Electronic Colloquium on Computational Complexity, Report No. 170 (2021)

Pseudorandom Self-Reductions for NP-Complete Problems

Reyad Abed Elrazik[1] Robert Robere[2] Assaf Schuster[3] Gal Yehuda[4]

November 25, 2021

Abstract

A language L is *random-self-reducible* if deciding membership in L can be reduced (in polynomial time) to deciding membership in L for uniformly random instances. It is known that several "number theoretic" languages (such as computing the permanent of a matrix) admit random self-reductions. Feigenbaum and Fortnow showed that NP-complete languages are not non-adaptively random-self-reducible unless the polynomial-time hierarchy collapses, giving suggestive evidence that NP may not admit random self-reductions. Hirahara and Santhanam introduced a weakening of random self-reductions that they called *pseudorandom* self-reductions, in which a language L is reduced to a distribution that is computationally indistinguishable from the uniform distribution. They then showed that the Minimum Circuit Size Problem (MCSP) admits a non-adaptive pseudorandom self-reduction, and suggested that this gave further evidence that MCSP is "distinguished" from standard NP-Complete problems.

We show that, in fact, the NP-Complete Clique problem admits a non-adaptive pseudorandom self-reduction, assuming the planted clique conjecture. More generally we show the following. Call a property of graphs π *hereditary* if $G \in \pi$ implies $H \in \pi$ for every induced subgraph of G. We show that for *any* infinite hereditary property π, the problem of finding a maximum induced subgraph $H \in \pi$ of a given graph G admits a non-adaptive pseudorandom self-reduction.

1 Introduction

A language L is *randomly-self-reducible* if L admits a "worst-case" to "average-case" reduction on the uniform distribution — that is, if we can reduce solving the problem on any worst-case instance to solving the problem on uniformly-random instances in polynomial time. For example, it was famously shown by Lipton [Lip89] that the problem of computing the permanent of a matrix admits a random-self-reduction. Many other central examples of random-self-reducibility come from cryptography — such as the discrete logarithm and the quadratic non-residuosity problems [AFK89] — where it is typically exploited to strengthen several cryptographic assumptions from average-case hardness to worst-case hardness without loss of generality.

A central open question in complexity theory is whether or not any NP-Complete set admits a random self-reduction [BL99, FFLS92, HNOS96]. This is closely related to the problem of whether or not the hardness of *distributional* languages in NP can be based on typical NP-Completeness assumptions (and, in particular, if "natural" NP-Complete problems are still hard over natural input distributions) [AGGM06]. Feigenbaum and Fortnow [FF90] famously showed that if an NP-Complete language is *non-adaptively* random-self-reducible (meaning that the queries to the random distribution must not be allowed to adaptively depend on earlier queries), then the polynomial hierarchy collapses to the third level. Feigenbaum and Fortnow's result was improved by Bogdanov and Trevisan [BT06] to show that if an NP-Complete set is non-adaptively self reducible to *any* polynomial-time sampleable distribution, then the polynomial hierarchy similarly collapses to the third level. Bogdanov and Trevisan's result shows that if we can base the distributional hardness of an NP-language on standard worst-case NP-Completeness, then the reduction witnessing this theorem must be adaptive. Along these lines, it is important to mention recent work by Hirahara, which showed that showing that average-case hardness of NP *can* at least be based on *exponential* hardness of NP [Hir21].

[1]Technion, reyad.2002r@gmail.com
[2]McGill University, robere@cs.mcgill.ca. Supported by NSERC.
[3]Technion, assaf@technion.ac.il
[4]Technion, ygal@cs.technion.ac.il

ISSN 1433-8092

In a recent paper, Hirahara and Santhanam [HS17] introduced a generalization of random self-reducibility that they called *pseudorandom self-reducibility*; now, the algorithm that performs the reduction is allowed to reduce to a distribution that is *computationally indistinguishable* from the uniform distribution. Under standard cryptographic assumptions they showed that the *Minimum Circuit Size Problem* (MCSP) admits a pseudorandom self-reduction (and, furthermore, their reduction is easily seen to be non-adaptive). There is much other evidence indicating that any reduction that would prove MCSP is NP-Complete must be surprisingly different from "standard" reductions [AH19, MW17, HW16, AD17], and so, comparing this with the prior results about random self-reducibility for NP, Hirahara and Santhanam suggested that their pseudorandom self-reduction for MCSP further distinguished it from other NP-Complete problems.

1.1 Our Results

In this work we show that, somewhat surprisingly, the classic NP-Complete Clique problem *does* admit a non-adaptive pseudorandom self-reduction under a non-uniform variant of *planted clique conjecture*. Let $G(n,p)$ denote the usual Erdős-Rényi random graph, and let $G(n,p,k)$ denote the distribution obtained by first sampling $G \sim G(n,p)$ and then choosing a random set of k vertices and planting a clique on those vertices.

Conjecture 1.1 (Non-Uniform Planted Clique Conjecture)**.** There is some $0 < \varepsilon_0 < 1/2$ such that, letting $k = n^{\varepsilon_0}$, for every sequence of polynomial-size circuits $\{C_n\}$

$$\left| \Pr_{G \sim G(n,1/2)}[C_n(G) = 1] - \Pr_{H \sim G(n,1/2,k)}[C_n(H) = 1] \right| \leq \frac{1}{n}.$$

Our main result is the following (as the formal definition of a pseudorandom self-reduction is somewhat technical we refer the reader to Section 2):

Theorem 1.1. The Clique problem admits a non-adaptive pseudorandom self-reduction, assuming the Non-Uniform Planted Clique Conjecture.

In fact, using our techniques, we can prove something a bit stronger. A *graph property* π is a set of graphs closed under isomorphism. A graph property π is *hereditary* if $G \in \pi$ implies $H \in \pi$ for each induced subgraph H of G, and it is *non-trivial* if both π and its complement are infinite. Consider the following decision problem:

Definition 1.1. Let π be an infinite, hereditary graph property. The π-induced subgraph problem, denoted π-SUB, is defined as follows. As input, we receive an undirected graph G, as well as a positive integer k. The goal is to decide if G contains an induced subgraph H such that $H \in \pi$ and H has at least k vertices.

First, it is easy to see that the π-SUB is more general than the Clique problem, since the Clique problem is simply the case where π is the set of all complete graphs. The π-induced subgraph problem has been considered in several previous works [LY80, FK05, BT95, LY93], where it was shown that it is NP-Complete and hard to approximate within a factor $n^{1-\varepsilon}$ for *every* infinite hereditary π. We show the following:

Theorem 1.2. For every non-trivial, hereditary π, the π-SUB problem admits a pseudorandom self-reduction, assuming the Non-uniform Planted Clique Conjecture.

To the best of our knowledge, this is the first worst-case to average-case reduction for any NP-Complete problem to a distribution that is "near" uniform, in any reasonable sense. However, as we will see next in our technical overview, our reduction relies crucially on some very special properties of the Clique problem (properties that are shared by the π-SUB problem), and because of this it appears to be difficult to extend it to other NP-Complete problems.

1.2 Technical Overview

We now sketch our reduction, specialized to the Clique problem. Our reduction relies crucially on the following special properties of the Clique problem that seem to distinguish it among NP-Complete problems:

- *Very Hard to Approximate.* Approximating the size of the largest clique in a graph is NP-Hard even within a multiplicative $n^{1-\varepsilon}$ factor for all $\varepsilon > 0$.
- *Small Value on Random Instances.* When $G \sim G(n, 1/2)$, then the size of the largest clique in G is $2\log n$ (up to lower-order terms) with high probability (see e.g. [Mat76]).

It seems that nearly all standard NP-Complete problems break one of these two requirements. For instance, the *Colouring* problem is hard to approximate, but, random graphs require a large number of colours to properly colour. On the other hand, random instances of the MAX-k-SAT problem have been very well-studied and it is easy to find random instances (below the "SAT threshold") which are easy to satisfy; but, it is well known that the MAX-k-SAT problem is easy to approximate by simply choosing a random assignment.

On to discussing our reduction for Clique. By standard hardness-of-approximation results, we can assume that the Gap-Clique promise problem — where we must distinguish between graphs containing cliques of size $n^{1-\varepsilon}k$ or graphs in which every clique has size at most k — is hard. Our reduction then proceeds as follows: we choose a random subset $U \subseteq V$ of vertices of G (say, of size n^{ϵ_0} for some appropriate constant ϵ_0) and randomize *all* edges with at most one endpoint inside of U. If G originally contained a large clique (of size $\gg n^{1-\varepsilon}k$), then a large portion of that clique will intersect U with high probability, and by using the fact that random graphs have very small cliques, it follows that solving the clique problem on the resulting graph will yield a good approximation to the size of the clique on the original graph G. Note here we have crucially used both properties (1) and (2) listed above.

The novel part of the reduction is proving that it is pseudorandom modulo the Planted Clique Conjecture. To do this we use the following "Xor-trick" (a form of this trick also played a role in the pseudorandom self-reduction for MCSP by Hirahara and Santhanam [HS17]). Suppose that the reduction was not pseudorandom, so that we obtain a sequence of graphs $\{G_n\}$ and a family of boolean circuits C_n such that C_n can distinguish between the above "planted" distribution (obtained by taking G_n and randomizing all edges outside a random subset of vertices) and uniformly random graphs $G \sim G(n, 1/2)$. Using this family of circuits C_n we will construct a new family of circuits C'_n that can detect the existence of planted cliques in random graphs, violating the Planted Clique Conjecture. The new family C'_n is defined as follows: given a graph H as input, C'_n takes the Xor of the edge-set of the complement graph $\overline{H}$ with the edge-set of G_n. If H was a uniformly random graph, then the result will be a uniformly random graph. If, however, H had a planted clique, then $\overline{H}$ will have a planted independent set. Thus, taking the Xor of $\overline{H}$ with G_n will result in a random graph with uniformly random edges except for a random subset of G_n. We can therefore apply the family of circuits C_n that differentiates between the "planted" distribution and uniformly random graphs and differentiate between planted cliques and random graphs.

Now that we have discussed our reduction, note that a worst-case to average-case reduction for any problem implies that an efficient algorithm solving the average-case problem also implies an efficient algorithm that solves the worst-case problem. Of course, approximating the value of the largest clique on a $G(n, 1/2)$ graph is actually *easy*: as we have discussed above, the size of the largest clique is $(2 - o(1))\log n$ with high probability, and a simple greedy algorithm will find a clique of size roughly $\log n$ with high probability [Kar76]. However, in our case, if the Planted Clique conjecture is true then this good approximation algorithm does not imply a good approximation algorithm for the Max-Clique problem as, intuitively, the pseudorandomness "fools" the polynomial-time algorithm into thinking that there is a clique of size $\approx 2\log n$, when in fact the graph actually contains a much larger clique. On the other hand, if the Planted Clique conjecture is false, then a polynomial-time algorithm could perhaps distinguish the output graphs of the reduction from $G(n, 1/2)$, but, in order to solve Clique it must find a very large clique inside the randomly planted portion, which still could be a very hard problem.

1.3 Related Work

The *planted clique problem* is a well-studied problem in both complexity theory and algorithms that was introduced independently by Jerrum [Jer92] and Kučera [Kuc95]; although, the hardness of finding cliques in random graphs was initially observed by Karp, who observed that there is no known polynomial-time algorithm finding cliques of size $\approx 2\log n$ in random graphs [Kar76], even though they exist with probability $1 - o(1)$. It is well-known that the planted clique problem can be solved by a quasipolynomial-time algorithm

that simply enumerates all potential cliques of size $O(\log n)$. As for polynomial-time algorithms, a classic result due to Alon, Krivelevich and Sudakov [AKS98] finds planted cliques of size $\Omega(n^{1/2})$ using semidefinite programming. The planted clique problem has also been used in prior works as a hardness assumption in complexity theory and cryptography (see, e.g. [JP00, HK11]), and it is known to be hard to solve in both the Lovász-Shrijver and Sum-of-Squares convex programming hierarchies [FK03, BHK$^+$19].

There has been much work regarding the study of average-case self-reducibility of NP problems. Thanks to the *negative* results by Feigenbaum and Fortnow [FF90] and Bogdanov and Trevisan [BT06], the power of non-adaptive random reductions inside of NP is now fairly well understood: it is known that any such problem must lie in $\mathsf{NP/poly} \cap \mathsf{coNP/poly}$ [BT06]. There have also been some *positive* results. Feigenbaum, Fortnow, Lund and Spielman showed that under plausible assumptions, there is a function in $\mathsf{NP} \setminus \mathsf{P}$ which is adaptively random-self-reducible but not nonadaptively random-self-reducible [FFLS92]. Hemaspaandra, Naik, Ogihara and Selman showed that if $\mathsf{NP} \not\subseteq \mathsf{BPE}$ then there is a set in $\mathsf{NP} \setminus \mathsf{BPP}$ which is adaptively random-self-reducible, but neither nonadaptively random-self reducible nor self-reducible [HNOS96]. Hirahara recently gave a worst-case to average-case reduction from the Minimum Time-Bounded Kolmogorov Complexity problem (MinKT) (which is widely believed to lie *outside* of NP) to a distributional problem inside of NP [Hir18]. Another recent result of Hirahara shows that average-case hardness of problems in NP *can* be based on sufficiently strong exponential hardness of the closely related class UP [Hir21]. We refer to Bogdanov and Trevisan [BT06] for an excellent (if somewhat dated) survey of the average-case complexity of NP, and to Hirahara [Hir21] for a modern discussion of frontier open questions.

Worst case to average case reductions for problems in P were studied in [GR20]. They showed a subclass of problems in P which admit a random self reductions, such as counting the number of fixed-size cliques in a graph.

2 Preliminaries

If D is a probability distribution, we denote by $x \sim D$ an element sampled according to D. A promise problem is a pair $\Pi = (\Pi_{\text{YES}}, \Pi_{\text{NO}})$, where $\Pi_{\text{YES}}, \Pi_{\text{NO}} \subseteq \{0,1\}^*$ and $\Pi_{\text{YES}} \cap \Pi_{\text{NO}} = \emptyset$. A language $L \subseteq \{0,1\}^*$ is consistent with the promise problem $\Pi = (\Pi_{\text{YES}}, \Pi_{\text{NO}})$ if $\Pi_{\text{YES}} \subseteq L$ and $\Pi_{\text{NO}} \subseteq \bar{L}$.

In this paper, graphs are simple and undirected. Denote by $\mathcal{G}(n)$ the set of all graphs over n vertices. We assume that a graph G with n vertices is encoded using a binary string of length $\binom{n}{2}$. We denote by $\omega(G)$ the largest clique in G.

We borrow some definitions from [HS17].

Definition 2.1. (Indistinguishability). Let $\mathcal{C}$ be a (uniform or non-uniform) complexity class, and $\{D_n\}_{n\in\mathbb{N}}$, $\{D'_n\}_{n\in\mathbb{N}}$ two sequences of distributions such that for all n, D_n and D'_n are supported on $\{0,1\}^n$. We say that $\{D_n\}$ and $\{D'_n\}$ are indistinguishable by $\mathcal{C}$, if for all $A \in \mathcal{C}$ and for all sufficiently large n,

$$|\Pr_{x\sim D_n}[A(x)=1] - \Pr_{x\sim D'_n}[A(x)=1]| \leq \frac{1}{n}\ .$$

Definition 2.2. (Pseudorandom Self-Reducibility, [HS17]). Let $\mathcal{C}$ be a complexity class. Let $Q = (\Pi_{\text{YES}}, \Pi_{\text{NO}})$ be a promise problem, where $\Pi_{\text{YES}}, \Pi_{\text{NO}} \subseteq \{0,1\}^*$, and let $L \subseteq \{0,1\}^*$ be a language. Q is said to be pseudorandomly reducible to L with respect to $\mathcal{C}$ if there are constants q, ℓ and polynomial time computable functions $g : \{0,1\}^* \to \{0,1\}^*$ and $h : \{0,1\}^* \to \{0,1\}^*$ satisfying the following conditions:

1. For every sequence $\{(x_n, i_n)\}_{n\in\mathbb{N}}$ where $x_n \in \{0,1\}^n$ and $1 \leq i_n \leq n^q$ for all $n \in \mathbb{N}$, the distributions $\{g(i_n, x_n, U_{n^\ell})\}_{n\in\mathbb{N}}$ and $\{U_n\}_{n\in\mathbb{N}}$ are indistinguishable by $\mathcal{C}$.
2. For large enough n and for every $x \in (\Pi_{\text{YES}} \cup \Pi_{\text{NO}}) \cap \{0,1\}^n$:

$$Q(x) = h(x, r, L(g(1,x,r)), L(g(2,x,r)), \ldots, L(g(n^q, x, r))),$$

with probability at least $1 - 2^{-n}$ when $r \sim U_{n^\ell}$.

The reduction is *non-adaptive* if the later queries to random instances cannot depend on earlier queries to random instances.

Probabilistic bounds. The Chernoff-type bound we use in this paper is stated below.

Theorem 2.1. (Chernoff's inequality, [Che52]). Let $X = X_1 + \ldots + X_n$ where X_i are independent random variables taking values in $\{0, 1\}$. Then

$$\Pr[|X - \mathbb{E}[X]| \geq \frac{1}{2}\mathbb{E}[X]] \leq 2e^{-\mathbb{E}[X]/16}.$$

In addition, we will need the following result by Hoeffding (Theorem 4 in [Hoe94]).

Lemma 2.1. Let $S = (s_1, \ldots, s_N)$ be a finite population of N real points, $X_1, \ldots, X_n$ denote a uniformly random sample without replacement from S and $Y_1, \ldots Y_n$ denote a uniformly random sample with replacement from S. If $f : \mathbb{R} \to \mathbb{R}$ is continuous and convex, then

$$\mathbb{E}[f\Big(\sum_{i=1}^{n} X_i\Big)] \leq \mathbb{E}[f\Big(\sum_{i=1}^{n} Y_i\Big)].$$

Lemma 2.1 implies that we can use Chernoff's inequality for the random variables $\{X_i\}$, even though they are dependent[1].

3 Non-uniform Planted Clique

In this section we state our hardness assumption, which essentially says that polynomial size circuits cannot distinguish between a random graph, and a random graph with a planted clique of size n^{ϵ_0}, for some $\epsilon_0 \in (0, \frac{1}{2})$.

For a graph G and a parameter $0 < \epsilon < 1$, we define the distribution $P(G, \epsilon)$ by picking a random subset of vertices of G of size n^ϵ, keep the induced subgraph generated by this set, and randomize all edges not contained inside of G. Formally,

Definition 3.1. (Planted Subgraph Distribution). Let $G = (V, E)$ be a graph with n vertices, and let $\epsilon \in (0, 1)$. The distribution $P(G, \epsilon)$ is defined to be the output distribution of the following algorithm. Start with the graph $G = (V, E)$. Then, pick uniformly at random a subset $S \subset V$ of vertices of size $\lceil n^\epsilon \rceil$. Output a graph $G' = (V', E')$ where $V' = V$ and

$$\Pr[\{u, v\} \in E'] = \begin{cases} \frac{1}{2} & \text{if } u \notin S \text{ or } v \notin S, \\ 1 & \text{if } \{u, v\} \in E \text{ and } u, v \in S, \\ 0 & \text{if } \{u, v\} \notin E \text{ and } u, v \in S. \end{cases}$$

Let $\{K_n\}_{n\in\mathbb{N}}$ be the sequence of complete graphs over n vertices, and let $G(n, \frac{1}{2})$ be the uniform distribution over graphs with n vertices. Note that the distribution $P(K_n, \varepsilon)$ is exactly the same distribution as $G(n, 1/2, n^\varepsilon)$ (that is, choosing a random graph and planting a random clique the same as starting with a complete graph and randomizing all edges outside of random small set). The Planted Clique Conjecture states that there is a constant $\epsilon_0 \in (0, \frac{1}{2})$ such that there is no polynomial time algorithm that can distinguish between $G(n, \frac{1}{2})$ and $P(K_n, \epsilon_0)$ with high probability[2]. In this paper we use a slightly stronger version of the Planted Clique Conjecture that requires hardness for polynomial-size circuits.

Conjecture 3.1. There exists some $\epsilon_0 \in (0, \frac{1}{2})$ such that there is no sequence of polynomial size circuits $\{C_n\}_{n\in\mathbb{N}}$ satisfying

$$|\Pr_{G\sim G(n,\frac{1}{2})}[C_n(G) = 1] - \Pr_{G'\sim P(K_n,\epsilon_0)}[C_n(G') = 1]| \leq \frac{1}{n}\ .$$

[1] Taking the function $f(x) = e^{tx}$, we get that $\mathbb{E}[\Pi e^{tX_i}] \leq \mathbb{E}[\Pi e^{tY_i}]$, where the random variables $\{Y_i\}$ are independent. Thus the Chernoff bound can be derived for the random variables $\{X_i\}$ as well.

[2] Some papers states this conjecture for all $\epsilon \in (0, \frac{1}{2})$, but for our purposes it is enough to assume the weaker version of the conjecture.

We observe that the non-uniform planted clique conjecture is equivalent to the following conjecture where we replace the sequence of graphs $\{K_n\}$ with *any* fixed sequence of graphs $\{G_n\}$.

Conjecture 3.2. There exists some $\epsilon_0 \in (0, \frac{1}{2})$ such that for any sequence of graphs over n vertices $\{G_n\}_{n\in\mathbb{N}}$, there is no sequence of polynomial size circuits $\{C_n\}_{n\in\mathbb{N}}$ satisfying

$$|\Pr_{G\sim G(n,\frac{1}{2})}[C_n(G)=1] - \Pr_{G'\sim P(G_n,\epsilon_0)}[C_n(G')=1]| \leq \frac{1}{n}\ .$$

Claim 3.1. Conjectures 3.1 and 3.2 are equivalent.

Proof. Clearly, Conjecture 3.2 implies Conjecture 3.1. We show the converse direction.

Assume by way of contradiction that Conjecture 3.2 is false, and we show that Conjecture 3.1 is false. In particular, assume that there is a sequence $\{G_n\}_{n\in\mathbb{N}}$ of graphs and a sequence $\{C_n\}_{n\in\mathbb{N}}$ of polynomial size circuits, such that

$$|\Pr_{G\sim G(n,\frac{1}{2})}[C_n(G)=1] - \Pr_{G'\sim P(G_n,\epsilon_0)}[C_n(G')=1]| > \frac{1}{n}.$$

Define the boolean circuit $C'_n(G) = C_n(G \oplus G_n \oplus K_n)$, where $\oplus$ is the symmetric difference of edge sets of graphs. If $G \sim P(K_n, \varepsilon_0)$ then $G \oplus K_n \oplus G_n$ is distributed according to $P(G_n, \varepsilon_0)$, and if $G \sim G(n, 1/2)$ then $G \oplus K_n \oplus G_n$ is also distributed according to $G(n, 1/2)$. This implies that the sequence of circuits $\{C'_n\}_{n\in\mathbb{N}}$ can distinguish between a random graph and a random graph with a planted clique of size n^{ϵ_0}, contradicting Conjecture 3.1. □

4 Self-Reductions for Clique

Before proving the general theorem, we demonstrate our method on the language CLIQUE: we show that CLIQUE is pseudorandomly self-reducible. Thanks to the hardness of approximation results for CLIQUE [Zuc06, Has96], it is enough to consider the promise problem GAP-CLIQUE$_\beta$, defined below.

Definition 4.1. For $\beta \in (0, 1)$, define the promise problem GAP-CLIQUE$_\beta = (\Pi_{\text{YES}}, \Pi_{\text{NO}})$ by

$$\Pi_{\text{YES}} = \{G : G \text{ is a graph with } n \text{ vertices and } \omega(G) \geq n^{1-\beta}\},$$

and

$$\Pi_{\text{NO}} = \{G : G \text{ is a graph with } n \text{ vertices and } \omega(G) < n^{\beta}\}.$$

Theorem 4.1 ([Zuc06])**.** For any $\beta > 0$ the GAP-CLIQUE$_\beta$ problem is NP-Hard under polynomial-time many-one reductions.

We proceed to stating the main theorem of this section, which shows that GAP-CLIQUE$_\beta$ is pseudorandomly self-reducible. By combining this with the above NP-Hardness result and the fact that GAP-CLIQUE$_\beta$ is itself a subproblem of Clique we immediately obtain pseudorandom self-reducibility of Clique.

Theorem 4.2. If the Planted Clique Conjecture holds, then for every $\beta \in (0, \epsilon_0)$, where ϵ_0 is the constant from Conjecture 3.2, GAP-CLIQUE$_\beta$ is pseudorandomly self-reducible with respect to SIZE(poly).

Before we prove the theorem, we will need the following lemma showing that if we take a graph with a large clique and choose a random subset of vertices U and randomize every edge with at most one endpoint in U, then the size of the largest clique will not be badly perturbed with high probability.

Lemma 4.1. Let $G = (V, E)$ be a graph over n vertices and $\beta \in (0, \epsilon_0)$. Set $\delta := \epsilon_0 - \beta$. Let $P(G, \epsilon_0)$ be the planted distribution for G defined in Definition 3.1. Then, for large enough n:

1. If $\omega(G) \geq n^{1-\beta}$, then

$$\Pr_{G'\sim P(G,\epsilon_0)}[\omega(G') \geq \frac{1}{2}n^{\delta}] \geq 1 - 2e^{-\frac{n^\delta}{16}} = 1 - o(1).$$

2. If $\omega(G) < n^\beta$, then

$$\Pr_{G' \sim P(G,\epsilon_0)}[\omega(G') < \frac{1}{2}n^\delta] \geq 1 - 2^{-\frac{1}{36}n^{2\delta}} = 1 - o(1).$$

Proof. We start by proving the first statement. Let G' be the graph obtained from G by $P(G, \epsilon_0)$, let S, $|S| = n^{\epsilon_0}$ be the set of vertices in G' preserved by $P(G, \epsilon_0)$, and let T, $|T| \geq n^{1-\beta}$ be the set of the maximal clique vertices in G. We have,

$$\Pr[\omega(G') \geq \frac{1}{2}n^\delta] \geq \Pr[|T \cap S| \geq \frac{1}{2}n^\delta].$$

For a vertex $v \in T$, define an indicator random variable X_v, such that $X_v = 1$ if and only if $v \in S$. Note that $\Pr[X_v = 1] = n^{\epsilon_0 - 1}$. We have,

$$\mathbb{E}[|T \cap S|] = \sum_{v \in T} \mathbb{E}X_v \geq \frac{n^{1-\beta}}{n^{1-\epsilon_0}} = n^\delta.$$

By Lemma 2.1 we can use the Chernoff bound for the random variables X_v, even though they are dependent. Thus, for n large enough,

$$\begin{aligned}\Pr[||T \cap S| < \frac{1}{2}n^\delta] &\leq \Pr[||T \cap S| - \mathbb{E}|T \cap S|| \geq \frac{1}{2}\mathbb{E}|T \cap S|] \\ &\leq 2e^{-\mathbb{E}|T \cap S|/16} \leq 2e^{-\frac{n^\delta}{16}}.\end{aligned}$$

We move to the second part of the Lemma. Intuitively, with high probability, the largest clique in G' is of size at most $\omega(G) + 2\log n$: on the set S of preserved vertices the largest clique is of size at most $\omega(G)$, and with high probability the largest clique outside S is roughly of size $2\log n$. Thus, the probability that $\omega(G') \geq \frac{1}{2}n^\delta$ is tiny. We formalize this intuition. Denote by $G' \setminus S$ the induced subgraph obtained by removing the vertices in S from G'. For large enough n,

$$\begin{aligned}\Pr[\omega(G') \geq \frac{1}{2}n^\delta] \leq \Pr[\omega(G' \setminus S) \geq \frac{1}{2}n^\delta - n^\beta] &\leq \Pr[\omega(G' \setminus S) > \frac{1}{3}n^\delta] \\ &\leq \binom{n}{\frac{1}{3}n^\delta} \frac{1}{2^{\binom{\frac{1}{3}n^\delta}{2}}}.\end{aligned}$$

Using $\binom{n}{m} \leq n^m$ we get:

$$\binom{n}{\frac{1}{3}n^\delta} \frac{1}{2^{\binom{\frac{1}{3}n^\delta}{2}}} \leq n^{\frac{1}{3}n^\delta} 2^{-\binom{\frac{1}{3}n^\delta}{2}} \leq 2^{-\frac{1}{36}n^{2\delta}}.$$ □

Proof of Theorem 4.2. We show that there is a pseudorandom reduction from GAP-CLIQUE$_\beta$ to CLIQUE. We need to define the functions h and g, as required by Definition 2.2.

The function g. On input (i, G, r), where $1 \leq i \leq n^4$, G is an encoding of a graph with n vertices, and r is a random binary string composed of n^4 blocks of size $\binom{n}{2}$ each, g uses the i'th block of the random bits in r in order to sample a graph $G' \sim P(G, \epsilon_0)$, for ϵ_0 as in Lemma 4.1. Then, g outputs G'.

The function h. The function h simply computes the majority of the queries g_i and answers accordingly.

We show that the functions h and g satisfy the requirements of Definition 2.2. We need to prove that the queries are pseudorandom, and that the reduction works with high probability. The fact that the queries are pseudorandom follows immediately from Conjecture 3.2. Considering any G output by the function g the graph G' is distributed according to $P(G, \epsilon_0)$. By Conjecture 3.2 (which is equivalent to the Planted Clique Conjecture) the graph G' is indistinguishable from a uniformly random graph, and thus the queries are pseudorandom.

The reduction succeeds with high probability by Lemma 4.1 and a standard Chernoff bound argument.

5 Self-Reductions for Hereditary Properties

Instead of searching for the largest clique in a graph, we can search for the largest induced subgraph satisfying some property (e.g. largest planner subgraph, largest connected subgraph, etc.). Formally, a *graph property* π is a set of graphs, closed under isomorphism. A property π is *non-trivial* if both π and its complement are infinite. For a property π and a graph G, denote by $\alpha_\pi(G)$ the size of the largest set of nodes inducing a graph in π. The promise problem for π is defined in the natural way,

Definition 5.1. For $\beta \in (0, \epsilon_0)$, where ϵ_0 is the constant from Conjecture 3.2[3], define the promise problem GAP-$\pi_\beta = (\Pi_{\text{YES}}, \Pi_{\text{NO}})$ by

$$\Pi_{\text{YES}} = \{G : G \text{ is a graph with } n \text{ vertices and } \alpha_\pi(G) \geq n^{1-\beta}\},$$

and

$$\Pi_{\text{NO}} = \{G : G \text{ is a graph with } n \text{ vertices and } \alpha_\pi(G) < n^{\beta}\}.$$

For which graph properties can the pseudorandom self reduction from the previous section work? A more careful look at the previous result shows that in order for the reduction to work, it is sufficient for the property π to satisfy:

1. *Stability.* A graph property π is stable if whenever a graph G has a "large" subgraph in the property π, then $P(G, \epsilon_0)$ also has a "large" subgraph in the property, where $P(G, \epsilon_0)$ is the distribution defined in Definition 3.1. Formally,

$$\alpha_\pi(G) \geq n^{1-\beta} \implies \Pr_{G' \sim P(G,\epsilon_0)}[\alpha_\pi(G') \geq \frac{1}{2}n^\delta] \geq \frac{2}{3},$$

   where ϵ_0 is the constant from Conjecture 3.2, $\beta \in (0, \epsilon_0)$ and $\delta = \epsilon_0 - \beta$. Intuitively, it means that a "YES" instance is mapped to a "YES" instance.

2. *Non-density*[4]. For a graph property π, denote by $\text{gr}_\pi(n) := |\pi \cap \mathcal{G}(n)|$ the number of graphs over n vertices in the property. A graph property π is non-dense if there exists a constant $\epsilon > 0$ such that for large enough n

$$\text{gr}_\pi(n) \leq 2^{(1-\epsilon)\binom{n}{2}}.$$

   Intuitively, we need this requirement in order to make sure that in the process of randomizing "most" of the input graph, with high probability we did not create a large subgraph in the property. It means that a "NO" instance is mapped to a "NO" instance.

3. *Hard to Approximate.* The conditions above give a pseudorandom self reduction for the promise problem GAP-π. In case we want to obtain a pseudorandom self reduction for the language π-SUB mentioned in the introduction, we need to use hardness of approximation results.

We now characterize a family of graph properties satisfying the three above requirements.

Definition 5.2. A graph property π is hereditary if whenever a graph G is in π, then every induced subgraph of G is also in π.

Theorem 5.1. Let π be a non-trivial hereditary graph property. Then π is stable, non-dense and hard to approximate.

First, we sketch the proof that π is stable, as it is essentially the same as in the case of Clique.

Proof. Let π be a non-trivial hereditary graph property. Let G be a graph so that $\alpha_\pi(G) \geq n^{1-\beta}$, and let H be the largest subgraph of G in the property π. As shown in the proof of Lemma 4.1, with probability at least $1 - 2e^{-\frac{n^\delta}{16}}$, $G' \sim P(G, \epsilon_0)$ contains a subgraph of H with $\frac{1}{2}n^\delta$ vertices. Since π is a hereditary property, this subgraph is also in π, and so $\alpha_\pi(G') \geq \frac{1}{2}n^\delta$ with probability at least $1 - o(1)$. □

[3] We choose to address only the case where $\beta \in (0, \epsilon_0)$, since our reduction only works in this range.

[4] We call this "non-density" rather than "sparsity" since the level of density we can afford is really quite high!

The hardness of approximation for non-trivial hereditary properties was proven by Lurid and Yannakakis [LY93], and was later improved by Feige and Kogan [FK05]. The result in [FK05] can be derandomized in the same manner as in [Zuc06] in order to obtain the following theorem.

Theorem 5.2. For every nontrivial hereditary property π and for every $\beta > 0$, the π-SUB problem cannot be approximated within a factor of $n^{1-\beta}$, unless P = NP.

Additionally, Bollobás and Thomason showed in [BT95, Theorem 8] that for a hereditary property π, if $\mathrm{gr}_\pi(n) = 2^{c_n\binom{n}{2}}$ then c_n is monotonically decreasing, therefore the following result follows:

Theorem 5.3. Let π be a non-trivial hereditary property, then there exists some $\epsilon_1 > 0$ such that for n large enough

$$\mathrm{gr}_\pi(n) < 2^{(1-\epsilon_1)\binom{n}{2}}$$

Thus, every non-trivial hereditary property is non-dense.

To see why this requirement guarantees that a "NO" instance is mapped to a "NO" instance with high probability, observe the following claim.

Claim 5.1. Let π be a non-trivial graph property. Then for every $\delta > 0$,

$$\Pr_{G\sim G(n,\frac{1}{2})}[\alpha_\pi(G) \geq n^\delta] = o(1).$$

Proof. Let $\epsilon_1 > 0$ satisfying $\mathrm{gr}_\pi(n) \leq 2^{(1-\epsilon_1)\binom{n}{2}}$, then:

$$\Pr_{G\sim G(n,\frac{1}{2})}[\alpha_\pi(G) \geq n^\delta] \leq \binom{n}{n^\delta}\frac{\mathrm{gr}_\pi(n^\delta)}{2^{\binom{n^\delta}{2}}} \leq n^{n^\delta}2^{-\epsilon_1\binom{n^\delta}{2}} = o(1). \qquad \square$$

Therefore Lemma 4.1 still holds for *any* non-trivial hereditary property, and so for every non-trivial hereditary property the language π-SUB admits a pseudorandom self-reduction.

References

[AD17] Eric Allender and Bireswar Das. Zero knowledge and circuit minimization. *Information and Computation*, 256:2–8, 2017.

[AFK89] Martín Abadi, Joan Feigenbaum, and Joe Kilian. On hiding information from an oracle. *Journal of Computer and System Sciences*, 39(1):21–50, 1989.

[AGGM06] Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on np-hardness. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 701–710, 2006.

[AH19] Eric Allender and Shuichi Hirahara. New insights on the (non-) hardness of circuit minimization and related problems. *ACM Transactions on Computation Theory (ToCT)*, 11(4):1–27, 2019.

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

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

[BL99] László Babai and Sophie Laplante. Stronger separations for random-self-reducibility, rounds, and advice. In *Proceedings. Fourteenth Annual IEEE Conference on Computational Complexity (Formerly: Structure in Complexity Theory Conference)(Cat. No. 99CB36317)*, pages 98–104. IEEE, 1999.

[BT95] Bela Bollobas and Andrew Thomason. Projections of bodies and hereditary properties of hypergraphs. *Bull. London Math. Soc*, 27:417–424, 1995.

[BT06] Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for np problems. *SIAM Journal on Computing*, 36(4):1119–1159, 2006.

[Che52] Herman Chernoff. A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the sum of Observations. *The Annals of Mathematical Statistics*, 23(4):493 – 507, 1952.

[FF90] J Feigenbaum and L Fortnow. On the random-self-reducibility of complete sets, university of chicago technical report 90-22. *Computer Science Department*, 1990.

[FFLS92] Joan Feigenbaum, Lance Fortnow, Carsten Lund, and Daniel A Spielman. The power of adaptiveness and additional queries in random-self-reductions. In *Computational Complexity Conference*, pages 338–346, 1992.

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

[FK05] Uriel Feige and Shimon Kogan. The hardness of approximating hereditary properties. *Available on: http://research. microsoft. com/research/theory/feige/homepagefiles/hereditary. pdf*, pages 1–12, 2005.

[GR20] Oded Goldreich and Guy N Rothblum. Worst-case to average-case reductions for subclasses of p., 2020.

[Has96] Johan Hastad. Clique is hard to approximate within $n^{1-\varepsilon}$. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 627–636. IEEE, 1996.

[Hir18] Shuichi Hirahara. Non-black-box worst-case to average-case reductions within NP. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 247–258. IEEE Computer Society, 2018.

[Hir21] Shuichi Hirahara. Average-case hardness of np from exponential worst-case hardness assumptions. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 292–302, 2021.

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

[HNOS96] Edith Hemaspaandra, Ashish V Naik, Mitsunori Ogihara, and Alan L Selman. P-selective sets and reducing search to decision vs self-reducibility. *Journal of Computer and System Sciences*, 53(2):194–209, 1996.

[Hoe94] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In *The collected works of Wassily Hoeffding*, pages 409–426. Springer, 1994.

[HS17] Shuichi Hirahara and Rahul Santhanam. On the average-case complexity of mcsp and its variants. In *32nd Computational Complexity Conference (CCC 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[HW16] Shuichi Hirahara and Osamu Watanabe. Limits of minimum circuit size problem as oracle. In *31st Conference on Computational Complexity (CCC 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[Jer92] Mark Jerrum. Large cliques elude the metropolis process. *Random Struct. Algorithms*, 3(4):347–360, 1992.

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

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

[Kuc95] Ludek Kucera. Expected complexity of graph partitioning problems. *Discret. Appl. Math.*, 57(2-3):193–212, 1995.

[Lip89] R. Lipton. New directions in testing. In *Distributed Computing And Cryptography*, 1989.

[LY80] John M Lewis and Mihalis Yannakakis. The node-deletion problem for hereditary properties is np-complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980.

[LY93] Carsten Lund and Mihalis Yannakakis. The approximation of maximum subgraph problems. In *International Colloquium on Automata, Languages, and Programming*, pages 40–51. Springer, 1993.

[Mat76] David W Matula. *The largest clique size in a random graph.* Department of Computer Science, Southern Methodist University Dallas, Texas ..., 1976.

[MW17] Cody D Murray and R Ryan Williams. On the (non) np-hardness of computing circuit complexity. *Theory of Computing*, 13(1):1–22, 2017.

[Zuc06] David Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 681–690, 2006.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il