Learning-Augmented Maximum Independent Set

Vladimir Braverman ✉
Rice University, Houston, TX, USA
Google Research

Prathamesh Dharangutte ✉
Rutgers University, NJ, USA

Vihan Shah ✉
University of Waterloo, ON, Canada

Chen Wang ✉
Rice University, Houston, TX, USA
Texas A&M University, College Station, TX, USA

Abstract

We study the Maximum Independent Set (MIS) problem on general graphs within the framework of learning-augmented algorithms. The MIS problem is known to be NP-hard and is also NP-hard to approximate to within a factor of $n^{1-\delta}$ for any $\delta > 0$. We show that we can break this barrier in the presence of an oracle obtained through predictions from a machine learning model that answers vertex membership queries for a fixed MIS with probability $1/2 + \varepsilon$. In the first setting we consider, the oracle can be queried once per vertex to know if a vertex belongs to a fixed MIS, and the oracle returns the correct answer with probability $1/2 + \varepsilon$. Under this setting, we show an algorithm that obtains an $\widetilde{O}(\sqrt{\Delta}/\varepsilon)^1$-approximation in $O(m)$ time where Δ is the maximum degree of the graph. In the second setting, we allow multiple queries to the oracle for a vertex, each of which is correct with probability $1/2 + \varepsilon$. For this setting, we show an $O(1)$-approximation algorithm using $O(n/\varepsilon^2)$ *total* queries and $\widetilde{O}(m)$ runtime. [2]

2012 ACM Subject Classification Theory of computation → Graph algorithms analysis; Computing methodologies → Machine learning

Keywords and phrases Learning-augmented algorithms, maximum independent set, graph algorithms

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2024.24

Category APPROX

Related Version *Full Version*: https://arxiv.org/abs/2407.11364

Funding *Vladimir Braverman*: Supported partially by the Naval Research (ONR) grant N00014-23-1-2737, and NSF-CNS 2333887 award.
Prathamesh Dharangutte: Supported by NSF through IIS-2229876 and CCF-2118953.
Vihan Shah: Supported in part by Sepehr Assadi's Sloan Research Fellowship and NSERC Discovery Grant.

Acknowledgements The authors are grateful to Sepehr Assadi and Samson Zhou for the helpful conversations regarding the project. The authors also thank anonymous APPROX reviewers for helpful suggestions.

1 Introduction

We consider learning-augmented *maximum independent set* (MIS) in this paper. Given a (unweighted, undirected) graph $G = (V, E)$, an independent set is a set of vertices $I \subseteq V$, such that for any $u, v \in I$, $(u, v) \notin E$, i.e., there is *no* edge between u and v. The maximum independent set problem asks to find the independent set with the largest size in G.

[1] Throughout we use $\widetilde{O}(\cdot)$ to hide polylog (n) factors.
[2] A full version appears on arxiv under the same title.

Finding the maximum independent set is one of the classical NP-hard problems [42]. Furthermore, the seminal work of [36, 60] demonstrates the NP-hardness of approximating the size of the MIS to within a factor of $n^{1-\delta}$ for any $\delta > 0$. In contrast, outputting any single vertex gives an n-approximation trivially. [10] gave a non-trivial $O(n/\log^2 n)$-approximation to MIS and this was later improved by [29]. These results indicate that the problem is quite hard in its general form and thus, many research efforts have been devoted to approximation algorithms in special settings, e.g., planar graphs [3, 47], rectangle-intersection graphs [16, 22, 32], and exponential-time algorithms [51, 31, 59, 12].

On the other hand, heuristic algorithms, despite their bad worst-case guarantees, often exhibit commendable performance on real-world graphs [4, 24, 57]. For instance, the greedy algorithm only offers an approximation guarantee of $O(\Delta)$, where Δ is the *maximum degree* of G. However, it frequently yields satisfactory empirical results. The gap between the worst-case hardness and practical efficiency motivates us to study the MIS problem through the lens of beyond worst-case analysis [11, 52]. In particular, under the modern context, we ask the question of finding the maximum independent set with *learning-augmented oracles*.

Learning-augmented algorithms

Learning-augmented algorithms, also known as algorithms with predictions, have attracted considerable attention in recent years (see, e.g. [50, 38, 46, 21, 56, 7, 9, 1, 37, 13, 53], and references therein). This paradigm of beyond worst-case analysis has been successful in surmounting classical thresholds and bridging the gap between the worst-case hardness and practical efficiency (see, e.g., [48], for an excellent summary). Typically, in learning-augmented algorithms, we assume the access to an oracle that gives part of the "right answer" to the problem, and fails with some small but non-negligible probability. Conceptually, these algorithms aim to take advantage of modern machine learning models, which are fairly accurate on predictive tasks yet make random mistakes in an inconsistent fashion. Learning-augmented algorithms provide a great way to analyze algorithms beyond the worst case, and these algorithms usually have immediate implications in practice (see the empirical results in, e.g., [20, 38, 26, 56, 1]). Inspired by the recent work in utilizing machine learning-based techniques for the maximum independent set [2, 49, 14], we consider the MIS problem through the lens of learning-augmented algorithms.

The advantage of the learning-augmented algorithms has inspired a flurry of work that studies *graph problems* within this framework [8, 30, 17, 18, 39, 5, 45, 6, 23, 33]. In a very recent work, [23] considered the Max-cut problem, in which the oracle model is closely related to our setting for the MIS problem. Under the Unique Game Conjecture (UCG), it is known that getting anything better than $\alpha \approx 0.878$ approximation for max-cut is NP-hard ([43]). In contrast, [23] showed that with a learning-augmented oracle, we could achieve better approximation than the α threshold in polynomial time. In another closely related work, [33] studied the more general constraint satisfaction problems (CSPs) trough the lens of the learning-augmented algorithms. There, they obtain results for both the Max-cut and the Max 2-Lin problem. Although [23, 33] defines more general learning-augmented oracles, they, unfortunately, fall short of capturing the MIS-type of CSP problems, and their results do not have direct implications on the MIS problem.

From the above discussion, we can see that *a*) studying the maximum independent set problem in the framework of learning-augmented algorithms has great potential; and *b*) to this end, the existing models and algorithms are not yet sufficient. In light of this, we ask the following question:

Under the framework of learning-augmented algorithms, what efficient algorithms can we get for the maximum independent set problem?

1.1 Our models and contributions

In what follows, we will define the learning-augmented oracle model we consider and present our main results.

Our oracle model

We consider the following natural learning-augmented oracle: for a fixed maximum independent set I^*, the oracle answers whether a vertex $v \in I^*$ correctly with probability $1/2 + \varepsilon$, and incorrectly with probability $1/2 - \varepsilon$. In addition, the randomness is *independent* across the vertices. We denote by $\texttt{ORC}_{G,I^*}(v)$ the answer the oracle gives when queried for vertex v.

We study approximation algorithms for MIS with the learning-augmented oracle in two settings: the *persistent noise* setting and the *non-persistent noise* setting. We discuss the settings and the results, respectively, as follows.

- The **persistent noise** setting. In the persistent noise setting, the randomness of $\texttt{ORC}_{G,I^*}$ is drawn exactly *once*. Therefore, the answer for a vertex will remain the same no matter how many times you query the oracle. Another way to think about this is that the oracle can be queried at most once for a vertex. This setting is the most standard in the learning-augmented literature, and graph problems are often studied under persistent noise (see, e.g. [30, 17, 18, 58, 39, 5, 23, 33] and references therein). Our main result in this setting is a randomized algorithm that with high probability[3] achieves an $\widetilde{O}(\sqrt{\Delta})$ (multiplicative) approximation to the MIS in $O(m)$ time (Theorem 1).
- The **non-persistent noise** settings. In this setting, for each vertex v, we allow $\texttt{ORC}_{G,I^*}(v)$ to use *fresh randomness* for different queries. If we are allowed to make $O(n \log n)$ queries to the oracle in total, then we can trivially recover the entire set I^* with high probability by querying each vertex $O(\log n)$ times. The interesting case is when we are allowed to make only $O(n)$ queries, i.e., a number that is *asymptotically the same* as the persistent noise setting. Although the non-persistent noise setting is less frequently studied in the learning-augmented algorithm literature, it has recently sparked considerable interest in various problems [34, 35, 44]. In Appendix B of the full version, we show that it is easy to get an $O(\log n)$-approximation with $O(n)$ queries. Our main result considerably improved on the approximation factor: we show that we can indeed obtain an $O(1)$ approximation with $O(n)$ queries and $\widetilde{O}(m)$ runtime (Theorem 5).

Our results in the persistent noise setting hold assuming full independence, but it can be easily extended to the setting where oracle queries are assumed to use k-wise independent hash function for $k = O(\log n)$. Extending it to the pair-wise independent case is challenging as the failure probabilities in the concentration bounds are not enough for the application of a union bound.

1.2 Technical overview

The biggest challenge in leveraging the oracle information is distinguishing the case where $\texttt{ORC}_{G,I^*}(v)$ is indeed correct. In what follows, we give a high-level overview of our techniques describing how we can use the neighborhood information for this purpose. For the simplicity of the discussion, we always assume $\varepsilon = \Theta(1)$ in the technical overview.

[3] As standard in the literature, we use "with high probability" to denote a success probability of $1 - 1/\text{poly}(n)$.

Persistent noise setting

A natural approach in this setting would be to figure out the conditions in which a "yes" signal for a vertex v from the oracle implies $v \in I^*$, by aggregating signals from $N(v)$. However, such an idea is hard in the following sense. For a vertex v whose oracle query $\texttt{ORC}_{G,I^*}(v) = 1$, if there are many $u \in N(v)$ such that $\texttt{ORC}_{G,I^*}(u) = 1$, we can determine that v should *not* be in the MIS. However, the converse is not true: if a vertex v is *not* in I^*, it does *not* necessarily have many neighbors in I^*. As a result, simply aggregating neighborhood information might not be enough to determine the membership of a vertex in the MIS.

The key idea here is, instead of looking at the oracle answer for vertex v ($\texttt{ORC}_{G,I^*}(v)$), we look at what the oracle says for the *neighborhood* of the vertex v. This turns out to be a good enough signal to eliminate vertices that have *many* edges to the MIS I^*. Specifically, we can show that if v has $\widetilde{\Omega}(\sqrt{\deg(v)})$ edges to I^*, then the oracle queries for $N(v)$ contain enough information to identify such a vertex v. Upon removal of such vertices, the remaining vertices have a small degree ($\widetilde{O}(\sqrt{\Delta})$) to I^*, and a greedy independent set on the residual vertices gives a good approximation.

Non-persistent noise setting

Our algorithm for this setting is a bit more nuanced as we aim to minimize the query complexity to the oracle while aiming to achieve a good approximation. The starting point of our algorithm is from the viewpoint of the classical *pure exploration* algorithms in *multi-armed bandits (MABs)*. If we ignore the nature of MIS in our oracle, we can reduce to the following MABs problem: given n arms with mean rewards as either $\frac{2}{3}$ or $\frac{1}{3}$, find *all* the arms with mean reward $\frac{2}{3}$ with $O(n)$ arm pulls. It is well-known that one can find a *single* best arm with high constant probability in $O(n)$ queries. The question is, can we solve the problem by resorting to purely MABs algorithms, and simply ignoring the nature of the MIS?

It turns out that the above plan is not generally feasible. In particular, we note that returning the set of *all* arms with the higher reward is very similar to finding the *top-k arms* in the MABs literature (see, e.g. [40, 41, 15, 19, 55]). In general, it would require $\Omega(n \log k)$ arm pulls to obtain top-k arms with high constant probability ([41, 55]). In Appendix C of the full version, we provide lower bound results, showing that to find even $O(1)$ fraction of the high-reward arms in the instance distribution requires $\omega(n)$ queries. The lower bounds teach us that to obtain the desired query efficiency and approximation guarantee, we have to exploit the structure of the MIS.

To better understand the hardness and the insights of MABs algorithms on our problem, let us look at the elimination-based algorithm as in the classical algorithm of [27, 28]. The first idea we can try is to adapt the elimination algorithm to our problem. To this end, a natural idea is to perform elimination based on whether the mean empirical reward of an arm is more than $\frac{1}{2}$. More concretely, we maintain a pool $\tilde{I}$ of surviving vertices and use s_r as the number of queries to each vertex in round r with $s_1 = O(1)$. In round r, we can query $\texttt{ORC}_{G,I^*}(v)$ for s_r time for each $v \in \tilde{I}$. We then eliminate all vertices $v \in \tilde{I}$ whose number of "yes" answers is less than $s_r/2$, and recurse on the new $\tilde{I}$ to round $r+1$, for which we set $s_{r+1} = 1.5 s_r$.

Since the probability for any $v \notin I^*$ to survive decreases doubly-exponentially with the number of rounds, we can show that *all* vertices $v \notin I^*$ are eliminated after $O(\log \log n)$ rounds, and the total sample complexity on the *non-MIS* vertices is at most $O(n)$. Furthermore, the probability of losing any $v \in I^*$ decreases exponentially, we can argue that in the end, $\tilde{I}$

contains at least $\Omega(1)$ fraction of the vertices in I^*. Unfortunately, due to this fact, for each vertex $v \in I^* \cap \tilde{I}$, i.e., the vertices in the MIS that survive till the end, we need to pay for $2^{O(\log\log n)} = \text{polylog}\, n$ on the sample complexity. Therefore, this pure exploration algorithm only works when the size of I^* is upper-bounded by $n/\text{polylog}\, n$, and its worst-case guarantee is only a $\text{polylog}\, n$ approximation.

Note that a $\text{polylog}\, n$ approximation is far from what we want: after all, there is a trivial algorithm that achieves $O(\log n)$ approximation with $O(n)$ samples (see Appendix B of full version for details). Nevertheless, the existence of such an algorithm teaches us that the problematic case is when the MIS size is large and, in particular, *comparable* to the size of the non-MIS vertices. As such, a natural idea is to design an algorithm that handles the case when the numbers of the MIS and the non-MIS vertices are comparable, and fuse this algorithm with the elimination-based MABs procedure we discussed above.

The above idea is quite close to the final strategy we adapt, albeit we proceed differently for the roles of the two components. In particular, we use the pure exploration MABs algorithm not to output a set with vertex set $\tilde{I} \subseteq I^*$, but to output a set of vertex set $\tilde{I}$ whose *majority (but not necessarily all) of* vertices are in I^*. To this end, we use a more *conservative* elimination strategy than the ones in the line of [27, 28]: instead of increasing the number of samples by a multiplicative factor, we increase the number of samples in each round by an *additive* factor. In this way, we cannot guarantee that all the "wrong" arms are eliminated; however, we can argue that, since the probability for the non-MIS vertices to survive decreases exponentially, we have i). the number of samples used on the vertices in I^* is bounded by $O(n)$ *before* the size of $\tilde{I} \setminus I^*$ reduces to the size of $\tilde{I} \cap I^*$; and ii). the number of vertices in $\tilde{I} \cap I^*$ only decreases by a constant fraction. In this way, we can design an efficient procedure that eliminates the "surplus" non-MIS vertices to always create cases when the number of non-MIS vertices is smaller.

The final missing piece is the MIS algorithm that deals with the case when the number of MIS vertices takes the majority of the vertex set. Our algorithm to handle this case is to compute an *approximate vertex cover* of the graph and the remaining vertices will form an approximate independent set. It is a well-known fact that if we compute a *maximal matching* and take *all* their endpoints, it forms a 2-approximate vertex cover that covers all edges in the graph. Furthermore, since the size of the non-MIS vertices is small, there can be only a *limited number* of vertices $v \in I^*$ that can be counted in the vertex cover. As such, we can simply *remove* these vertices from the graph. The rest of the graph would form an independent set, and since we remove at most a constant fraction of vertices from I^* throughout the two phases, we get an $O(1)$ approximation.

2 Preliminaries

Notation

For a graph $G = (V, E)$, we use $\deg(v)$ and $N(v)$ for each vertex $v \in V$ to denote the degree and neighborhood of v, respectively. We use $G[U]$ for any set U of vertices to denote the induced subgraph of G on U.

We let I^* denote a fixed maximum independent set of the graph G. We let $N_{I^*}(v) = N(v) \cap I^*$ be the set of neighbors of the vertex v in the independent set and let $\deg_{I^*}(v) := |N_{I^*}(v)|$ be its size. Furthermore, we let $\widetilde{N_{I^*}}(v)$ be the set of neighbors of the vertex v that are claimed to be in the independent set by the oracle and let $\widetilde{\deg}_{I^*}(v)$ be its size.

For the purpose of conciseness, we defer the technical preliminaries to Appendix A.

3 An Algorithm in the Persistent Noise Setting

In this section we present an algorithm for the learning-augmented MIS problem with persistent noise. Formally we prove the following

▶ **Theorem 1.** *There exists a randomized algorithm that given*

i) *an input graph $G = (V, E)$ with maximum degree Δ and*

ii) *an MIS oracle $\texttt{ORC}_{G,I^*}$ with* persistent noise *for an unknown maximum independent set I^*,*

in $O(m)$ time outputs an independent set I such that $|I| \geqslant \frac{\varepsilon}{12} \cdot (\Delta \ln n)^{-0.5} \cdot |I^*|$ with high probability.

We dedicate the remainder of this section to the proof of Theorem 1. We start with the assumption that $\varepsilon \leqslant 1/4$ (we can do this for any constant > 0). This assumption is needed for technical reasons. If $\varepsilon > 1/4$, then it is easy to simulate an oracle with $\varepsilon = 1/4$ by flipping the oracle answer with probability $p = \frac{\varepsilon - 1/4}{1/2+\varepsilon}$ ($p \geqslant 0$ since $\varepsilon > 1/4$). If we do this then the probability that the oracle gives the incorrect answer is $(1/2 - \varepsilon) + p \cdot (1/2 + \varepsilon) = 1/4$ which is exactly what we wanted. Note that the final bound we get on the approximation factor now changes by a factor of at most 2. This is because when $\varepsilon > 1/4$ we are replacing it with an oracle for $\varepsilon = 1/4$ and the approximation factor linearly depends on ε.

The algorithm and analysis

We now state our algorithm.

Algorithm 1 An algorithm for MIS in persistent noise setting.

Input: A graph $G = (V, E)$ with maximum degree Δ that contains an unknown maximum independent set I^*; an MIS oracle $\texttt{ORC}_{G,I^*}$ in the persistent noise setting

Output: A set of vertices I such that I forms an independent set and $|I| \geqslant \frac{\varepsilon}{3} \cdot (\Delta \ln n)^{-0.5} \cdot |I^*|$.

Parameters: $s_v := (1/2 - \varepsilon)\deg(v) + 6\sqrt{\ln n} \cdot (1/2 - \varepsilon)\sqrt{\deg(v)}$.

1. Calculate $\widetilde{\deg}_{I^*}(v)$ for all vertices $v \in V$.
2. Let L be the set of vertices where $\deg(v) \leqslant 36 \ln n$ for $v \in V$.
3. Let S be the set of vertices where $\widetilde{\deg}_{I^*}(v) \leqslant s_v$ for $v \in V \setminus L$.
4. Output the greedy MIS I on $G[S \cup L]$.

We first show that if $v \in I^*$, the number of "yes" answers in $N(v)$ cannot be too high.

▷ Claim 2. If $v \in I^* \setminus L$ then with high probability, $\widetilde{\deg}_{I^*}(v) \leqslant (1/2 - \varepsilon)\deg(v) + 6\sqrt{\ln n} \cdot (1/2 - \varepsilon)\sqrt{\deg(v)}$.

Proof. If $v \in I^*$ then $\deg_{I^*}(v) = 0$ which means that the expected size of $\widetilde{\deg}_{I^*}(v)$ is $(1/2 - \varepsilon)\deg(v)$. Since we assume complete independence for the oracle we can use the Chernoff bound to get concentration.

Let $X_i = 1$ if i^{th} neighbor is claimed to be in I^* by the oracle where $i \in [\deg(v)]$. Observe that $\widetilde{\deg}_{I^*}(v) = \sum_i X_i$ is the number of neighbors that claim to be in I^*. We know $\mu = \mathbb{E}\left[\widetilde{\deg}_{I^*}(v)\right] = (1/2 - \varepsilon)\deg(v)$. Using the Chernoff (Proposition 12) bound with $\delta_v = 6\left(\frac{\ln n}{\deg(v)}\right)^{0.5} \leqslant 1$:

$$\Pr\left(\widetilde{\deg}_{I^*}(v) > (1 + \delta_v)\mu\right) \leqslant \exp\left(-\frac{\delta_v^2 \cdot \mu}{3}\right) \leqslant n^{-3}. \qquad \text{(since } \varepsilon \leqslant 1/4\text{)}$$

Notice that as $\deg(v)$ gets larger we get better concentration. ◁

Note that Claim 2 does *not* rule out the case that a vertex $v \in V \setminus I^*$ and has very few neighbors in I^*. Nevertheless, it tells us that if we simply eliminate the vertices that "block" a large number of neighbors in I^*, we will not mistakenly drop vertices in I^*.

Next, we show that if a vertex v has many neighbors in I^* i.e. $\deg_{I^*}(v)$ is large then $\widetilde{\deg}_{I^*}(v)$ should also be large and hence we should be able to detect such a vertex $v \notin I^*$.

▷ Claim 3. If $v \notin I^*$ and $\deg_{I^*}(v) \geqslant (3/\varepsilon)\sqrt{\ln n}\sqrt{\deg(v)}$ then with high probability, $\widetilde{\deg}_{I^*}(v) > (1/2-\varepsilon)\deg(v) + 6\sqrt{\ln n}\cdot(1/2-\varepsilon)\sqrt{\deg(v)}$.

Proof. If $v \notin I^*$ and $\deg_{I^*}(v) = k$ then the expected size of $\widetilde{\deg}_{I^*}(v)$ is

$$\mu = \mathbb{E}\left[\widetilde{\deg}_{I^*}(v)\right] = k(1/2+\varepsilon) + (\deg(v)-k)(1/2-\varepsilon) = (1/2-\varepsilon)\deg(v) + 2\varepsilon k.$$

We now use the Chernoff bound (Proposition 11) with $t = \varepsilon k$ to get concentration:

$$\begin{aligned}\Pr\left(\widetilde{\deg}_{I^*}(v) < \mu - t\right) &\leqslant \exp\left(-2t^2/\deg(v)\right)\\ &= \exp\left(-2\varepsilon^2k^2/\deg(v)\right)\\ &\leqslant n^{-3}. && \text{(using the lower bound on k)}\end{aligned}$$

Thus, with high probability we have:

$$\begin{aligned}\widetilde{\deg}_{I^*}(v) &\geqslant \mu - \varepsilon k\\ &= (1/2-\varepsilon)\deg(v) + \varepsilon k\\ &= (1/2-\varepsilon)\deg(v) + 3\sqrt{\ln n}\sqrt{\deg(v)}\\ &> (1/2-\varepsilon)\deg(v) + 6\sqrt{\ln n}\cdot(1/2-\varepsilon)\sqrt{\deg(v)}\,.\end{aligned}$$ ◁

We can conclude that the events in Claim 2 and Claim 3 happen with high probability by a union bound over all vertices.

Finalizing the proof of Theorem 1. Calculating $\widetilde{\deg}_{I^*}(v)$ for all vertices $v \in V$ and finding set S takes $O(m)$ time. The greedy MIS can also be computed in $O(m)$ time.

We first condition on the events in Claim 2 and Claim 3 for all vertices (this happens with high probability). Notice that for all vertices in $v \in S$ we have $\widetilde{\deg}_{I^*}(v) \leqslant s_v$. By Claim 2 all vertices in I^* are in S. By Claim 3 we know that any non-MIS vertices v that are in S have $\deg_{I^*}(v) \leqslant (3/\varepsilon)\sqrt{\ln n}\sqrt{\deg(v)} \leqslant (6/\varepsilon)\sqrt{\Delta\ln n}$. Also, vertices in L have $\deg_{I^*}(v) \leqslant \deg(v) = \sqrt{\deg(v)}\cdot\sqrt{\deg(v)} \leqslant \sqrt{\Delta}\sqrt{36\ln n} \leqslant (6/\varepsilon)\sqrt{\Delta\ln n}$.

This means that when we run the greedy MIS algorithm and pick a non-MIS vertex, we eliminate at most $(6/\varepsilon)\sqrt{\Delta\ln n}$ vertices in I^*. Thus, we have $|I| \geqslant \frac{\varepsilon}{6}\cdot(\Delta\ln n)^{-0.5}\cdot|I^*|$. Finally, because of the assumption on ε ($\varepsilon \leqslant 1/4$), we lose a factor of at most 2 in the approximation, giving us the final bound $|I| \geqslant \frac{\varepsilon}{12}\cdot(\Delta\ln n)^{-0.5}\cdot|I^*|$. ◀

▶ Remark 4. We assume that we have complete independence between the oracle queries for the vertices. But we can get essentially the same result (up to constants) when the oracle answers the queries using a k-wise independent hash function instead of a completely random function for $k = O(\log n)$.

This holds because we use Proposition 13 with $k = O(\log n)$ instead of the Chernoff bound (Proposition 12). The min in the exponent always picks the second term because k is large enough and so we get something very similar to the Chernoff bound in Proposition 12 where the exponent only differs by some constants. Thus, the approximation we get will be a small constant factor worse but will remain the same asymptotically.

4 An Algorithm in the Non-persistent Noise Setting

In this section, we consider algorithms in the *non-persistent noise setting (MABs setting)* of the MIS oracle, i.e., the algorithm can access the learning-augmented MIS oracle with *fresh randomness* for each query of a vertex v. The formal statement of our main result in this setting is as follows.

▶ **Theorem 5.** *There exists a randomized algorithm that given a parameter $\delta \in (0, 1)$ and*

i) *an input graph $G = (V, E)$ with a maximum independent set I^*; and*

ii) *an MIS oracle $\texttt{ORC}_{G,I^*}$ in the* non-persistent noise setting,

with probability at least $(1 - \delta)$, in $O(m \log n)$ time and $\frac{30n}{\varepsilon^2} \cdot \log \frac{1}{\delta}$ total queries to $\texttt{ORC}_{G,I^*}$, computes a set I such that $|I| \geqslant \frac{48}{50} \cdot |I^*|$.

We dedicate the remainder of this section to the proof of Theorem 5.

The algorithm

As we have discussed in our high-level overview, our algorithm proceeds in two phases. In the first phase, our algorithm focuses on eliminating most of the vertices in the non-MIS vertex set. Then, in the second phase, we show that a good approximation to vertex cover is enough to get a good approximation to the independent set. We can easily find a 2-approximate vertex cover in $O(m)$ time by computing a maximal matching and picking all its endpoints. The detailed description of the algorithm is as follows.

Algorithm 2 An algorithm for MIS in non-persistent noise setting.

Input: A graph $G = (V, E)$ that contains an unknown maximum independent set I^*; an MIS oracle $\texttt{ORC}_{G,I^*}$ in the multi-armed bandit setting; a confidence parameter $\delta \in (0, 1)$.

Output: A set of vertices I such that I forms an independent set and $|I| = O(|I^*|)$.

Parameters: $q_r = \frac{4}{\varepsilon^2} \cdot (r + \log \frac{1}{\delta})$.

- Maintain a set of V_r with the initialization $V_0 \leftarrow V$.
- For $r = 1$ to ∞, do the following:
  1. **Elimination phase:**
     - For each vertex $v \in V_{r-1}$:
       - **a.** Query v for q_r times.
       - **b.** *Remove* v from V_{r-1} if the number of 1 returned by $\texttt{ORC}_{G,I^*}(v)$ ("yes" answers) is less than $q_r/2$.
     - Let the updated vertex set be V_r, i.e., V_r is a subset of vertices of V_{r-1} that gets at least $q_r/2$ "yes" answers from $\texttt{ORC}_{G,I^*}$ (v).
  2. **Vertex Cover phase:**
     - **a.** Compute a 2-approximate vertex cover U_r of the induced subgraph $G[V_r]$.
     - **b.** Let $I_r \leftarrow V_r \setminus U_r$.
  3. Maintain the set I with the maximum size among all I_r's, i.e., let $I \leftarrow I_r$ if I_r is larger than I and keep I unchanged otherwise.
  4. If the total number of $\texttt{ORC}_{G,I^*}$ queries is more than $30 \cdot \frac{n}{\varepsilon^2} \cdot \log \frac{1}{\delta}$ then terminate and return the currently maintained I.

Note that since we do *not* necessarily know the actual size of I^*, we compute a vertex cover after every elimination phase and simply output the independent set with the largest size throughout the process.

The analysis

We now proceed to the analysis of the algorithm. Before diving into the main lemmas, we first show some straightforward technical claims that characterize the behavior of the MIS and non-MIS vertices in the elimination phase. We first show that the probabilities of an MIS vertex being eliminated and a non-MIS vertex surviving in round r are both small.

▷ Claim 6. The following statements are true:

1. Let $v \in V_{r-1} \cap I^*$; then, the probability that v is removed from V_r is at most $\frac{1}{100} \cdot \frac{\delta}{4^r}$.
2. Let $v \in V_{r-1} \setminus I^*$; then, the probability that v is *not* removed from V_r is at most $\frac{1}{100} \cdot \frac{\delta}{4^r}$.

Proof. We prove this claim by applying the Chernoff bound in Proposition 11. For any vertex $v \in I^*$, let the random variable $X_v^i = 1$ if the i^{th} query for vertex v is a "yes" and $X_v^i = 0$ otherwise for $i \in [q_r]$. Observe that $X_v = \sum_i X_v^i$ is the number of "yes" answers returned by $\texttt{ORC}_{G,I^*}(v)$ out of the q_r queries. Clearly, we have that $\mathbb{E}[X_v] = (1/2 + \varepsilon) \cdot q_r$, and X_v is a summation of the independent indicator random variables so, we can apply Proposition 11 to show that

$$\begin{aligned}
\Pr\left(X_v < \frac{q_r}{2}\right) &= \Pr\left(X_v - \mathbb{E}[X_v] \leqslant -\varepsilon \cdot q_r\right) \\
&\leqslant \exp\left(-2 \cdot \varepsilon^2 \cdot q_r\right) && \text{(applying Proposition 11)} \\
&= \exp\left(-8r - 8\log\frac{1}{\delta}\right) && \text{(by the definition of } q_r\text{)} \\
&\leqslant \exp(-6) \cdot \exp(-2r) \cdot \exp\left(-8\log\frac{1}{\delta}\right) \\
&\leqslant \frac{1}{100} \cdot \frac{\delta}{4^r}.
\end{aligned}$$

Note that the vertices in $v \in I^* \cap V_{r-1}$ are in I^*. Therefore, we can get the desired statement for $v \in I^* \cap V_{r-1}$.

We can similarly define Y_v for the number of "yes" answers returned by $\texttt{ORC}_{G,I^*}(v)$ with q_r queries for a vertex $v \in V \setminus I^*$. Here, we have that $\mathbb{E}[Y_v] = (1/2 - \varepsilon)q_r$. As such, we have that

$$\begin{aligned}
\Pr\left(Y_v \geqslant \frac{q_r}{2}\right) &= \Pr\left(Y_v - \mathbb{E}[Y_v] \geqslant \varepsilon \cdot q_r\right) \\
&\leqslant \exp\left(-2 \cdot \varepsilon^2 \cdot q_r\right) && \text{(applying Proposition 11)} \\
&= \exp\left(-8r - 8\log\frac{1}{\delta}\right) && \text{(by the definition of } q_r\text{)} \\
&\leqslant \exp(-6) \cdot \exp(-2r) \cdot \exp\left(-8\log\frac{1}{\delta}\right) \\
&\leqslant \frac{1}{100} \cdot \frac{\delta}{4^r}.
\end{aligned}$$

This gives us the desired statement for $v \in V_{r-1} \setminus I^*$ as well. ◁

We now prove the main technical lemma of our algorithm that helps eventually prove Theorem 5. In what follows, we will denote the size of I^* as αn for some $\alpha \in (0, 1)$. Our main lemma for the elimination phase is as follows.

▶ **Lemma 7.** *Let $|I^*| = \alpha n$ for some $\alpha \in (0,1)$ and $\tilde{r} = 1 + \log\frac{1}{\alpha}$. With probability at least $1 - \delta$, the following statements about Algorithm 2 are true:*

I) *The number of vertices in $V_{\tilde{r}}$ that are* not *in I^* is at most $\alpha n/100$, i,e,*

$$|V_{\tilde{r}} \setminus I^*| \leqslant \frac{\alpha n}{100}.$$

II) *The number of vertices in $V_{\tilde{r}}$ that are in I^* is at least $^{49}/_{50} \cdot \alpha n$, i.e.,*

$$|V_{\tilde{r}} \cap I^*| \geqslant \frac{49}{50} \cdot \alpha n.$$

III) *The* total *number of $\texttt{ORC}_{G,I^*}$ queries in the first $\tilde{r}$ rounds is at most $^{30n}/_{\varepsilon^2} \cdot \log 1/\delta$, i.e.,*

$$\sum_{r=1}^{\tilde{r}} |V_{r-1}| \cdot q_r \leqslant 30 \cdot \frac{n}{\varepsilon^2} \cdot \log \frac{1}{\delta}.$$

Note that in the above, $|V_{r-1}| \cdot q_r$ is exactly the number of queries used in round r.

Proof. We prove the statements in order.

Proof of i). Note that by Claim 6, the probability that a vertex in $V \setminus I^*$ survives round r is at most $\frac{1}{100} \cdot \frac{\delta}{4^r}$. As such, we have that

$$\begin{aligned}
\mathbb{E}\left[|V_{\tilde{r}} \setminus I^*|\right] &= \sum_{v \in V_{\tilde{r}-1} \setminus I^*} \Pr\left(v \text{ survives round } \tilde{r}\right) \\
&= \sum_{v \in V \setminus I^*} \Pr\left(v \text{ survives all rounds till } \tilde{r}\right) \\
&= \sum_{v \in V \setminus I^*} \prod_{i=1}^{\tilde{r}} \Pr\left(v \text{ survives round } i \mid v \text{ survives all rounds till } i-1\right) \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{(All rounds are independent)} \\
&\leqslant \sum_{v \in V \setminus I^*} \prod_{i=1}^{\tilde{r}} \frac{\delta}{100} \cdot \frac{1}{4^i} \\
&\leqslant n \cdot \left(\frac{\delta}{100}\right)^{\tilde{r}} \cdot \left(\frac{1}{4}\right)^{\binom{\tilde{r}}{2}} \\
&\leqslant \frac{\delta n}{100} \cdot \left(\frac{1}{4}\right)^{\tilde{r}} \\
&\leqslant \frac{\alpha \cdot n \cdot \delta}{400}. \qquad \text{(using } \alpha \in (0,1))
\end{aligned}$$

Therefore, by Markov inequality, we have

$$\Pr\left(|V_r \setminus I^*| > \frac{\alpha n}{100}\right) \leqslant \frac{\delta}{4}$$

as desired.

Proof of ii). By Claim 6, the probability that a vertex v is eliminated in round r is at most $\frac{\delta}{100} \cdot \frac{1}{4^r}$. We analyze the number of vertices in I^* that are eliminated by round r. We can show that the expected value is

$$\begin{aligned}
\mathbb{E}\left[|I^* \setminus V_{\tilde{r}}|\right] &= \sum_{v \in I^*} \Pr\left(v \text{ is eliminated by round } \tilde{r}\right) \\
&\leqslant \sum_{v \in I^*} \sum_{i=1}^{\tilde{r}} \Pr\left(v \text{ is eliminated in round } i\right) \qquad \text{(Union Bound)} \\
&\leqslant \sum_{v \in I^*} \sum_{i=1}^{\tilde{r}} \frac{\delta}{100} \cdot \frac{1}{4^i} \\
&\leqslant (\alpha n) \cdot \frac{\delta}{100} \cdot \frac{1}{3}. \qquad \text{(Geometric Sum)}
\end{aligned}$$

Therefore, by a simple Markov bound, we have that

$$\Pr\left(|I^* \setminus V_r| > \frac{\alpha n}{50}\right) \leqslant \frac{\delta}{6}.$$

Thus, with probability at least $1-\delta/6$ we have $|I^* \cap V_{\tilde{r}}| \geqslant \frac{49}{50}\cdot \alpha n$.

Proof of iii). Note that we are proving this bound holds even if we remove the termination condition from the algorithm. This will show that we will reach round $\tilde{r}$ with high probability. We first condition on the events in the proofs of i) and ii). Note that, unlike the standard analysis of elimination-based algorithms, here, we cannot directly upper-bound the total number of queries each round. Instead, we separately analyze the number of queries induced by the vertices in I^* and $V \setminus I^*$.

We first analyze the number of queries induced by the vertices in $V \setminus I^*$. Let us define $X_{\neg I^*}$ as the total number of queries induced by the non-MIS vertices. Similarly, we can define $X^r_{\neg I^*}$ as the queries induced by the non-MIS vertices at round r. Thus, we have that

$$\begin{aligned}
\mathbb{E}\left[X_{\neg I^*}\right] &= \sum_{v\in V-I^*}\sum_{i=1}^{\tilde{r}} \Pr\left(v \text{ survives till round } i\right)\cdot q_i \\
&\leqslant n\sum_{i=1}^{\tilde{r}} q_i \prod_{j=1}^{i} \Pr\left(v \text{ survives round } j \mid v \text{ survives till round } j-1\right) \\
&\leqslant n\sum_{i=1}^{\tilde{r}} q_i \prod_{j=1}^{i} \frac{\delta}{100}\cdot\frac{1}{4^j} && \text{(Claim 6)} \\
&\leqslant n\sum_{i=1}^{\tilde{r}} \left(\frac{\delta}{100}\right)^i\cdot\left(\frac{1}{4}\right)^{\binom{i}{2}}\cdot q_i \\
&= n\sum_{i=1}^{\tilde{r}} \left(\frac{\delta}{100}\right)^i\cdot\left(\frac{1}{4}\right)^{\binom{i}{2}}\cdot \frac{4}{\varepsilon^2}\cdot(i+\log 1/\delta) \\
&\leqslant \frac{4\delta n}{100\varepsilon^2}\sum_{i=1}^{\tilde{r}}\left(\frac{1}{4}\right)^i\cdot(i+\log 1/\delta) && \text{(Since } \delta\leqslant 1\text{)} \\
&\leqslant \frac{4\delta n}{100\varepsilon^2}\left(1+\log 1/\delta\right). && \text{(using properties of geometric sums)}
\end{aligned}$$

Therefore, by Markov inequality, we can show that

$$\Pr\left(X_{\neg I^*} > \frac{2n}{5\varepsilon^2}\log 1/\delta\right) \leqslant \delta/5.$$

We now analyze the queries induced by the vertices in I^*. Similar to the case of the non-MIS analysis, let us define X_{I^*} as the total number of queries induced by the MIS vertices. We will trivially upper bound X_{I^*} in the following way:

$$\begin{aligned}
X_{I^*} &\leqslant \alpha n\sum_{i=1}^{\tilde{r}} q_i \\
&= \alpha n\sum_{i=1}^{\tilde{r}} \frac{4}{\varepsilon^2}\cdot\left(i+\log\frac{1}{\delta}\right)
\end{aligned}$$

$$\leqslant \frac{4\alpha n}{\varepsilon^2} \cdot \left(\tilde{r}^2 + \tilde{r} \cdot \log \frac{1}{\delta}\right)$$
$$\leqslant \frac{4\alpha n}{\varepsilon^2} \cdot \left(1 + (\log 1/\alpha)^2 + \lg 1/\alpha \cdot (2 + \log 1/\delta) + \log 1/\delta\right)$$
$$\leqslant \frac{4n}{\varepsilon^2} \cdot (5 + 2\log 1/\delta) \qquad \text{(using } \alpha \cdot \lg \tfrac{1}{\alpha} \leqslant 1 \text{ and } \alpha \cdot \lg^2 \tfrac{1}{\alpha} \leqslant 2 \text{ for any } \alpha \in (0,1))$$

We can then add the number of queries used by $X_{\neg I^*}$ and X_{I^*} to get the desired sample complexity bound of $\frac{30n}{\varepsilon^2} \cdot \log 1/\delta$.

Finally, we can apply a union bound over the failure probabilities of the events in the proofs of *i*), *ii*), and *ii*) to argue that with probability at least $1 - \delta$, all the statements hold.

Lemma 7 ◄

We now proceed to show the guarantee of the matching and MIS phase. Our main lemma for this part is as follows.

► **Lemma 8.** *Let $V_r \subseteq V$ be any subset of vertices in Algorithm 2. Furthermore, assume that the number of MIS vertices in V_r is at least* 50 *times the number of non-MIS vertices in V_r, i.e.,*

$$|V_r \cap I^*| \geqslant 50 \cdot |V_r \setminus I^*| .$$

Then, the set I_r returned by Algorithm 2 is a valid independent set, and we have

$$|I_r| \geqslant \frac{49}{50} \cdot |V_r \cap I^*| .$$

Proof. Recall that we compute a 2-approximate vertex cover U_r in the vertex cover phase. We know that the complement $I_r \leftarrow V_r \setminus U_r$ is an independent set. This is because all edges of the graph are incident on the vertex cover so the remaining vertices form an independent set.

We know that $V_r \setminus I^*$ is a vertex cover since $V_r \cap I^*$ is an independent set. Thus, we have

$$|I_r| = |V_r| - |U_r| \qquad \text{(by definition)}$$
$$\geqslant |V_r \cap I^*| + |V_r \setminus I^*| - 2\,|V_r \setminus I^*| \qquad \text{(since } U_r \text{ is a 2-approximation)}$$
$$\geqslant |V_r \cap I^*| - \frac{1}{50} \cdot |V_r \cap I^*| \qquad \text{(using the assumption)}$$
$$= \frac{49}{50} \cdot |V_r \cap I^*|$$

Lemma 8 ◄

The final missing piece is the *efficiency* of the algorithm. We now prove that the algorithm is efficient both in time and the number of $\texttt{ORC}_{G,I^*}$ oracle queries.

► **Lemma 9.** *Algorithm 2 runs in $O(m \log n)$ time and uses at most $\frac{30n}{\varepsilon^2} \cdot \log \frac{1}{\delta}$ queries on $\texttt{ORC}_{G,I^*}$.*

Proof. The query complexity is by the design of the algorithm as we terminate upon using more than $30 \cdot \frac{n}{\varepsilon^2} \cdot \log \frac{1}{\delta}$ queries.

For the running time, note that in each iteration of r, we only need to: *i*). take the majority for all queried vertices, which can be maintained in $O(n)$ time; and *ii*). compute a greedy matching and remove the vertices, which takes $O(m)$ time. By Lemma 7, the process terminates in $O(\log \frac{1}{\alpha}) = O(\log n)$ time ($\alpha \geqslant \frac{1}{n}$ since there has to be at least one vertex in I^*). Therefore, the entire algorithm takes $O(m \log n)$ time in total. ◄

Finalizing the proof of Theorem 5. The query efficiency is by the design of the algorithm, and the running time simply follows from Lemma 9. For the approximation guarantee, note that by Lemma 7, we will proceed to round $\tilde{r} = 10 \log \frac{1}{\alpha}$, at which point we will have $|V_{\tilde{r}} \cap I^*| \geqslant \frac{49}{50} \cdot \alpha n$ and $|V_r \cap I^*| \geqslant 50 \cdot |V_r \setminus I^*|$. Therefore, by Lemma 8, the returned $I_{\tilde{r}}$ is of size at least

$$|I_{\tilde{r}}| \geqslant \frac{49}{50} \cdot |V_{\tilde{r}} \cap I^*| \geqslant \frac{49}{50} \cdot \frac{49}{50} \cdot \alpha n,$$

which gives us the desired 48/50 approximation. ◀

▶ Remark 10. We aim to get the $O(1)$ approximation in our algorithm and analysis. However, we remark that we can get both non-asymptotic and asymptotic trade-offs between the number of queries and the approximation factor. For the non-asymptotic trade-off (i.e., using more queries to get a better constant approximation), we can increase the leading constant on the sample complexity, and obtain the approximation with a larger constant. For the asymptotic trade-off, we can perform the simple trick by sampling k vertices uniformly at random and running Algorithm 2 on the sampled vertices. This will give us an $O(\frac{k}{n})$-approximation algorithm with $O(\frac{k}{\varepsilon^2} \cdot \log \frac{1}{\delta})$ queries as long as $\alpha k = \Omega(\log n)$.

5 Discussion and Open Problems

We discussed learning-augmented algorithms for the Maximum Independent Set problem in this paper. Our main results include algorithms for both persistent and non-persistent noise settings, demonstrating that a learning-augmented oracle could lead to MIS algorithms with considerably better efficiency. There are several intriguing open problems following our work.

- For the **persistent noise** setting, the main open question is whether we could beat the $\widetilde{\Theta}(\sqrt{\Delta}/\varepsilon)$ approximation bound with the same oracle. We do not have any lower bounds for the persistent noise setting in this paper, and it is unclear what type of techniques could be used to prove lower bounds for learning-augmented algorithms.
- For the **non-persistent noise** setting, our algorithm matches the *asymptotically* optimal approximation factor using $O(n)$ queries. In Appendix C of full version, we also proved that we cannot obtain the same results by only querying the oracle (and *not* looking into the graph). An open problem here is that if we want to recover a $1 - o(1)$ fraction of the MIS vertices, how many queries do we need? We suspect there is a lower bound on the number of queries (e.g., $\omega(n)$), but it is not immediately clear how to prove it.
- We can also ask about **sublinear** number of queries on the oracle $\mathtt{ORC}_{G,I^*}$, i.e., if we make $o(n)$ queries on the oracle, what is the best we can do for both persistent and non-persistent noise settings? Currently, our algorithms in both settings require $\Omega(n)$ queries to the oracle.
- Finally, for the **practical** aspect of the algorithms, we believe the oracles are possible to implement in practice. For instance, if we have features on the nodes, it is possible to use forward-pass graph convolution networks (GCNs), and simply run greedy in each "cluster" of nodes whose final features are sufficiently similar. Exploring practical oracles for this purpose would also be an interesting problem to resolve.

References

1 Anders Aamand, Justin Y. Chen, Huy Lê Nguyen, Sandeep Silwal, and Ali Vakilian. Improved frequency estimation algorithms with and without predictions. *CoRR*, abs/2312.07535, 2023. doi:10.48550/arXiv.2312.07535.

2 Sungsoo Ahn, Younggyo Seo, and Jinwoo Shin. Learning what to defer for maximum independent sets. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 134–144. PMLR, 2020. URL: `http://proceedings.mlr.press/v119/ahn20a.html`.

3 Vladimir E. Alekseev, Vadim V. Lozin, Dmitriy S. Malyshev, and Martin Milanic. The maximum independent set problem in planar graphs. In Edward Ochmanski and Jerzy Tyszkiewicz, editors, *Mathematical Foundations of Computer Science 2008, 33rd International Symposium, MFCS 2008, Torun, Poland, August 25-29, 2008, Proceedings*, volume 5162 of *Lecture Notes in Computer Science*, pages 96–107. Springer, 2008. `doi:10.1007/978-3-540-85238-4_7`.

4 Diogo V Andrade, Mauricio GC Resende, and Renato F Werneck. Fast local search for the maximum independent set problem. *Journal of Heuristics*, 18:525–547, 2012.

5 Antonios Antoniadis, Hajo Broersma, and Yang Meng. Online graph coloring with predictions. *arXiv preprint*, 2023. `arXiv:2312.00601`.

6 Yossi Azar, Debmalya Panigrahi, and Noam Touitou. Discrete-smoothness in online algorithms with predictions. *Advances in Neural Information Processing Systems*, 36, 2024.

7 Eric Balkanski, Vasilis Gkatzelis, Xizhi Tan, and Cherlin Zhu. Online mechanism design with predictions. *CoRR*, abs/2310.02879, 2023. `doi:10.48550/arXiv.2310.02879`.

8 Etienne Bamas, Andreas Maggiori, and Ola Svensson. The primal-dual method for learning augmented algorithms. *Advances in Neural Information Processing Systems*, 33:20083–20094, 2020.

9 Siddhartha Banerjee, Vincent Cohen-Addad, Anupam Gupta, and Zhouzi Li. Graph searching with predictions. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*, volume 251 of *LIPIcs*, pages 12:1–12:24. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. `doi:10.4230/LIPICS.ITCS.2023.12`.

10 Ravi Boppana and Magnús M Halldórsson. Approximating maximum independent sets by excluding subgraphs. *BIT Numerical Mathematics*, 32(2):180–196, 1992.

11 Ravi B Boppana. Eigenvalues and graph bisection: An average-case analysis. In *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*, pages 280–285. IEEE, 1987.

12 Nicolas Bourgeois, Bruno Escoffier, Vangelis Th. Paschos, and Johan M. M. van Rooij. Fast algorithms for max independent set. *Algorithmica*, 62(1-2):382–415, 2012. `doi:10.1007/S00453-010-9460-7`.

13 Jan van den Brand, Sebastian Forster, Yasamin Nazari, and Adam Polak. On dynamic graph algorithms with predictions. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3534–3557. SIAM, 2024.

14 Lorenzo Brusca, Lars C. P. M. Quaedvlieg, Stratis Skoulakis, Grigorios Chrysos, and Volkan Cevher. Maximum independent set: Self-training through dynamic programming. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL: `http://papers.nips.cc/paper_files/paper/2023/hash/7fe3170d88a8310ca86df2843f54236c-Abstract-Conference.html`.

15 Wei Cao, Jian Li, Yufei Tao, and Zhize Li. On top-k selection in multi-armed bandits and hidden bipartite graphs. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1036–1044, 2015. URL: `https://proceedings.neurips.cc/paper/2015/hash/ab233b682ec355648e7891e66c54191b-Abstract.html`.

16 Parinya Chalermsook and Julia Chuzhoy. Maximum independent set of rectangles. In Claire Mathieu, editor, *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2009, New York, NY, USA, January 4-6, 2009*, pages 892–901. SIAM, 2009. `doi:10.1137/1.9781611973068.97`.

17 Justin Chen, Sandeep Silwal, Ali Vakilian, and Fred Zhang. Faster fundamental graph algorithms via learned predictions. In *International Conference on Machine Learning*, pages 3583–3602. PMLR, 2022.

18 Justin Y Chen, Talya Eden, Piotr Indyk, Honghao Lin, Shyam Narayanan, Ronitt Rubinfeld, Sandeep Silwal, Tal Wagner, David P Woodruff, and Michael Zhang. Triangle and four cycle counting with predictions in graph streams. *arXiv preprint*, 2022. arXiv:2203.09572.

19 Lijie Chen, Jian Li, and Mingda Qiao. Nearly instance optimal sample complexity bounds for top-k arm selection. In Aarti Singh and Xiaojin (Jerry) Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pages 101–110. PMLR, 2017. URL: http://proceedings.mlr.press/v54/chen17a.html.

20 Jakub Chledowski, Adam Polak, Bartosz Szabucki, and Konrad Tomasz Zolna. Robust learning-augmented caching: An experimental study. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 1920–1930. PMLR, 2021. URL: http://proceedings.mlr.press/v139/chledowski21a.html.

21 Nicolas Christianson, Tinashe Handina, and Adam Wierman. Chasing convex bodies and functions with black-box advice. In Po-Ling Loh and Maxim Raginsky, editors, *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178 of *Proceedings of Machine Learning Research*, pages 867–908. PMLR, 2022. URL: https://proceedings.mlr.press/v178/christianson22a.html.

22 Julia Chuzhoy and Alina Ene. On approximating maximum independent set of rectangles. In Irit Dinur, editor, *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 820–829. IEEE Computer Society, 2016. doi:10.1109/FOCS.2016.92.

23 Vincent Cohen-Addad, Tommaso d'Orsi, Anupam Gupta, Euiwoong Lee, and Debmalya Panigrahi. Max-cut with ϵ-accurate predictions. *CoRR*, abs/2402.18263, 2024. doi:10.48550/arXiv.2402.18263.

24 Jakob Dahlum, Sebastian Lamm, Peter Sanders, Christian Schulz, Darren Strash, and Renato F. Werneck. Accelerating local search for the maximum independent set problem. In Andrew V. Goldberg and Alexander S. Kulikov, editors, *Experimental Algorithms - 15th International Symposium, SEA 2016, St. Petersburg, Russia, June 5-8, 2016, Proceedings*, volume 9685 of *Lecture Notes in Computer Science*, pages 118–133. Springer, 2016. doi:10.1007/978-3-319-38851-9_9.

25 Devdatt P Dubhashi and Alessandro Panconesi. *Concentration of measure for the analysis of randomized algorithms*. Cambridge University Press, 2009.

26 Jon C. Ergun, Zhili Feng, Sandeep Silwal, David P. Woodruff, and Samson Zhou. Learning-augmented k-means clustering. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL: https://openreview.net/forum?id=X8cLTHexYyY.

27 Eyal Even-Dar, Shie Mannor, and Yishay Mansour. PAC bounds for multi-armed bandit and markov decision processes. In Jyrki Kivinen and Robert H. Sloan, editors, *Computational Learning Theory, 15th Annual Conference on Computational Learning Theory, COLT 2002, Sydney, Australia, July 8-10, 2002, Proceedings*, volume 2375 of *Lecture Notes in Computer Science*, pages 255–270. Springer, 2002. doi:10.1007/3-540-45435-7_18.

28 Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *J. Mach. Learn. Res.*, 7:1079–1105, 2006. URL: http://jmlr.org/papers/v7/evendar06a.html.

29 Uriel Feige. Approximating maximum clique by removing subgraphs. *SIAM Journal on Discrete Mathematics*, 18(2):219–225, 2004.

30 Willem Feijen and Guido Schäfer. Using machine learning predictions to speed-up dijkstra's shortest path algorithm. *CoRR*, pages 1–28, 2021.

31 Fedor V. Fomin, Fabrizio Grandoni, and Dieter Kratsch. Measure and conquer: a simple $o(2^{0.288n})$ independent set algorithm. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2006, Miami, Florida, USA, January 22-26, 2006*, pages 18–25. ACM Press, 2006. URL: http://dl.acm.org/citation.cfm?id=1109557.1109560.

32 Waldo Gálvez, Arindam Khan, Mathieu Mari, Tobias Mömke, Madhusudhan Reddy Pittu, and Andreas Wiese. A 3-approximation algorithm for maximum independent set of rectangles. In Joseph (Seffi) Naor and Niv Buchbinder, editors, *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022, Virtual Conference / Alexandria, VA, USA, January 9 - 12, 2022*, pages 894–905. SIAM, 2022. doi:10.1137/1.9781611977073.38.

33 Suprovat Ghoshal, Konstantin Makarychev, and Yury Makarychev. Constraint satisfaction problems with advice. *arXiv preprint*, 2024. arXiv:2403.02212.

34 Francesco Gullo, Domenico Mandaglio, and Andrea Tagarelli. A combinatorial multi-armed bandit approach to correlation clustering. *Data Min. Knowl. Discov.*, 37(4):1630–1691, 2023. doi:10.1007/S10618-023-00937-5.

35 Shubham Gupta, Peter W. J. Staar, and Christian de Sainte Marie. Clustering items from adaptively collected inconsistent feedback. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li, editors, *International Conference on Artificial Intelligence and Statistics, 2-4 May 2024, Palau de Congressos, Valencia, Spain*, volume 238 of *Proceedings of Machine Learning Research*, pages 604–612. PMLR, 2024. URL: https://proceedings.mlr.press/v238/gupta24a.html.

36 Johan Håstad. Clique is hard to approximate within $n^{(1-\epsilon)}$. In *37th Annual Symposium on Foundations of Computer Science, FOCS '96, Burlington, Vermont, USA, 14-16 October, 1996*, pages 627–636. IEEE Computer Society, 1996. doi:10.1109/SFCS.1996.548522.

37 Monika Henzinger, Andrea Lincoln, Barna Saha, Martin P Seybold, and Christopher Ye. On the complexity of algorithms with predictions for dynamic graph problems. *arXiv preprint*, 2023. arXiv:2307.16771.

38 Chen-Yu Hsu, Piotr Indyk, Dina Katabi, and Ali Vakilian. Learning-based frequency estimation algorithms. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL: https://openreview.net/forum?id=r1lohoCqY7.

39 Bingbing Hu, Evangelos Kosinas, and Adam Polak. Connectivity oracles for predictable vertex failures. *arXiv preprint*, 2023. arXiv:2312.08489.

40 Shivaram Kalyanakrishnan and Peter Stone. Efficient selection of multiple bandit arms: Theory and practice. In Johannes Fürnkranz and Thorsten Joachims, editors, *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 511–518. Omnipress, 2010. URL: https://icml.cc/Conferences/2010/papers/410.pdf.

41 Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. PAC subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012. URL: http://icml.cc/2012/papers/359.pdf.

42 Richard M Karp. *Reducibility among combinatorial problems.* Springer, 2010.

43 Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for max-cut and other 2-variable csps? *SIAM Journal on Computing*, 37(1):319–357, 2007.

44 Yuko Kuroki, Atsushi Miyauchi, Francesco Bonchi, and Wei Chen. Query-efficient correlation clustering with noisy oracle. *CoRR*, abs/2402.01400, 2024. doi:10.48550/arXiv.2402.01400.

45 Silvio Lattanzi, Ola Svensson, and Sergei Vassilvitskii. Speeding up bellman ford via minimum violation permutations. In *International Conference on Machine Learning*, pages 18584–18598. PMLR, 2023.

46 Thomas Lavastida, Benjamin Moseley, R. Ravi, and Chenyang Xu. Learnable and instance-robust predictions for online matching, flows and load balancing. In Petra Mutzel, Rasmus Pagh, and Grzegorz Herman, editors, *29th Annual European Symposium on Algorithms, ESA 2021, September 6-8, 2021, Lisbon, Portugal (Virtual Conference)*, volume 204 of *LIPIcs*, pages 59:1–59:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.ESA.2021.59.

47 Avner Magen and Mohammad Moharrami. Robust algorithms for MAX INDEPENDENT SET on minor-free graphs based on the sherali-adams hierarchy. 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 258–271. Springer, 2009. doi:10.1007/978-3-642-03685-9_20.

48 Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. *Communications of the ACM*, 65(7):33–35, 2022.

49 Thomas Pontoizeau, Florian Sikora, Florian Yger, and Tristan Cazenave. Neural maximum independent set. In *Machine Learning and Principles and Practice of Knowledge Discovery in Databases - International Workshops of ECML PKDD 2021, Virtual Event, September 13-17, 2021, Proceedings, Part I*, volume 1524 of *Communications in Computer and Information Science*, pages 223–237. Springer, 2021. doi:10.1007/978-3-030-93736-2_18.

50 Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ML predictions. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9684–9693, 2018. URL: https://proceedings.neurips.cc/paper/2018/hash/73a427badebe0e32caa2e1fc7530b7f3-Abstract.html.

51 John M Robson. Finding a maximum independent set in time o (2n/4). Technical report, Technical Report 1251-01, LaBRI, Université Bordeaux I, 2001.

52 Tim Roughgarden. *Beyond the worst-case analysis of algorithms*. Cambridge University Press, 2021.

53 Karim Abdel Sadek and Marek Elias. Algorithms for caching and mts with reduced number of predictions. *arXiv preprint*, 2024. arXiv:2404.06280.

54 Jeanette P. Schmidt, Alan Siegel, and Aravind Srinivasan. Chernoff-hoeffding bounds for applications with limited independence. *SIAM J. Discret. Math.*, 8(2):223–250, 1995.

55 Max Simchowitz, Kevin G. Jamieson, and Benjamin Recht. The simulator: Understanding adaptive sampling in the moderate-confidence regime. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017*, volume 65 of *Proceedings of Machine Learning Research*, pages 1794–1834. PMLR, 2017. URL: http://proceedings.mlr.press/v65/simchowitz17a.html.

56 Clifford Stein and Hao-Ting Wei. Learning-augmented online packet scheduling with deadlines. *CoRR*, abs/2305.07164, 2023. doi:10.48550/arXiv.2305.07164.

57 Jose L. Walteros and Austin Buchanan. Why is maximum clique often easy in practice? *Oper. Res.*, 68(6):1866–1895, 2020. doi:10.1287/OPRE.2019.1970.

58 Jinghui Xia and Zengfeng Huang. Optimal clustering with noisy queries via multi-armed bandit. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 24315–24331. PMLR, 2022. URL: https://proceedings.mlr.press/v162/xia22a.html.

59 Mingyu Xiao and Hiroshi Nagamochi. Exact algorithms for maximum independent set. In Leizhen Cai, Siu-Wing Cheng, and Tak Wah Lam, editors, *Algorithms and Computation - 24th International Symposium, ISAAC 2013, Hong Kong, China, December 16-18, 2013, Proceedings*, volume 8283 of *Lecture Notes in Computer Science*, pages 328–338. Springer, 2013. doi:10.1007/978-3-642-45030-3_31.

60 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.

A Technical Preliminaries

We use the following standard forms of Chernoff bound.

▶ **Proposition 11** (Chernoff-Hoeffding bound). *Let $X_1, \ldots, X_m$ be m independent random variables with support in $[0,1]$. Define $X := \sum_{i=1}^{m} X_i$. Then, for every $t > 0$,*

$$\Pr\left(X - \mathbb{E}[X] \geqslant t\right) \leqslant \exp\left(-\frac{2t^2}{m}\right)$$
$$\Pr\left(X - \mathbb{E}[X] \leqslant -t\right) \leqslant \exp\left(-\frac{2t^2}{m}\right).$$

▶ **Proposition 12** (Chernoff bound; c.f. [25]). *Suppose $X_1, \ldots, X_m$ are m independent random variables with range $[0,1]$ each. Let $X := \sum_{i=1}^{m} X_i$ and $\mu_L \leqslant \mathbb{E}[X] \leqslant \mu_H$. Then, for any $\delta \in [0,1]$,*

$$\Pr\left(X > (1+\delta)\cdot\mu_H\right) \leqslant \exp\left(-\frac{\delta^2\cdot\mu_H}{3+\delta}\right) \quad \text{and} \quad \Pr\left(X < (1-\delta)\cdot\mu_L\right) \leqslant \exp\left(-\frac{\delta^2\cdot\mu_L}{2+\delta}\right).$$

We also consider limited independence hash functions. Roughly speaking, a k-wise independent hash function behaves like a totally random function when considering at most k elements. Formally, a family of hash functions $H = \{h : [n] \to [m]\}$ is k-wise independent if for any $x_1, x_2, \ldots, x_k \in [n]$ and $y_1, y_2, \ldots, y_k \in [m]$ the following holds:

$$\Pr_{h \in_R H}\left(h(x_1) = y_1 \wedge h(x_2) = y_2 \wedge \ldots \wedge h(x_k) = y_k\right) = m^{-k}.$$

We shall use the following concentration result on an extension of Chernoff-Hoeffding bounds for limited independence hash function.

▶ **Proposition 13** ([54]). *Suppose h is a k-wise independent hash function and $X_1, \ldots, X_m$ are m random variables in $\{0,1\}$ where $X_i = 1$ iff $h(i) = 1$. Let $X := \sum_{i=1}^{m} X_i$. Then, for any $\delta > 0$,*

$$\Pr\left(|X - \mathbb{E}[X]| \geqslant \delta\cdot\mathbb{E}[X]\right) \leqslant \exp\left(-\min\left\{\frac{k}{2}, \frac{\delta^2}{4+2\delta}\cdot\mathbb{E}[X]\right\}\right).$$