon.$$

Expressed equivalently, we have

$$\Pr_{(\mathbf{A},\mathbf{s},\mathbf{e}) \text{ is sampled as in } \mathcal{H}_{1,\$}}[(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in \mathcal{T}_{\geq \varepsilon}] > 0.30 - \varepsilon.$$

This is useful because, for all $(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in \mathcal{T}_{\geq\varepsilon}$, by the definition of $\mathcal{T}_{\geq\varepsilon}$, we have the *lower* bound that

$$\Pr_{\mathbf{y} \xleftarrow{\$} \mathbb{F}_p^n \text{ conditioned on } \tilde{\mathbf{s}}\cdot\mathbf{y}=1}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 1] - \Pr_{\mathbf{y} \xleftarrow{\$} \mathbb{F}_p^n \text{ conditioned on } \tilde{\mathbf{s}}\cdot\mathbf{y}=0}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 1] \geq \varepsilon. \quad (3)$$

This analysis actually shows that we can interpret $\mathcal{B}$ as a predictor for the dot product $\mathbf{y} \cdot \mathbf{s}$. (This will serve as algorithm $\mathcal{Q}$ when we apply the Goldreich-Levin Theorem.) To see this, first let

$$p_{\text{avg}} = \frac{1}{2} \cdot \left(\Pr_{\mathbf{y} \xleftarrow{\$} \mathbb{F}_p^n \text{ conditioned on } \tilde{\mathbf{s}}\cdot\mathbf{y}=1}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 1] + \Pr_{\mathbf{y} \xleftarrow{\$} \mathbb{F}_p^n \text{ conditioned on } \tilde{\mathbf{s}}\cdot\mathbf{y}=0}[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 1] \right). \quad (4)$$

Now we analyze the behavior of $\mathcal{B}$ on $(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$, where $\mathbf{y}$ is a random vector $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^n$ and we assume $(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in \mathcal{T}_{\geq\varepsilon}$. With probability $\frac{1}{2}$ over the choice of $\mathbf{y}$, we have that $\mathbf{y} \cdot \mathbf{s} = 1$, and in this case

$$\Pr[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 1 = \mathbf{y} \cdot \tilde{\mathbf{s}}] \geq p_{\text{avg}} + \frac{\varepsilon}{2}.$$

With the remaining probability $\frac{1}{2}$ over the choice of $\mathbf{y}$, we have $\mathbf{y} \cdot \tilde{\mathbf{s}} = 0$, and in this case

$$\Pr[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 1] \leq p_{\text{avg}} - \frac{\varepsilon}{2}.$$

which implies

$$\Pr[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 0 = \mathbf{y} \cdot \mathbf{s}] \geq 1 - p_{\text{avg}} + \frac{\varepsilon}{2}.$$

Accumulating these probabilities, we have the following when $\mathbf{y} \xleftarrow{\$} \mathbb{F}_2^n$ and $(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in \mathcal{T}_{\geq\varepsilon}$:

$$\Pr[\mathcal{B}(\mathbf{y}, \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = \mathbf{y} \cdot \mathbf{s}] \geq \frac{1}{2} \cdot (p_{\text{avg}} + \frac{\varepsilon}{2}) + \frac{1}{2} \cdot (1 - p_{\text{avg}} + \frac{\varepsilon}{2}) = \frac{1}{2} + \frac{\varepsilon}{2}.$$

If we take $\varepsilon = 0.04$, then the predictor has accuracy $0.52 > 0.51$.

We can also define a poly(λ)-size algorithm $\mathcal{D}(\mathbf{s}', \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$ such that, with $1 - o(1)$ probability over the choice of $(\mathbf{A}, \mathbf{s}, \mathbf{e})$ as produced in $\mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$}$, we have that $\mathcal{D}(\mathbf{s}', \mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e}) = 1$ if and only if $\mathbf{s}' = \mathbf{s}$. (Note that we can freely convert between the bit representation $\tilde{\mathbf{s}'}$ and the original vector $\mathbf{s}'$.) This will serve as algorithm $\mathcal{Y}$ when we apply the Goldreich-Levin Theorem. The algorithm will simply compute $\mathbf{b}' = \mathbf{A}\mathbf{s}'$, determine the Hamming distance between $\mathbf{b}'$ and $\mathbf{b} = \mathbf{A}\mathbf{s}+\mathbf{e}$, and then return either 0 or 1 based on the magnitude of the Hamming distance. With $1 - o(1)$ probability over the choice of $(\mathbf{A}, \mathbf{s}, \mathbf{e})$ as produced in $\mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$}$, if we set $\mathbf{s}' = \mathbf{s}$, the Hamming distance will be significantly smaller than for all other choices of $\mathbf{s}'$. (This property is folklore. Roughly speaking, it follows because, when ζ is a constant, every column of $\mathbf{A}$ will have polynomial Hamming weight with very high probability.) We can determine an appropriate Hamming distance cutoff in polynomial time.

Now we apply Claim 8.6 to get a poly(λ)-size algorithm $\mathcal{X}$ that recovers $\mathbf{s}$ from $f(\mathbf{s})$, where $f(\mathbf{s}) = (\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$. The algorithm follows by simply applying the claim with $\mathcal{Q} := \mathcal{B}$, $\mathcal{Y} := \mathcal{D}$, and $\mathcal{X} := \mathcal{Z}$.

Assuming that $(\mathbf{A}, \mathbf{s}, \mathbf{e}) \in \mathcal{T}_{\geq 0.04}$, that we can construct the algorithm $\mathcal{D}$, and that our estimates for p_0, p_1 are accurate to within $\pm o(1)$, the algorithm $\mathcal{X}$ will recover the vector $\mathbf{s}$ with $1 - o(1)$ probability. Applying the probability bounds we derived above, we see that $\mathcal{X}$ recovers $\mathbf{s}$ from $(\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$ with probability

$0.26 - o(1)$ when $(\mathbf{A}, \mathbf{s}, \mathbf{e})$ is sampled as in $\mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$}$.

As in the proof of Claim 5.10, the matrix $\mathbf{A}$ will satisfy the preconditions of Conjecture 4.4 with $1 - o(1)$ probability. Furthermore, the sampling procedure for $\mathbf{s}$ and $\mathbf{e}$ is the same in the statement of Conjecture 4.4 as in the hybrids $\mathcal{H}_{0,\$}$ and $\mathcal{H}_{1,\$}$. Combining this with the above analysis of algorithm $\mathcal{X}$, we see that $\mathcal{X}$ is an algorithm for the search problem in Conjecture 4.4 that succeeds with probability at least $0.26 - o(1) > \frac{1}{4}$, which is a contradiction. □

Combining Claims 8.4, 8.5, and 8.7, and accounting for the fact that $\lambda = \bar{n}^{\Theta(\ell)}$ and $\lambda = n^{\Theta(1)}$ with very high probability, we get

$$\Big| \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 0)) = 1,\right] - \Pr\left[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda); \mathcal{A}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, 1)) = 1\right] \Big| \leq \frac{29}{30} + o(1) < 0.99.$$

□

Theorem 8.8. *Assuming Conjecture 1.1 and Conjecture 4.4 both hold, there exists a semantically secure public key encryption scheme.*

Proof. Identical to the proof of Theorem 5.12. □

9 References

[AAK+07] Noga Alon, Alexandr Andoni, Tali Kaufman, Kevin Matulef, Ronitt Rubinfeld, and Ning Xie. Testing k-wise and almost k-wise independence. In David S. Johnson and Uriel Feige, editors, *39th ACM STOC*, pages 496–505. ACM Press, June 2007. [2, 3, 11]

[ABBG11] Sanjeev Arora, Boaz Barak, Markus Brunnermeier, and Rong Ge. Computational complexity and information asymmetry in financial products. *Communications of the ACM*, 54(5):101–107, 2011. [2, 3, 11]

[ABI+23] Damiano Abram, Amos Beimel, Yuval Ishai, Eyal Kushilevitz, and Varun Narayanan. Cryptography from planted graphs: Security with logarithmic-size messages. In Guy N. Rothblum and Hoeteck Wee, editors, *TCC 2023, Part I*, volume 14369 of *LNCS*, pages 286–315. Springer, Cham, November / December 2023. [2, 3, 6, 11]

[ABR12] Benny Applebaum, Andrej Bogdanov, and Alon Rosen. A dichotomy for local small-bias generators. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 600–617. Springer, Berlin, Heidelberg, March 2012. [6]

[ABW10] Benny Applebaum, Boaz Barak, and Avi Wigderson. Public-key cryptography from different assumptions. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 171–180, 2010. [2, 3, 4, 5, 6, 7, 10, 11, 30]

[AK19] Benny Applebaum and Eliran Kachlon. Sampling graphs without forbidden subgraphs and unbalanced expanders with negligible error. In David Zuckerman, editor, *60th FOCS*, pages 171–179. IEEE Computer Society Press, November 2019. [6]

[AKS98] Noga Alon, Michael Krivelevich, and Benny Sudakov. Finding a large hidden clique in a random graph. *Random Structures & Algorithms*, 13(3-4):457–466, 1998. [2, 3, 11]

[AL16] Benny Applebaum and Shachar Lovett. Algebraic attacks against random local functions and their countermeasures. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 1087–1100, 2016. [6, 12]

[Ale03] Michael Alekhnovich. More on average case vs approximation complexity. In *44th FOCS*, pages 298–307. IEEE Computer Society Press, October 2003. [2, 3, 5]

[AOW15] Sarah R Allen, Ryan O'Donnell, and David Witmer. How to refute a random csp. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 689–708. IEEE, 2015. [3, 4]

[App12] Benny Applebaum. Pseudorandom generators with long stretch and low locality from random local one-way functions. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 805–816, 2012. [6, 12]

[App16] Benny Applebaum. Cryptographic hardness of random local functions: Survey. *Computational complexity*, 25:667–722, 2016. [3, 4, 11, 12]

[BB20] Matthew Brennan and Guy Bresler. Reducibility and statistical-computational gaps from secret leakage. In *Conference on Learning Theory*, pages 648–847. PMLR, 2020. [2, 3, 11]

[BBH18] Matthew Brennan, Guy Bresler, and Wasim Huleihel. Reducibility and computational lower bounds for problems with planted sparse structure. In *Conference On Learning Theory*, pages 48–166. PMLR, 2018. [2, 3, 11]

[Bel99] Mihir Bellare. The goldreich-levin theorem. 1999. [38]

[BFKL94] Avrim Blum, Merrick L. Furst, Michael J. Kearns, and Richard J. Lipton. Cryptographic primitives based on hard learning problems. In Douglas R. Stinson, editor, *CRYPTO'93*, volume 773 of *LNCS*, pages 278–291. Springer, Berlin, Heidelberg, August 1994. [2]

[BHK+19] Boaz Barak, Samuel Hopkins, Jonathan Kelner, Pravesh K Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. *SIAM Journal on Computing*, 48(2):687–735, 2019. [2, 3]

[BKR23] Andrej Bogdanov, Pravesh K. Kothari, and Alon Rosen. Public-key encryption, local pseudorandom generators, and the low-degree method. In Guy N. Rothblum and Hoeteck Wee, editors, *TCC 2023, Part I*, volume 14369 of *LNCS*, pages 268–285. Springer, Cham, November / December 2023. [6]

[BM16] Boaz Barak and Ankur Moitra. Noisy tensor completion via the sum-of-squares hierarchy. In *Conference on Learning Theory*, pages 417–445. PMLR, 2016. [3, 4, 11]

[BQ09] Andrej Bogdanov and Youming Qiao. On the security of goldreich's one-way function. In Irit Dinur, Klaus Jansen, Joseph Naor, and José D. P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 12th International Workshop, APPROX 2009, and 13th International Workshop, RANDOM 2009, Berkeley, CA, USA, August 21-23, 2009. Proceedings*, volume 5687 of *Lecture Notes in Computer Science*, pages 392–405. Springer, 2009. [6]

[BR13] Quentin Berthet and Philippe Rigollet. Complexity theoretic lower bounds for sparse principal component detection. In *Conference on learning theory*, pages 1046–1066. PMLR, 2013. [2, 3, 11]

[Bra08] Mark Braverman. Polylogarithmic independence fools ac 0 circuits. *Journal of the ACM (JACM)*, 57(5):1–10, 2008. [31]

[BS] Boaz Barak and David Steurer. Higher-degree integrality gaps: from computational hardness to limitations of sum-of-squares. *Sum-of-squares: proofs, beliefs, and algorithms*. [30]

[BS92] Piotr Berman and Georg Schnitger. On the complexity of approximating the independent set problem. *Information and Computation*, 96(1):77–94, 1992. [8, 11]

[BSV19] Andrej Bogdanov, Manuel Sabin, and Prashant Nalini Vasudevan. Xor codes and sparse learning parity with noise. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 986–1004. SIAM, 2019. [3, 4, 11, 12]

[CDM+18] Geoffroy Couteau, Aurélien Dupin, Pierrick Méaux, Mélissa Rossi, and Yann Rotella. On the concrete security of Goldreich's pseudorandom generator. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 96–124. Springer, Cham, December 2018. [6]

[CEMT09] James Cook, Omid Etesami, Rachel Miller, and Luca Trevisan. Goldreich's one-way function candidate and myopic backtracking algorithms. In Omer Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 521–538. Springer, Berlin, Heidelberg, March 2009. [6]

[CJ24] Caicai Chen and Chris Jones. Key-recovery attack on a public-key encryption related to planted clique. *Cryptology ePrint Archive*, 2024. [6]

[CM01] Mary Cryan and Peter Bro Miltersen. On pseudorandom generators in NC. In Jirí Sgall, Ales Pultr, and Petr Kolman, editors, *Mathematical Foundations of Computer Science 2001, 26th International Symposium, MFCS 2001 Marianske Lazne, Czech Republic, August 27-31, 2001, Proceedings*, volume 2136 of *Lecture Notes in Computer Science*, pages 272–284. Springer, 2001. [6]

[CMZ23] Zongchen Chen, Elchanan Mossel, and Ilias Zadik. Almost-linear planted cliques elude the metropolis process. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4504–4539. SIAM, 2023. [3]

[DGGP14] Yael Dekel, Ori Gurel-Gurevich, and Yuval Peres. Finding hidden cliques in linear time with high probability. *Combinatorics, Probability and Computing*, 23(1):29–49, 2014. [3]

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976. [2]

[DIJL23] Quang Dao, Yuval Ishai, Aayush Jain, and Huijia Lin. Multi-party homomorphic secret sharing and sublinear MPC from sparse LPN. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part II*, volume 14082 of *LNCS*, pages 315–348. Springer, Cham, August 2023. [3]

[DM15] Yash Deshpande and Andrea Montanari. Finding hidden cliques of size n/e n/e in nearly linear time. *Foundations of Computational Mathematics*, 15:1069–1128, 2015. [3]

[ERSY22] Reyad Abed Elrazik, Robert Robere, Assaf Schuster, and Gal Yehuda. Pseudorandom self-reductions for NP-complete problems. In Mark Braverman, editor, *ITCS 2022*, volume 215, pages 65:1–65:12. LIPIcs, January / February 2022. [2, 3, 11]

[FGR+17] Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh S Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. *Journal of the ACM (JACM)*, 64(2):1–37, 2017. [3]

[FK03] Uriel Feige and Robert Krauthgamer. The probable value of the lovász–schrijver relaxations for maximum independent set. *SIAM Journal on Computing*, 32(2):345–370, 2003. [3]

[FKO06] Uriel Feige, Jeong Han Kim, and Eran Ofek. Witnesses for non-satisfiability of dense random 3CNF formulas. In *47th FOCS*, pages 497–508. IEEE Computer Society Press, October 2006. [3, 4, 5, 11]

[FPV15] Vitaly Feldman, Will Perkins, and Santosh Vempala. On the complexity of random satisfiability problems with planted solutions. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *47th ACM STOC*, pages 77–86. ACM Press, June 2015. [3, 4, 11]

[FR10] Uriel Feige and Dorit Ron. Finding hidden cliques in linear time. *Discrete Mathematics & Theoretical Computer Science*, (Proceedings), 2010. [3]

[GL89] Oded Goldreich and Leonid A Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 25–32, 1989. [35, 38]

[Gol00] Oded Goldreich. Candidate one-way functions based on expander graphs. *Electron. Colloquium Comput. Complex.*, TR00-090, 2000. [6]

[Gol11] Oded Goldreich. Candidate one-way functions based on expander graphs. *Studies in Complexity and Cryptography*, pages 76–87, 2011. [12]

[Gri01] Dima Grigoriev. Linear lower bound on degrees of positivstellensatz calculus proofs for the parity. *Theoretical Computer Science*, 259(1-2):613–622, 2001. [3, 4, 11, 30]

[HK11] Elad Hazan and Robert Krauthgamer. How hard is it to approximate the best nash equilibrium? *SIAM Journal on Computing*, 40(1):79–91, 2011. [2, 3, 11]

[Hop18] Samuel Hopkins. *Statistical inference and the sum of squares method*. PhD thesis, Cornell University, 2018. [2]

[HR05] Thomas Holenstein and Renato Renner. One-way secret-key agreement and applications to circuit polarization and immunization of public-key encryption. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 478–493. Springer, Berlin, Heidelberg, August 2005. [10]

[HS24] Shuichi Hirahara and Nobutaka Shimizu. Planted clique conjectures are equivalent. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *56th ACM STOC*, pages 358–366. ACM Press, June 2024. [2, 3, 8, 11]

[Hud16] Péter Hudoba. Public key cryptography based on the clique and learning parity with noise problems for post-quantum cryptography. In *Proceedings of the 11th Joint Conference on Mathematics and Computer Science*, pages 102–112, 2016. [6]

[HWX15] Bruce Hajek, Yihong Wu, and Jiaming Xu. Computational lower bounds for community detection on random graphs. In *Conference on Learning Theory*, pages 899–928. PMLR, 2015. [3, 11]

[IKOS08] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Cryptography with constant computational overhead. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 433–442. ACM Press, May 2008. [3, 6]

[Jer92] Mark Jerrum. Large cliques elude the metropolis process. *Random Structures & Algorithms*, 3(4):347–359, 1992. [2, 3, 11]

[JP00] Ari Juels and Marcus Peinado. Hiding cliques for cryptographic security. *Designs, Codes and Cryptography*, 20(3):269–280, 2000. [2, 3, 11]

[Kar76] Richard Karp. Probabilistic analysis of some combinatorial search problems. *Algorithms and Complexity: New Directions and Recent Results*, 1976. [2, 3, 11]

[Kat87] Gyula Katona. A theorem of finite sets. *Classic Papers in Combinatorics*, pages 381–401, 1987. [20, 21]

[Kea98] Michael Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM (JACM)*, 45(6):983–1006, 1998. [2]

[KM] Nicholas Kocurek and Peter Manohar. Strong refutation of semirandom k-lin over larger fields. 30

[KMOW17] Pravesh K. Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer. Sum of squares lower bounds for refuting any CSP. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *49th ACM STOC*, pages 132–145. ACM Press, June 2017. 3, 4, 6, 11

[Kru63] Joseph B Kruskal. The number of simplices in a complex. *Mathematical optimization techniques*, 10:251–278, 1963. 20, 21

[Kuč95] Luděk Kučera. Expected complexity of graph partitioning problems. *Discrete Applied Mathematics*, 57(2-3):193–212, 1995. 2, 3, 11

[KZ14] Pascal Koiran and Anastasios Zouzias. Hidden cliques and the certification of the restricted isometry property. *IEEE transactions on information theory*, 60(8):4999–5006, 2014. 3, 11

[Las01] Jean B Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on optimization*, 11(3):796–817, 2001. 2

[MRS20] Pasin Manurangsi, Aviad Rubinstein, and Tselil Schramm. The strongish planted clique hypothesis and its consequences. *arXiv preprint arXiv:2011.05555*, 2020. 2, 3, 8, 11

[MST03] Elchanan Mossel, Amir Shpilka, and Luca Trevisan. On e-biased generators in NC0. In *44th FOCS*, pages 136–145. IEEE Computer Society Press, October 2003. 6

[MW15] Zongming Ma and Yihong Wu. Computational barriers in minimax submatrix detection. 2015. 3, 11

[Nes00] Yurii Nesterov. *Squared Functional Systems and Optimization Problems*, pages 405–440. Springer US, Boston, MA, 2000. 2

[Nis91] Noam Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica*, 11:63–70, 1991. 5

[OST19] Igor C Oliveira, Rahul Santhanam, and Roei Tell. Expander-based cryptography meets natural proofs. *10th Innovations in Theoretical Computer Science*, 2019. 5

[OW14] Ryan O'Donnell and David Witmer. Goldreich's PRG: evidence for near-optimal polynomial stretch. In *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014*, pages 1–12. IEEE Computer Society, 2014. 6

[Par00] Pablo A Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. California Institute of Technology, 2000. 2

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005. 2

[Ros08] Benjamin Rossman. On the constant-depth complexity of k-clique. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 721–730. ACM Press, May 2008. 3

[RRS17] Prasad Raghavendra, Satish Rao, and Tselil Schramm. Strongly refuting random CSPs below the spectral threshold. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *49th ACM STOC*, pages 121–131. ACM Press, June 2017. 3, 4, 11

[RSA78] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978. [2]

[Sch08] Grant Schoenebeck. Linear level lasserre lower bounds for certain k-csps. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 593–602. IEEE, 2008. [3, 4, 11, 30]

[Sho85] Naum Z. Shor. Minimization methods for non-differentiable functions and applications. *Cybernetics*, 13(1):94–96, 1985. [2]