Proceedings of Machine Learning Research vol 336:1–37, 2026 39th Annual Conference on Learning Theory

Active Learning on Adversarially Corrupted Graphs

Marco Bressan MARCO.BRESSAN@UNIMI.IT
Università degli Studi di Milano, Italy

Nicolò Cesa-Bianchi NICOLO.CESA-BIANCHI@UNIMI.IT
Università degli Studi di Milano, Italy

Tommaso d'Orsi TOMMASO.DORSI@UNIBOCCONI.IT
Bocconi University, Italy

Emmanuel Esposito EMMANUEL@EMMANUELESPOSITO.IT
Università degli Studi di Milano, Italy

Silvio Lattanzi SILVIOL@GOOGLE.COM
Google Research

Editors: Steve Hanneke and Tor Lattimore

Abstract

Motivated by real-world scenarios where malicious entities tamper with existing networks, we define a model where an adversary seeks to hide a set of *corrupted vertices* inside a graph G^*. To this end, the adversary can add edges between the corrupted vertices, as well as edges between the corrupted vertices and G^*, and its power is then measured by the size of the *neighborhood* of the corrupted vertices in G^*. Our goal is to design an active learning algorithm that efficiently finds the subset of corrupted vertices using a small number of label queries. We devise an efficient algorithm that approximately recovers the corrupted vertices with a query complexity that depends polynomially on both the power of the adversary and the *vertex expansion* of G^*, a fundamental measure of graph connectivity. At the heart of this result is a polynomial-time algorithm, obtained by carefully adapting sum-of-squares algorithms for approximating minimum expansion, that finds a set with small vertex expansion subject to cardinality constraints. To the best of our knowledge, this is the first time that the vertex expansion is shown to play a key role in determining the query complexity of active learning algorithms robust to structural adversarial attacks.

Keywords: active learning, weak recovery, adversarial robustness

Authors are listed in alphabetical order.

1. Introduction

Graph-based machine learning is a powerful paradigm for analyzing relational data across diverse domains such as social networks, bioinformatics, web analysis, and recommendation systems. A key application is node classification, aiming to infer node labels or attributes from the graph structure and any available node or edge features. While standard approaches typically depend on the integrity of the observed graph structure, this presumption is often undermined by adversarial interventions. Malicious actors can, in fact, manipulate the graph by creating deceptive nodes or engineering spurious connections. This threat is especially pronounced in real-world applications like anti-abuse systems, where adversaries strategically deploy fake entities and artificial links to propagate spam, misinformation, or engage in fraudulent activities (Yu et al., 2008a,b; Danezis and Mittal, 2009; Tran et al., 2011; Alvisi et al., 2013).

This paper addresses the problem of active learning on graphs in the presence of such structural adversarial manipulations. Active learning in graphs aims to minimize the cost of data labeling by exploiting the graph structure to select which nodes to query for their true labels (Guillory and Bilmes, 2009). Most previous work focuses on adversaries who choose the node labeling, rather than adversaries who modify the graph structure itself. In this paper, we focus instead on adversarial structural changes and on algorithms with formal guarantees for general graphs.

Towards this end, we introduce an adversary that captures many practical safety scenarios. Given an initial graph G^*, the adversary can (1) introduce an arbitrary graph $\widetilde{G}$ with up to $|V(G^*)|$ vertices, (2) arbitrarily connect at most b vertices of G^* with arbitrarily many vertices in $\widetilde{G}$, and (3) add arbitrarily many edges in G^*. The algorithm is then given access to the resulting graph G, and has to distinguish $V(G^*)$ from $V(\widetilde{G})$.

The intuition behind this adversary is that a malicious actor can easily create new corrupted vertices and connect them, but has to put a significant effort in order to corrupt an existing vertex of G^* and connect it to $\widetilde{G}$. This captures real-world experimental observations where, for example, in social networks attackers create arbitrary structures between malicious users and tend to connect only to a smaller number of users that are more likely to accept friendship (Yang et al., 2014) or in link farms on the Web where an arbitrary structure is generated between malicious pages but only a few pages (often corrupted or acquired) have links to bad ones (Zhang et al., 2004; Becchetti et al., 2008).

In this model, the main goal of the adversary is to maximize the number of corrupted vertices (the fake profiles) that remain unidentified by the learning algorithm. The algorithm, in turn, seeks to identify a high fraction of corrupted nodes by querying the labels of a small number of nodes.[1]

Our result. Our main contribution is an efficient active learning algorithm tailored to this adversarial setting. We prove that the query complexity required by our algorithm to achieve a high accuracy is fundamentally linked to the vertex expansion of the input graph and the adversary's budget b. Vertex expansion is a measure of connectivity of the graph, quantifying the minimum ratio of the

1. A similar model has also been studied in the security literature (Yu et al., 2008a,b; Danezis and Mittal, 2009; Tran et al., 2011; Alvisi et al., 2013) where the budget of the adversary was linked to the number of edges that the adversary would add to the corrupted nodes. The setting analyzed in this paper is more realistic and strictly harder as we allow the adversary to add an unbounded number of edges to the corrupted nodes.

frontier of a set to its size (appropriately normalized). Intuitively, graphs with low vertex expansion allow adversaries to "hide" new nodes more easily within sparsely connected components. Our analysis formally connects this structural property to the number of queries needed for detection. This explicit link between vertex expansion and the query complexity of active learning under structural attacks is, to the best of our knowledge, novel, and provides theoretical grounding for designing algorithms robust to this class of adversarial behavior. We believe this connection may inspire further research into leveraging graph expansion properties for robust learning on graphs.

Problem formulation. We can now define our model and our problem more formally. If $G = (V, E)$ is a graph and $S \subseteq V$, the *frontier* $\partial_G(S)$ of S in G is the set of neighbors of S in $V \setminus S$. We consider graphs G constructed in the following way.

Definition 1 (Adversarial model) *Let $G^* = (V^*, E^*)$ be an arbitrary graph, and let $b \in \mathbb{N}$. An* adversary with budget b *creates a graph G by manipulating G^* in three steps:*

(i) The adversary creates an arbitrary graph on a set I of fresh vertices, with $|I| \leqslant |V^*| =: n$. We refer to I as the set of corrupted *or* malicious *nodes.*

(ii) The adversary adds an arbitrary set of edges between I and at most b vertices of V^*, so that $|\partial_G(I)| \leqslant b$. Note that, save for this constraint, the adversary can place the edges arbitrarily.

(iii) The adversary adds an arbitrary number of edges within *G^*.*

It shall be noted that (iii) is not intended to model the adversary's behavior; we include it simply because our algorithm is unaffected by this kind of perturbations. Given a graph G constructed as in Definition 1, we seek to identify the corrupted vertices I with good accuracy. Note that this model grants significant power to the adversary. For instance, already in the simplest settings in which G is a stochastic block with communities V^* and I (so that the starting graph G^* is Erdős-Rényi, and the edits introduced in (i), (ii) are random), the monotone perturbations introduced in (iii) are known to make the task information-theoretically harder for an important range of parameters (Moitra et al., 2016). In general, the original graph G^* and the corrupted subgraph $G[I]$ are both arbitrary, and the connections between them adversarial. Therefore, the structural properties of the resulting graph G we may rely on are significantly weaker than other common assumptions studied in the literature, such as requiring I to be dense (e.g., a clique) or expanding.[2] Similarly, bounding the number of *neighbors* of I in G^*, rather than the number of *edges* between I and G^*, gives more power to the adversary, since there are at least as many edges as the number of neighbors.

Clearly, without further assumptions, G^* and $G[I]$ are indistinguishable and may even be isomorphic, thus in general there is little hope to recover I even approximately. In several concrete scenarios, however, one can *learn* whether a given vertex v is malicious or not, for instance by carefully observing the behavior of a profile in a social network. We model this assumption by equipping the algorithm with a *label oracle*. A label oracle for I in G, denoted by $\mathcal{O}_{G,I}$, returns $\mathbb{I}\{v \in I\}$ on input $v \in V$. Clearly, label queries should be considered as expensive. The algorithm

2. See the discussion in Section 2 for a more in-depth comparison with the related literature.

should then recover I efficiently, with good accuracy, and by making few queries to $\mathcal{O}_{G,I}$. This leads to the following definition.

Definition 2 (Weak recovery with oracle) *Let G^* be a graph, and let G be generated from G^* by an adversary with budget b. For $\gamma, \delta \in (0,1]$ and $q \geqslant 0$, an algorithm achieves (γ, δ, q, b)-*weak recovery *of I in G if, given solely G, γ, δ, and a label oracle $\mathcal{O}_{G,I}$, the algorithm performs at most q calls to $\mathcal{O}_{G,I}$ and with probability at least $1-\delta$ returns a set $\hat{I} \subseteq V(G)$ satisfying $|\hat{I} \triangle I| \leqslant \gamma \cdot |V(G)|$.*

Importantly, note that $|I|$ and b are unknown to the algorithm. It is immediate to see that, in general, one cannot achieve (γ, δ, q, b)-weak recovery with a small query budget q, even given $b = 0$; for instance, if G is edgeless, then one needs $\Omega(n)$ queries to learn I for any constant γ, δ. Therefore (γ, δ, q, b)-weak recovery must exploit some additional property of G. For $0 < m < n$, the *m-large frontier of G* is:

$$\partial_m(G) \coloneqq \min_{\substack{S \subseteq V \\ m \leqslant |S| \leqslant n-m}} |\partial_G(S)| \,. \tag{1.1}$$

Our main result is:

Theorem 3 (Weak recovery) *There exists a randomized polynomial-time algorithm[3] that achieves (γ, δ, q, b)-weak recovery whenever:*

(i) $q \geqslant \Omega\left(\frac{\operatorname{poly}\log \frac{1}{\gamma}}{\gamma} \left(\log \frac{1}{\delta} + b\sqrt{\log n} \right) \right)$;

(ii) $b \leqslant O\left(\frac{\gamma^6}{\log^3(1/\gamma)\sqrt{\log n}} \cdot \partial_{\eta n}(G^*) \right), \quad \eta = O\left(\frac{\gamma}{\log \frac{1}{\gamma}} \right).$

To appreciate the result, consider the case $|I| = \widetilde{\Omega}(n)$. By letting $\gamma = \varepsilon \cdot \frac{|I|}{n}$, and ignoring polylogarithmic factors in the parameters, Theorem 3 says that one can recover I with a constant *multiplicative* accuracy of ε by making roughly b/ε queries, as long as b is significantly smaller than the smallest frontier of subsets of G^* of size comparable to I. Thus, the query budget of the algorithm is linear in the budget of the adversary. Theorem 3 continues to hold even for $|I| = O(n^a)$, but, as a decreases, the second constraint will degenerate to $b = 0$, requiring the corrupted graph $G[I]$ to be disconnected from G^*. Alternatively, one can tolerate a nontrivial budget b, accepting in exchange a larger misclassification rate. Similarly, one can consider $\gamma = o(1)$, and Theorem 3 gives nontrivial bounds (i.e., admits $b > 0$) as long as $\gamma = \widetilde{\Omega}\left(n^{-1/6}\right)$. In summary, Theorem 3 gives a tradeoff between the accuracy γ, the query budget q, and the adversary's budget b. More generally, the theorem shows that weak recovery remains possible even when γ is small, at the cost of proportionally increasing the number of queries and considering adversaries with reduced budget.

On a more technical level, note that the second constraint of Theorem 3 only imposes a bound on the vertex expansion of subsets of $V(G^*)$ with size at least $\eta n = O\left(\frac{\gamma n}{\log(1/\gamma)} \right)$, that is, a $O(1/\log(1/\gamma))$ fraction of the sought error γn. On smaller subsets, the theorem imposes no restriction whatsoever. That is, *no* structural assumption is required on those sets. This means that the

3. Here and unless otherwise specified, by randomized polynomial-time algorithm we mean a Monte Carlo algorithm.

ground-truth graph G^* may be far from being a (small-set) vertex expander; in fact, it may even contain exponentially many vertex cuts that are significantly smaller than $|\partial_G(I)| \leqslant b$.

Auxiliary result: unbalanced vertex expansion. A by-product of our analysis is a randomized polynomial-time algorithm that finds a set with small vertex expansion and size within $[m, n-m]$. We obtain this result by carefully modifying the algorithm of Feige et al. (2005). We believe this statement may be of independent interest. The *vertex expansion* of S in G is:

$$\phi_G(S) \coloneqq \frac{|\partial_G(S)|}{|S| \cdot |V \setminus S|} .$$

For $0 < m(n) < n = |V(G)|$, the *m-large vertex expansion* of G is:

$$\phi_m(G) \coloneqq \min_{\substack{\emptyset \neq S \subset V(G) \\ m \leqslant |S| \leqslant n-m}} \phi_G(S) .$$

Note that $\phi_m(G)$ is a generalization of the canonical notion of vertex expansion, which is in fact $\phi_1(G)$. For $m \neq 1$ we obtain the following theorem.

Theorem 4 ((m/n)-balanced vertex expansion) *There exists a randomized polynomial-time algorithm that, given an n-vertex graph G and $0 < m \leqslant n/2$, returns $S \subset V$ satisfying*

(i) $\phi_G(S) \leqslant \phi_m(G) \cdot O\big(\sqrt{\log n} + \frac{n}{m}\big)$,

(ii) $\min\{|S|, |V \setminus S|\} = \Omega(m)$.

Observe that, for $n/m = O\big(\sqrt{\log n}\big)$, the approximation factor of Theorem 4 remains $\sqrt{\log n}$. Interestingly, for the related but possibly more challenging problem of small-set vertex expansion, known efficient algorithms (Louis and Makarychev, 2016) only achieve an approximation of the order $\frac{n}{m} \cdot \log \frac{n}{m} \cdot \log\log \frac{n}{m} \cdot \sqrt{\log n}$.

2. Related work

The literature on active learning, spam, and abuse is vast. Here we focus on summarizing the contributions closest to our work.

Sybil attacks. A very related area of research is the design of algorithms to protect social networks under Sybil attack (Yu et al., 2008a,b; Danezis and Mittal, 2009; Tran et al., 2011; Alvisi et al., 2013). Similarly to our setting, the attacker is able to add nodes to the networks. However, there are two main differences with our approach. First, the budget of the adversary is based on the number of edges added by the adversary between the malicious nodes and the good nodes, and not on the size of the boundary like in our setting. Second, the good side of the graph is often assumed to be an expander—a property that is not often true in practice (Leskovec et al., 2008; Mohaisen et al., 2010). In comparison, our paper analyzes a harder and more realistic setting.

Foundations of active learning. The seminal paper by Balcan et al. (2006) introduced the first active learning algorithm, A^2, with sample complexity bounds showing that active learning can achieve exponential label savings over passive learning even in the non-realizable case. The thread of work started by that work has later evolved into a complex theory of disagreement-based active learning, nicely summarized in the monograph by Hanneke (2014). However, it seems not possible to achieve our query bounds via general-purpose active learning algorithms, even ignoring computation. To see why, given G^* define the concept class $\mathcal{C} = \{S \subseteq V : |\partial_{G^*}(S)| \leqslant b\}$; note that the task of (weakly) recovering I with oracle queries can be cast as an active learning problem over $\mathcal{C}$. Now suppose G^* is a 3-regular expander, let $\gamma, \eta = \Omega(1)$, and suppose $\delta_G(I) = \frac{b}{2}$ for $b = \Omega(\partial_{\eta n}(G^*)/\sqrt{\log n}) = \Omega(n)$. Note that the assumptions of Theorem 3 are satisfied. However, for almost all subsets $U \subseteq V^*$ with $|U| \leqslant b/6$, the set $S = I \cup U$ has boundary $\delta_G(S) \leqslant b$, and the ηn-large frontier of $G^* \setminus S$ is still $\Omega(\eta n) = \Omega(n)$. As there are $\binom{n^*}{b/6} = 2^{\Omega(n)}$ such subsets U, we get $|\mathcal{C}| = 2^{\Omega(n)}$. Thus, the bound on the number of active learning queries is no better than the trivial $O(n)$.

Active learning on graphs. Previous theoretical work on active learning on graphs has mainly explored query strategies for node classification based on minimizing prediction mistakes under label smoothness assumptions, often relating performance to the graph's cut size or related measures—like the $\Psi(L)$ function introduced by Guillory and Bilmes (2009)—which quantify the difficulty of separating unlabeled nodes from the labeled set L. The works by Cesa Bianchi et al. (2010); Cohen-Addad et al. (2025) use $\Psi(L)$ to prove results related to the optimal placement of queries on trees and general graphs. The performance of these algorithms scales linearly in the size of the cut between infected and non-infected nodes which, in our setting, can be superlinear in the adversary's budget. The S^2 algorithm (Dasarathy et al., 2015) uses binary search to locate the nodes to query—see also Afshani et al. (2007) for earlier results along these lines. Similarly to our analysis, the performance of S^2 is affected by the relative size of the infected subset. However, the S^2 query budget scales linearly with the size $|\partial C|$ of the boundary of the cut set C, which, again, can be superlinear in the adversary's budget (in our asymmetric model, the adversary only pays the non-infected nodes in the boundary of the cut set). More specifically, if the infected nodes have m connected components each of size $\Omega(n)$, and these connected components are well clustered, then the query budget for exact recovery with probability $1 - \delta$ is of order $m \log n + |\partial C| + \log \frac{1}{\delta}$. The work by Thiessen and Gärtner (2021) assumes that the sets of infected and non-infected nodes are geodesically convex in G. Under this assumption, the query budget for exact recovery is $h + \log d + 2t$, where h is the hull number of G (smallest number of vertices whose convex closure returns V), d is the diameter, and t is the treewidth. It is unclear how the convexity assumption can be related to the adversary's budget in our setting. Active learning under other notions of convexity for graphs have been also investigated in Bressan et al. (2024, 2025). Wu and Yuan (2024) introduce a query strategy based on spectral sparsification techniques achieving near-optimal query complexity while remaining robust to noisy node labels. Gu and Han (2012) base their query selection strategy on a data-dependent generalization bound derived via transductive Rademacher complexity.

Data poisoning on graphs. In the area of adversarial machine learning, data poisoning in graph-based learning is an established line of work—see the survey by Chen et al. (2020) (also Jin et al. (2021) for applications to graph neural networks). The work by Liu et al. (2019) focuses on graphs induced by datasets of feature vectors, where the adversary modifies the graph by flipping the labels or perturbing the features. A more general model of active learning under poisoning attacks is studied in Balcan et al. (2022), where they consider pool-based active learning in a traditional statistical learning setting. Yang et al. (2024) prove robustness of classification in a train/test learning model against a bounded number of edge additions or deletions. Their work is for graph classification, but can also be applied to node classification. This line of work is further explored in Li et al. (2025). These results are based on graph partitioning techniques combined with a majority vote. The robustness against graph perturbations is established by comparing the number of perturbations with the unbalance in the vote.

Graph partitioning. A substantial body of work has considered graph partitioning questions in semi-random models (Makarychev et al., 2012; Buhai et al., 2023; Cohen-Addad et al., 2024; Błasiok et al., 2024). For community detection, a celebrated line of research considered models in which the input graph is first sampled from a known stochastic block model, then an adversarially corrupted copy is received in input by the algorithm. The corruptions studied include monotone perturbations (Moitra et al., 2016; Liu and Moitra, 2022), edge perturbations (Banks et al., 2021; Ding et al., 2022; Mohanty et al., 2024) and node perturbations (Ding et al., 2023). As Definition 1 is significantly more general and captures all these models, the algorithms introduced in these prior works cannot recover the original graph in Definition 1. Indeed, our algorithm necessarily relies on access to a label oracle to achieve weak recovery.

From an algorithmic perspective, related to our results are the the state-of-the-art algorithms for the vertex expansion problem and its variants (Feige et al., 2005; Trevisan, 2009; Raghavendra et al., 2010; Louis et al., 2013; Louis and Makarychev, 2016; Ghoshal and Louis, 2024; Kwok et al., 2022). We discuss these results more in detail in Section 3.

Organization. The rest of the paper is organized as follows. In Section 3 we outline the main ideas behind Theorem 3. The appendix contains deferred proofs and the necessary background. In particular, Appendix A contains the proof of Theorem 3 and Appendix B contains the proof of Theorem 4. Finally, Appendix C contains the notation and technical results about sum-of-squares framework, which is the main algorithmic tool used to derive Theorem 4.

Notation. We write $G^* = (V^*, E^*)$ and $G = (V, E)$. When the context is clear, we use n to denote the number of vertices in the graph at hand. We use the notation $\widetilde{\Omega}(\cdot), \widetilde{O}(\cdot)$ to hide polylogarithmic factors in n plus the arguments of the notation. For a graph G and a set of vertices $S \subseteq V(G)$, we let $G[S]$ be the induced subgraph of G on S. For two sets $S, S' \subseteq V(G)$, we denote by $S \triangle S' = (S \setminus S') \cup (S' \setminus S)$ their symmetric difference. When G is clear from the context we may omit it from the subscripts, for instance by writing ∂ in place of ∂_G.

3. Techniques and main results

We present the main ideas behind Theorem 3. Let again $G^* = (V^*, E^*)$ be any graph, and let again $G = (V, E)$ be created from G^* according to our adversarial model. Let $m = |I|$ and $n = |V|$. Note that, by querying the labels of $q = O\left(\frac{1}{\gamma} \log \frac{1}{\delta}\right)$ uniform random vertices of V, we can detect whether $m < \gamma n$ with probability $1 - \delta$ and return the empty set, which satisfies Theorem 3 regardless of b. Let us then assume $m \geqslant \gamma n$. For simplicity of exposition we assume $\gamma \geqslant \Omega(1)$ small enough, $b = |\partial_G(I)|$, and that m is known (our algorithm needs only a constant-factor estimate, which takes again $O\left(\frac{1}{\gamma} \log \frac{1}{\delta}\right)$ queries). Finally, again for simplicity of exposition, many of the inequalities and constraints are presented below in a simplified form (for example, using γn in place of $\frac{c\gamma n}{\log(1/\gamma)}$ for some universal $c > 0$). However, along the discussion we also provide the full formal statements, with the correct parameterization.

3.1. A vertex expansion problem

As a starting point, we devise sufficient conditions on G so that we can recover a subset S that is heavily correlated with the subset $I \subseteq V$ of corrupted vertices. For the moment let us assume that G^* and $G[I]$ are arbitrary; the only properties we have are therefore the bound $|\partial_G(I)| \leqslant b$ on the boundary of I and the bound $m = |I| \geqslant \gamma n$ on the size of I. Now, suppose we can find efficiently *some* subset $S \subseteq V$ that has these properties (even approximately):

(i) $|\partial_G(S)| = O(b)$,

(ii) $|S| = \Theta(|I|)$.

Then, we would like these two properties to imply that S is a good proxy for I—that is, that S is heavily correlated with I, in the sense that:

$$\frac{|I \cap S|}{|S|} \geqslant 1 - O(\gamma)\,. \tag{3.1}$$

This means that almost all vertices of S are in I, and moreover, together with *(ii)* above, S contains a constant fraction of points of I. We could then add S to our approximate set $\widetilde{I}$ of corrupted vertices, delete S from G, and repeat.

A natural way to rule out the existence of subsets S satisfying *(i)* and *(ii)* but violating Eq. (3.1) is to require G^* to satisfy a *boundary* condition. Suppose indeed that every $S \subseteq V^*$ with $\min\{|S|, |V^* \setminus S|\} \geqslant \gamma n$ has in G^* an outer boundary much larger than the outer boundary of I in G. More precisely, suppose:

$$|\partial_{G^*}(S)| > \frac{|\partial_G(I)|}{\gamma^2} \qquad \forall S \subseteq V^*, \min\{|S|, |V^* \setminus S|\} \geqslant \gamma n\,. \tag{3.2}$$

Then it is possible to show that *every* set $S \subseteq V$ satisfying *(i)* and *(ii)* satisfies:

$$\min\left\{\frac{|I \cap S|}{|S|}, \frac{|I \cap \bar{S}|}{|\bar{S}|}\right\} \geqslant 1 - O(\gamma)\,. \tag{3.3}$$

where $\bar{S} = V \setminus S$. Thus the idea is that, if every bipartition $(S, V^* \setminus S)$ of G^* which is not too unbalanced has boundaries significantly larger than the one of I, as given by Eq. (3.2), then every bipartition $(S, V \setminus S)$ of G satisfying *(i)* and *(ii)* above is strongly correlated with (I, V^*) in the sense of Eq. (3.3). We now use Eq. (3.2) to devise a constraint on the *vertex expansion* of G^*. In particular, suppose:

$$\phi_G(I) < \gamma^3 \cdot \phi_{\gamma n}(G^*)\,. \tag{3.4}$$

Then, using the definition of ϕ, we obtain:

$$\begin{aligned}
\min_{\substack{S \subset V^* \\ \min\{|S|, |V^* \setminus S|\} \geqslant \gamma n}} |\partial_{G^*}(S)| &\geqslant \phi_{\gamma n}(G^*) \cdot \gamma n(n^* - \gamma n) && (3.5)\\
&> \frac{1}{\gamma^3} \cdot \phi_G(I) \cdot \gamma n(n^* - \gamma n) && (3.6)\\
&= \frac{1}{\gamma^3} \cdot \frac{|\partial_G(I)|}{m(n-m)} \cdot \gamma n(n^* - \gamma n) && (3.7)\\
&\geqslant \frac{|\partial_G(I)|}{\gamma^2} && (3.8)
\end{aligned}$$

where in the last inequality we used $\gamma n \leqslant m \leqslant \frac{n}{2} \leqslant n^*$ and γ small enough to ensure $\frac{n(n^* - \gamma n)}{m(n-m)} \geqslant 1$. What we obtained, however, is precisely Eq. (3.2). Thus Eq. (3.4) is a sufficient condition to ensure, again, that every set satisfying *(i)* and *(ii)* above satisfies Eq. (3.3) too. Let us then turn Eq. (3.4) into a formal definition.

Definition 5 (Poorly expanding set) *Let G be an n-vertex graph. Let $0 < m \leqslant n/2$, $0 < t < n$, and $0 < \varepsilon < 1$. We say that a set $U \subseteq V$ is (ε, t, m)-expanding in G if $m \leqslant |U| \leqslant \frac{n}{2}$ and*

$$\phi_G(U) < \varepsilon \cdot \phi_t(G[V \setminus U])\,.$$

In words, the definition says that U has vertex expansion significantly smaller than the one of $G[V \setminus U]$, when the latter is measured only over "balanced" subsets (i.e., with size at least t and at most $|V \setminus U| - t$). Applying this definition to the set I of corrupted vertices, we obtain that if I is $(\gamma^3, \gamma n, m)$-expanding in G then Eq. (3.4) holds. We now see how, starting from the assumption that I is $(\gamma^3, \gamma n, m)$-expanding in G, we can actually compute an $S \subseteq V$ that satisfies Eq. (3.3).

3.2. Computing approximations to poorly expanding sets

The above discussion suggests that we shall compute a set $S \subseteq V$ that satisfies *(i)* and *(ii)*, that is, S has small outer boundary and relatively large size compared to I. This problem is a variation of vertex expansion and thus, unfortunately, NP-hard in general. In particular, existing efficient algorithms for vertex expansion fall short of finding the desired solution in two ways. First, they return sets whose vertex expansion can be a factor $O(\sqrt{\log n})$ (Feige et al., 2005) or $O(\sqrt{\phi \log d})$ (Louis et al., 2013) larger than the minimum vertex expansion, where d is the maximum degree of the graph. Second, they offer no guarantees on the size of returned sets. The algorithm of Louis et al.

(2013) relies on a connection between vertex expansion and a spectral profile parameter, called λ_∞, studied in Raghavendra et al. (2010). As this connection breaks down upon restricting the cardinality of admissible sets as in Eq. (3.4), our approach more closely resembles that of Feige et al. (2005). Note that variations of algorithms for small-set vertex expansion (Raghavendra et al., 2010; Louis and Makarychev, 2016; Ghoshal and Louis, 2024) could also be explored in principle. Ghoshal and Louis (2024) ensures the returned set is of size $\Theta(m)$ but requires time $O(n^{n/m})$. Louis and Makarychev (2016) does not guarantee a lower bound on the size of the returned set and ultimately has a dependency on the ratio $\frac{n}{m}$ that is $O(\sqrt{\log n})$ times worse compared to Theorem 4. Because of these drawbacks, we do not directly build on these works.

To circumvent these computational hardness barriers, we strengthen the constraint of Eq. (3.4) by increasing the gap between the vertex expansion of I and the vertex expansion of G^*, to obtain a constraint in the form:

$$\phi_G(I) < \frac{\gamma^3}{C\sqrt{\log n}} \cdot \phi_{\gamma n}(G^*) \tag{3.9}$$

where $C > 0$ is a sufficiently large constant. Note that on could view the difference between Eq. (3.4) and Eq. (3.9) as an information-computation gap for the weak recovery of I. Indeed, under hardness assumptions such as the Small Set Expansion Hypothesis (SSEH), it seems plausible that our variant of vertex expansion is as hard as the original problem. In terms of Theorem 5, we require I to be $\left(\frac{\gamma^3}{C\sqrt{\log n}}, \gamma n, m\right)$-expanding in G. At this point, we are able to compute a set S with outer boundary size $|\partial_G(S)|$ close to $b = |\partial_G(I)|$. To this end, we describe a sum-of-squares program based on the semidefinite relaxation for vertex expansion of Feige et al. (2005).[4] The crucial adaptation we make is the introduction of a novel, delicate rounding algorithm which ensures that the output set S satisfies $\min\{|S|, |V \setminus S|\} \geqslant \Omega(|I|)$; this is where the gap of Eq. (3.9) is exploited. Overall, we obtain a set S which satisfies both *(i)* and *(ii)* above, and thus, as argued, Eq. (3.3) as well. At this point we know that one among S and $\bar{S} = V \setminus S$ strongly overlaps with I in the sense of Eq. (3.1). To find this out, we simply query the oracle for the labels of $\frac{\log(1/p)}{\gamma}$ uniformly random points from S and from $V \setminus S$, and this correctly tells us which set to pick with probability $1 - p$. Formally, we prove the following result.

Lemma 6 (Simplified version of Lemma 10) *There exists a randomized polynomial-time algorithm with the following guarantees. Suppose $I \subseteq V$ is $\left(\frac{\eta^2}{\alpha}, \eta m, m\right)$-expanding in $G = (V, E)$, where $m = |I|$ and $\alpha = O\left(\sqrt{\log n} + \frac{n}{m}\right)$ is large enough and $\eta \in (0, 1)$ is small enough. Then, given G and $\hat{m} \in [m/2, m]$, access to a label oracle $\mathcal{O}_{G,I}$ for I, and $p \in (0, 1)$, the algorithm makes at most $\alpha \cdot |\partial_G(I)| + O\left(\log \frac{1}{p}\right)$ oracle queries and returns $S \subseteq V(G)$ that with probability $1 - p$ satisfies:*

(i) $|\partial_G(S)| = O(\alpha \cdot |\partial_G(I)|)$,

(ii) $|S| = \Theta(|I|)$,

4. More accurately, we use a degree-4 sum-of-squares relaxation and analyze its performance in a way similar to the SDP relaxation considered in Feige et al. (2005).

(iii) $|S \setminus I| = O(\eta \cdot |S|)$.

To employ Lemma 6, recall from above that we assume I is $\left(\frac{\gamma^3}{C\sqrt{\log n}}, \gamma n, m\right)$-expanding in G. It is not hard to check that I then satisfies the assumptions of the lemma with $\eta = \gamma \cdot \frac{n}{m}$. Hence, the lemma ensures we can compute a set S that satisfies *(i)-(iii)*. In particular, the outer boundary of S satisfies[5]

$$|\partial_G(S)| = O(\alpha \cdot |\partial_G(I)|) = O\left(\frac{\sqrt{\log n}}{\gamma} \cdot |\partial_G(I)|\right) \tag{3.10}$$

where we used $m \geqslant \gamma n$. Moreover, S satisfies

$$|S \setminus I| = O(\eta \cdot |S|) = O(\gamma n)\,, \tag{3.11}$$

as $\eta = \gamma \cdot \frac{n}{m}$, and since it must be $|S| = O(m)$, for otherwise we would not have $|S \setminus I| = O(\eta \cdot |S|)$. We use these results to analyze the other iterations of the algorithm, in the next section.

3.3. Refining the partition via recursion

The algorithm introduced in Lemma 6 in the previous section (with more precise details in Lemma 10 from Appendix A) allows us to find a set S of size $\Theta(|I|)$ strongly correlated with I in the sense of Eq. (3.1), that is, with $|S \setminus I| = O(\gamma n)$. This is not yet sufficient to get weak recovery, which requires the stronger condition $|S \triangle I| \leqslant \gamma n$; see Definition 2. One obvious way to proceed, however, is to remove S from G and apply again the algorithm of Lemma 6 to the resulting graph $G \setminus S$. The difficulty is that it is not immediately clear that $G \setminus S$ still satisfies the assumptions of Lemma 6. For example, since we are removing vertices from G^* (those in $S \setminus I$), the vertex expansion of small subsets of G^* may increase, failing the assumption of poor expansion of $I \setminus S$ in $G[V \setminus S]$. More in general, Lemma 6 needs several hypotheses (which is made clearer in Lemma 10), and it is not straightforward to show that, if we iterate several times the procedure of finding a set S and removing it from G, those hypotheses are still satisfied for values of the parameters that do not yield vacuous guarantees. We will now show that this is indeed the case; that is, if the initial set of corrupted vertices I is "poorly expanding" enough in G, then Lemma 6 can be applied iteratively several times, each time removing the newly found set S from the remaining graph G without an excessive degradation of the relevant parameters.

Then, suppose we have found S through the algorithm of Lemma 6. The first observation we make is that, since S is almost entirely contained in I, the subgraph $G^* \setminus S$ has roughly the same properties of G^*. More precisely, we show (see Lemma 13 in Appendix A.3) that, for all $t > 0$,

$$\partial_t(G^* \setminus S) \geqslant \partial_t(G^*) - |\partial_G(S)|\,. \tag{3.12}$$

Together with the guarantees of Lemma 6 on $|\partial_G(S)|$, we obtain:

$$\partial_t(G^* \setminus S) \geqslant \partial_t(G^*) - O\left(\frac{\sqrt{\log n}}{\gamma}\right) \cdot |\partial_G(I)|\,. \tag{3.13}$$

5. Note that the bound on $|\partial_G(S)|$ finally shows a multiplicative $O\left(\frac{1}{\gamma}\sqrt{\log n}\right)$ factor, while $\alpha = O\left(\sqrt{\log n} + 1/\gamma\right)$. While one may try to improve the guarantees we provide by being more careful, we opted for this slightly looser bound to simplify the overall presentation.

Thus, after deleting S, the outer boundaries of all relevant subsets of G^* remain roughly as large as before. Moreover, the outer boundary of $I \setminus S$ in $G \setminus S$ cannot be larger than the one of I in G. Now the same inequality can be propagated if we repeat the process multiple times. Let then $G_0 = G$ be the initial graph, and let $k = O\left(\log \frac{1}{\gamma}\right)$ large enough. For each $i = 0, 1, \dots, k-1$, let G_i be the remaining graph after i steps. At each step i, by using the algorithm from Lemma 6 on G_i, we compute a subset $S_i \subseteq V(G_i)$ that is heavily correlated with $I_i = I \cap V(G_i)$ in the sense above, and we then remove S_i from G_i, obtaining $G_{i+1} = G_i \setminus S_i$. Let G_k be the resulting graph after k steps, and let $G_k^* = G_k \setminus I$ be the subgraph of G_k consisting of non-corrupted vertices. By iterating Eq. (3.13), we get:

$$\partial_t(G_k^*) \geqslant \partial_t(G^*) - O\left(\frac{\log(1/\gamma) \cdot \sqrt{\log n}}{\gamma}\right) \cdot |\partial_G(I)| \,. \tag{3.14}$$

Thus, if

$$|\partial_G(I)| \leqslant \partial_t(G^*) \cdot \frac{c\gamma}{\log(1/\gamma) \cdot \sqrt{\log n}} \tag{3.15}$$

for a sufficiently small constant $c > 0$, then $\partial_t(G_k^*) = \Omega(\partial_t(G^*))$, and therefore G_k^* is essentially the same as G^* as far as the expansion guarantees are concerned. (To be precise, one must consider the vertex expansion $\phi_t(G_k^*)$, and this changes things by some additional γ factor due to the normalization by sizes as small as γn). Formally, we prove the following result:

Lemma 7 (Simplified version of Lemma 11) *Suppose G and I satisfy the hypotheses of Lemma 6 with parameters*

$$\varepsilon = O\left(\frac{\gamma^3}{\alpha}\right), \quad \eta = O\left(\gamma \frac{n}{m}\right), \quad \alpha = O\left(\frac{1}{\gamma} \cdot \left(\sqrt{\log n} + \frac{n}{m}\right)\right)$$

for some $\gamma > 0$ sufficiently small, where ε essentially replaces η^2/α in the poorly expanding assumption on I in G from Lemma 6. Moreover, let $k = O\left(\log \frac{1}{\gamma}\right)$, and suppose the first k applications of Lemma 6 are successful. Then G_k and I_k satisfy again the hypotheses of Lemma 6 with parameters

$$\varepsilon' = \frac{\varepsilon}{\eta} \cdot \frac{|V_k|}{|I_k|}, \quad \eta' = \eta \frac{|I_k|}{|V_k|}, \quad \alpha' = \alpha \,.$$

The proof of the more detailed version of this key lemma (Lemma 11) is in Appendix A.3.

In short, Lemma 7 says that the parameters in the hypotheses of Lemma 6 degrade gracefully as we find and remove the sets $S_1, S_2, \dots$ iteratively as previously described. Since after $k = O\left(\log \frac{1}{\gamma}\right)$ successful applications of the lemma we will have removed a fraction $(1-\gamma)$ of I, this yields the desired weak recovery guarantee. The only remaining tweak is that, using the claims above, we will obtain a total error of $\gamma n \cdot k = \gamma n \cdot O\left(\log \frac{1}{\gamma}\right)$, rather than just γn. To fix this, we decrease the parameter η in Lemma 7 by a further $\log \frac{1}{\gamma}$. This essentially concludes the description of our weak recovery algorithm. Technically speaking, we obtain an algorithm for finding *poorly expanding sets* up to error γn. Formally, we prove (note that, compared to Lemma 7, now $\alpha = O\left(\frac{1}{\gamma^2}\sqrt{\log n}\right)$ is absorbed in ε):

Theorem 8 (Simplified version of Theorem 9) *Suppose I is $(\varepsilon, \eta m, m)$-expanding in G, where $m = |I|$ and*

$$\varepsilon = O\left(\frac{\gamma^5}{\log^3(1/\gamma)\sqrt{\log n}}\right), \quad \eta = O\left(\frac{\gamma}{\log(1/\gamma)}\frac{n}{m}\right).$$

Then, given G, γ, a label oracle $\mathcal{O}_{G,I}$ for I, and $p = p(n) \in (0,1)$, in polynomial time and by making $\widetilde{O}\left(\frac{1}{\gamma}\right) \cdot O\left(\ln\frac{1}{p} + \sqrt{\log n} \cdot \left|\partial_G(I)\right|\right)$ oracle queries one can compute a set $\widetilde{I}$ such that $|\widetilde{I} \triangle I| = O(\gamma n)$ with probability $1 - p$.

The proof of the full result (Theorem 9) can be found in Appendix A.2.

It is not hard to see that Theorem 8 yields our main result, Theorem 3. The only difference is a factor of γ between the expression of the budget b and the parameter ε above; this is due to the fact that Theorem 8 talks about vertex expansion, and Theorem 3 about boundary size, and in the conversion between the two, one loses the ratio $\frac{m}{n} \geqslant \gamma$.

Acknowledgments

NCB and EE acknowledge the financial support from the EU Horizon CL4-2022-HUMAN-02 research and innovation action under grant agreement 101120237, project ELIAS (European Lighthouse of AI for Sustainability).

References

Peyman Afshani, Ehsan Chiniforooshan, Reza Dorrigiv, Arash Farzan, Mehdi Mirzazadeh, Narges Simjour, and Hamid Zarrabi-Zadeh. On the complexity of finding an unknown cut via vertex queries. In *Computing and Combinatorics: 13th Annual International Conference*, pages 459–469. Springer, 2007.

Lorenzo Alvisi, Allen Clement, Alessandro Epasto, Silvio Lattanzi, and Alessandro Panconesi. Sok: The evolution of sybil defense via social networks. In *2013 ieee symposium on security and privacy*, pages 382–396. IEEE, 2013.

Sanjeev Arora, Satish Rao, and Umesh Vazirani. Expander flows, geometric embeddings and graph partitioning. *Journal of the ACM (JACM)*, 56(2):1–37, 2009.

Maria-Florina Balcan, Alina Beygelzimer, and John Langford. Agnostic active learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 65–72, 2006.

Maria-Florina Balcan, Avrim Blum, Steve Hanneke, and Dravyansh Sharma. Robustly-reliable learners under poisoning attacks. In *Conference on Learning Theory*, pages 4498–4534. PMLR, 2022.

Jess Banks, Sidhanth Mohanty, and Prasad Raghavendra. Local statistics, semidefinite programming, and community detection. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1298–1316. SIAM, 2021.

Luca Becchetti, Carlos Castillo, Debora Donato, Ricardo Baeza-Yates, and Stefano Leonardi. Link analysis for web spam detection. *ACM Transactions on the Web (TWEB)*, 2(1):1–42, 2008.

Jarosław Błasiok, Rares-Darius Buhai, Pravesh K Kothari, and David Steurer. Semirandom planted clique and the restricted isometry property. In *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 959–969. IEEE, 2024.

Marco Bressan, Emmanuel Esposito, and Maximilian Thiessen. Efficient algorithms for learning monophonic halfspaces in graphs. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 669–696. PMLR, 2024.

Marco Bressan, Victor Chepoi, Emmanuel Esposito, and Maximilian Thiessen. Efficient algorithms for learning and compressing monophonic halfspaces in graphs. *arXiv preprint arXiv:2506.23186*, 2025.

Rares-Darius Buhai, Pravesh K Kothari, and David Steurer. Algorithms approaching the threshold for semi-random planted clique. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 1918–1926, 2023.

Nicolò Cesa Bianchi, Claudio Gentile, Fabio Vitale, and Giovanni Zappella. Active learning on trees and graphs. In *Proceedings of the 23rd Conference on Learning Theory*, pages 320–332. Omnipress, 2010.

Liang Chen, Jintang Li, Jiaying Peng, Tao Xie, Zengxu Cao, Kun Xu, Xiangnan He, Zibin Zheng, and Bingzhe Wu. A survey of adversarial learning on graphs. *arXiv preprint arXiv:2003.05730*, 2020.

Vincent Cohen-Addad, Tommaso DOrsi, and Aida Mousavifar. A near-linear time approximation algorithm for beyond-worst-case graph clustering. In *International Conference on Machine Learning*, pages 9208–9229. PMLR, 2024.

Vincent Cohen-Addad, Silvio Lattanzi, and Simon Meierhans. Algorithms and hardness for active learning on graphs. In *ICML*, 2025.

George Danezis and Prateek Mittal. Sybilinfer: Detecting sybil nodes using social networks. In *Ndss*, volume 9, pages 1–15. San Diego, CA, 2009.

Gautam Dasarathy, Robert Nowak, and Xiaojin Zhu. S^2: An efficient graph based active learning algorithm with application to nonparametric classification. In *Conference on Learning Theory*, pages 503–522. PMLR, 2015.

Jingqiu Ding, Tommaso d'Orsi, Rajai Nasser, and David Steurer. Robust recovery for stochastic block models. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 387–394. IEEE, 2022.

Jingqiu Ding, Tommaso dOrsi, Yiding Hua, and David Steurer. Reaching kesten-stigum threshold in the stochastic block model under node corruptions. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 4044–4071. PMLR, 2023.

Uriel Feige, MohammadTaghi Hajiaghayi, and James R Lee. Improved approximation algorithms for minimum-weight vertex separators. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 563–572, 2005.

Noah Fleming, Pravesh Kothari, Toniann Pitassi, et al. Semialgebraic proofs and efficient algorithm design. *Foundations and Trends® in Theoretical Computer Science*, 14(1-2):1–221, 2019.

Suprovat Ghoshal and Anand Louis. New approximation bounds for small-set vertex expansion. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2363–2375. SIAM, 2024.

Martin Grötschel, László Lovász, and Alexander Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 1981.

Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2. Springer Science & Business Media, 2012.

Quanquan Gu and Jiawei Han. Towards active learning on graphs: An error bound minimization approach. In *2012 IEEE 12th International Conference on Data Mining*, pages 882–887. IEEE, 2012.

Andrew Guillory and Jeff A Bilmes. Label selection on graphs. *Advances in Neural Information Processing Systems*, 22, 2009.

Steve Hanneke. Theory of disagreement-based active learning. *Foundations and Trends in Machine Learning*, 7(2-3):131–309, 2014.

Wei Jin, Yaxing Li, Han Xu, Yiqi Wang, Shuiwang Ji, Charu Aggarwal, and Jiliang Tang. Adversarial attacks and defenses on graphs. *ACM SIGKDD Explorations Newsletter*, 22(2):19–34, 2021.

Tsz Chiu Kwok, Lap Chi Lau, and Kam Chuen Tung. Cheeger inequalities for vertex expansion and reweighted eigenvalues. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 366–377. IEEE, 2022.

Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. Statistical properties of community structure in large social and information networks. In *Proceedings of the 17th international conference on World Wide Web*, pages 695–704, 2008.

Jiate Li, Meng Pang, Yun Dong, and Binghui Wang. Deterministic certification of graph neural networks against graph poisoning attacks with arbitrary perturbations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5020–5029, 2025.

Allen Liu and Ankur Moitra. Minimax rates for robust community detection. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 823–831. IEEE, 2022.

Xuanqing Liu, Si Si, Xiaojin Zhu, Yang Li, and Cho-Jui Hsieh. A unified framework for data poisoning attack to graph-based semi-supervised learning. *arXiv preprint arXiv:1910.14147*, 2019.

Anand Louis and Yury Makarychev. Approximation algorithms for hypergraph small-set expansion and small-set vertex expansion. *Theory of Computing*, 12(1):1–25, 2016.

Anand Louis, Prasad Raghavendra, and Santosh Vempala. The complexity of approximating vertex expansion. In *2013 IEEE 54th annual symposium on foundations of computer science*, pages 360–369. IEEE, 2013.

Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Approximation algorithms for semi-random partitioning problems. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 367–384, 2012.

Karl Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10(1):96–115, 1927.

Abedelaziz Mohaisen, Aaram Yun, and Yongdae Kim. Measuring the mixing time of social graphs. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 383–389, 2010.

Sidhanth Mohanty, Prasad Raghavendra, and David X Wu. Robust recovery for stochastic block models, simplified and generalized. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 367–374, 2024.

Ankur Moitra, William Perry, and Alexander S Wein. How robust are reconstruction thresholds for community detection? In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 828–841, 2016.

Prasad Raghavendra, David Steurer, and Prasad Tetali. Approximations for the isoperimetric and spectral profile of graphs and related parameters. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 631–640, 2010.

Maximilian Thiessen and Thomas Gärtner. Active learning of convex halfspaces on graphs. *Advances in Neural Information Processing Systems*, 34:23413–23425, 2021.

Nguyen Tran, Jinyang Li, Lakshminarayanan Subramanian, and Sherman SM Chow. Optimal sybil-resilient node admission control. In *2011 Proceedings IEEE INFOCOM*, pages 3218–3226. IEEE, 2011.

Luca Trevisan. Max cut and the smallest eigenvalue. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 263–272, 2009.

Yuanchen Wu and Yubai Yuan. Robust offline active learning on graphs. *Advances in Neural Information Processing Systems*, 37:58955–58983, 2024.

Han Yang, Binghui Wang, Jinyuan Jia, et al. Gnncert: Deterministic certification of graph neural networks against adversarial perturbations. In *The Twelfth International Conference on Learning Representations*, 2024.

Zhi Yang, Christo Wilson, Xiao Wang, Tingting Gao, Ben Y Zhao, and Yafei Dai. Uncovering social network sybils in the wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1): 1–29, 2014.

Haifeng Yu, Phillip B Gibbons, Michael Kaminsky, and Feng Xiao. Sybillimit: A near-optimal social network defense against sybil attacks. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 3–17. IEEE, 2008a.

Haifeng Yu, Michael Kaminsky, Phillip B Gibbons, and Abraham D Flaxman. Sybilguard: defending against sybil attacks via social networks. *IEEE/ACM Transactions on networking*, 16(3):576–589, 2008b.

Hui Zhang, Ashish Goel, Ramesh Govindan, Kahn Mason, and Benjamin Van Roy. Making eigenvector-based reputation systems robust to collusion. In *International Workshop on Algorithms and Models for the Web-Graph*, pages 92–104. Springer, 2004.

Appendix A. Algorithm for weak recovery

This section presents the main technical result behind Theorem 3. This result says that weak recovery is possible for a wide class of graphs and adversaries—all those where the subsets of corrupted vertices created by the adversary satisfy a small-expansion hypothesis akin to the one described in the overview above. We recall the definition of poorly expanding set (Theorem 5). Our main result is that, whenever the set I is poorly expanding, it is possible to achieve weak recovery.

Theorem 9 (Weak recovery for poorly expanding sets) *There is a universal $c_2 \in (0,1)$ such that what follows holds. Suppose I is $(\varepsilon, \eta m, m)$-expanding in G, where $m = |I|$ and*

$$\varepsilon = \frac{c_2^3 \gamma^5}{8 \log^3(1/\gamma) \sqrt{\log n}} \quad \text{and} \quad \eta = \frac{c_2 \gamma}{\log(1/\gamma)} \frac{n}{m}$$

for some $\gamma \in (0,1)$ small enough. Given G, γ, a label oracle for I, and $p = p(n) \in (0,1)$, in polynomial time and by making $\widetilde{O}\left(\frac{1}{\gamma}\right) \cdot O\left(\ln \frac{1}{p} + \sqrt{\log n} \cdot \left|\partial_G(I)\right|\right)$ queries one can compute a set $\widetilde{I}$ such that $|\widetilde{I} \triangle I| = O(\gamma n)$ with probability $1 - p$.

The constant c_2 is the one from Lemma 10 below. The rest of the section proves Theorem 9 and Theorem 3.

A.1. Proof of Theorem 3

Let G be the input graph, and let $n = |V(G)|$, $m = |I|$, and $n^* = n - m$. First, note that by querying the labels of $O\left(\frac{1}{\gamma} \ln \frac{1}{p}\right)$ uniform random vertices of G we can detect with probability $1 - p$ if $m < \gamma n$, in which case returning $\emptyset$ satisfies the guarantees. Suppose then $m \geqslant \gamma n$, and recall that $m \leqslant \frac{n}{2}$ by the assumption of the model. Using assumption (ii) of Theorem 3 and the relations between ϕ_G and $|\partial_G|$ and between $\partial_{\eta m}$ and $\phi_{\eta m}$, we get:

$$\begin{aligned}
\phi_G(I) &= \frac{|\partial_G(I)|}{m(n-m)} && \text{(A.1)} \\
&\leqslant \frac{2b}{\gamma n^2} && \text{(A.2)} \\
&\leqslant O\left(\frac{\gamma^5 \cdot \partial_{\eta m}(G^*)}{\log^3(1/\gamma) \cdot \sqrt{\log n} \cdot n^2}\right) && \text{(A.3)} \\
&\leqslant O\left(\frac{\gamma^5 \cdot \phi_{\eta m}(G^*)}{\log^3(1/\gamma) \cdot \sqrt{\log n}}\right) && \text{(A.4)}
\end{aligned}$$

By making the constants small enough we obtain $\phi_G(I) \leqslant \varepsilon \cdot \phi_{\eta m}(G^*)$, where ε is as in Theorem 9. This proves that I is $(\varepsilon, \eta m, m)$-expanding in G. Hence we may apply Theorem 9, which yields the result.

A.2. Proof of Theorem 9

Even though our statements are self-contained, we fix the following notation throughout the section to ease readability: let G^* be a graph, let G be the input graph and let $I \coloneqq V(G) \setminus V(G^*)$ be the set of malicious nodes. As outlined in Section 3, our algorithm consists of multiple iterations. In each iteration we seek to remove a large fraction of the corrupted nodes I while removing only a few vertices from $V(G^*)$. The procedure then ends when sufficiently many vertices in I have been removed. At the beginning of round $i = 0, 1, \ldots$, let G_i be the remaining subgraph with vertices $V_i = V(G_i)$, let $I_i = V_i \cap I$ be the subset of I in G_i, that is the set malicious vertices in G_i, and let G_i^* be the subgraph of G_i induced by $V_i \setminus I$. Notice that $G_0 = G\,, G_0^* = G^*\,, I_0 = I$.

A.2.1. SINGLE ITERATION

At a high level, at each iteration $i = 0, 1, \ldots$, the algorithm performs the following steps.

1. Query the oracle $\mathcal{O}_G$ to approximately determine the size of I_i. If the estimate is smaller than $\gamma \cdot |I|$ then *return* G_i.
2. Otherwise, find a set S_i of size $\Omega(|I_i|)$ with small vertex expansion and large intersection with I_i.
3. Set $G_{i+1} = G_i[V_i \setminus S_i]$ and repeat.

Step 1 is achieved performing sufficiently many random queries to the oracle $\mathcal{O}_G$ and analyzing their outcome via standard concentration bounds. The crucial step (step 2) of finding the set S_i at each iteration is captured by the following lemma. It states that, given a graph G containing an $(\varepsilon, \eta m, m)$-expanding set I, there exists a polynomial-time algorithm that, for a good choice of the parameters, finds a large set S almost entirely contained in I making only few queries to the oracle. Because at each iteration i the remaining subset of corrupted nodes I_i will be poorly expanding, this will allow us to remove a large fraction of corrupted nodes.

Lemma 10 *There exist universal constants $c_1, c_2, c_3, c_4, c_5 \in (0, 1)$ and a randomized polynomial-time algorithm with the following guarantees. Let $G = (V, E)$ be any n-vertex graph and $I \subseteq V$. Suppose m, α, η satisfy:*

(1) $0 \leqslant m \leqslant n$

(2) $\alpha \geqslant \frac{1}{c_1}\left(\sqrt{\log n} + \frac{n}{m}\right)$

(3) $0 < \eta \leqslant \frac{c_2}{2}$

(4) $m \leqslant |I| \leqslant \frac{m}{c_4}$

(5) I *is* $(\varepsilon, \eta m, m)$*-expanding in* G *for some* $\varepsilon \leqslant \frac{\eta^2}{\alpha}$

Given G, access to a label oracle for I, m, and $p \in (0, 1)$, the algorithm makes at most $\alpha \cdot |\partial_G(I)| + O\left(\log \frac{1}{p}\right)$ oracle queries and returns $S \subseteq V(G)$ that with probability $1 - p$ satisfies:

(i) $|\partial_G(S)| \leqslant \frac{\alpha}{c_5} \cdot |\partial_G(I)|$

(ii) $\min\{|S|, |V(G) \setminus S|\} \geqslant c_3 \cdot |I|$

(iii) $|S \setminus I| \leqslant \frac{\eta}{c_2} \cdot |S|$.

We prove Lemma 10 in Appendix B and directly use it here.

A.2.2. MULTIPLE ITERATIONS

Lemma 10 suggests that if I is $(\varepsilon, \eta m, m)$-expanding, then we should expect at the first iteration to find a set S with small external boundary that is heavily correlated with I. Call the i-th iteration *successful* if the high-probability claim of Lemma 10 holds. The next statement shows that if a sequence of $i-1$ iterations is successful, then we should expect the premises of Lemma 10 to be satisfied also at the i-th iteration.

Lemma 11 *Let $c_2 \in (0,1)$ be the universal constant of Lemma 10, and suppose I_0 satisfies the hypotheses of Lemma 10 with parameters*

$$m_0 = |I_0|,\ n_0 = |V(G_0)|,\ \eta_0 = \frac{c_2\,\gamma}{\log(1/\gamma)}\frac{n_0}{m_0},\ \varepsilon_0 = \left(\frac{c_2\,\gamma}{2\log(1/\gamma)}\right)^3 \cdot \frac{1}{\alpha_0},\ \alpha_0 = \alpha$$

for some $\gamma \in \left(0, \frac{c_2}{2}\right)$ and $\alpha \geqslant \frac{1}{\gamma} \cdot \frac{1}{c_1}\left(\sqrt{\log n_0} + \frac{n_0}{m_0}\right)$. Then, for all $1 \leqslant i \leqslant \frac{2\log^2(1/\gamma)}{\gamma^2}$, if the first i iterations are successful and $|I_i| \geqslant \gamma |I_0|$ then I_i satisfies again the hypotheses of Lemma 10.

We defer the proof of Lemma 11 to Appendix A.3. We are now ready to prove Theorem 9.

Proof of Theorem 9 First, observe that if $I = I_0$ satisfies the hypotheses of Theorem 9 then it satisfies the hypotheses of Lemma 11 as well, with $\varepsilon = \varepsilon_0$, $\eta = \eta_0$ and so on. To apply Lemma 11 we shall thus see how to make the first i iterations successful. Finally, we will bound the total error $|\widetilde{I} \setminus I|$ as well as the number of queries.

Let $r = \Theta(\gamma n)$ sufficiently small. The idea is to repeatedly check if the current graph contains more than r vertices of I; if not, then stop, otherwise perform one iteration of the algorithm in Lemma 10. Formally, let $G_0 = G$ and $I_0 = I$ and $\widetilde{I}_0 = \emptyset$. Before starting, sample and query the label of $O\left(\frac{1}{\gamma}\ln\frac{1}{p}\right)$ uniform random vertices of G to obtain an estimate $\widehat{|I|}$ of $|I|$. By standard Chernoff bounds we can make the following claims hold with probability $1 - \frac{p}{3}$. First, if $\widehat{|I|} < \frac{r}{2}$ then $|I| < r$. In this case we stop and return $\emptyset$. Otherwise, $|I| \leqslant \widehat{|I|} \leqslant 2|I|$. In this case we compute a sufficiently large $k = O\left(\log\frac{\widehat{|I|}}{r}\right) = O\left(\log\frac{1}{\gamma}\right)$. Then, for $i = 0, 1, \ldots$ perform the following procedure. First, we obtain an estimate $\widehat{|I_i|}$ by sampling and querying $O\left(\frac{n_i}{r}\ln\frac{k}{p}\right)$ uniform random vertices of G_i, as described above (for $i = 0$ we can just reuse that very estimate). By the same Chernoff bound arguments as above, with probability at least $1 - \frac{p}{3k}$ we have that if $\widehat{|I|} < \frac{r}{2}$ then $|I| < r$, and otherwise $|I_i| \leqslant \widehat{|I_i|} \leqslant 2|I_i|$. If $\widehat{|I|} < \frac{r}{2}$, then return $\widetilde{I}_i$. Otherwise, run one iteration of the algorithm of Lemma 10 to obtain S_i, using the parameter $p_i = \frac{p}{3k}$ for the failure probability. Compute $\widetilde{I}_{i+1} = \widetilde{I}_i \cup S_i$ and $G_{i+1} = G_i \setminus S_i$, and move to the next iteration. By a union bound, and by Lemma 11, the estimate as well the (at most) k iterations are successful with overall probability at

least $1-p$. If that is the case, as by point (ii) of Lemma 10 we have $|I_{i+1}| \leqslant |I_i|(1-c_3)$, then for some $i \leqslant k$ we have $|I_i| < r$, at which point the algorithm stops and returns $\widetilde{I} = \widetilde{I}_i$.

Now let us show that $|\widetilde{I} \triangle I| = |I \setminus \widetilde{I}| + |\widetilde{I} \setminus I| = O(\gamma n)$. On the one hand, $|I \setminus \widetilde{I}| \leqslant |I_k| = O(r) = O(\gamma n)$. It remains to show that $|\widetilde{I} \setminus I| \leqslant O(\gamma n)$, too; choosing the constants small enough yields the result. First, note that the choice of our parameters satisfies Lemma 11. Now:

$$|\widetilde{I} \setminus I| = \left| \bigcup_{i=0}^{k-1} S_i \setminus I_i \right| \tag{A.5}$$

$$\leqslant \sum_{i=0}^{k-1} |S_i \setminus I_i| \tag{A.6}$$

$$\leqslant \sum_{i=0}^{k-1} \frac{\eta_i}{c_2} |S_i| \qquad \text{by (iii) of Lemma 10} \tag{A.7}$$

$$\leqslant \sum_{i=0}^{k-1} \frac{\eta\, m}{c_2} \qquad \text{as } \eta_i = \eta \cdot \frac{m}{m_i} \text{ and } |S_i| = m_i \tag{A.8}$$

$$\leqslant \gamma n \qquad \text{as } \eta \leqslant \frac{\gamma n c_2}{km} \tag{A.9}$$

Therefore $|\widetilde{I} \setminus I| = O(\gamma n)$.

Finally, let us analyze the number of queries. First, note that, if $m = o(r)$, then the algorithm immediately detects that $|I| < r$ and stops, making only $O\left(\frac{1}{\gamma} \ln \frac{1}{p}\right)$ queries, see above. We can therefore turn to the case $m = \Omega(r) = \Omega(\gamma n)$. In this case, using the bounds of Lemma 10, the total number of queries performed is at most:

$$\sum_{i=0}^{k} O\left(\frac{n_i}{r} \ln \frac{1}{p_i} + \alpha \cdot |\partial_{G_i}(I_i)| + \log \tfrac{1}{p_i}\right) \tag{A.10}$$

$$= \sum_{i=0}^{k} O\left(\frac{1}{\gamma} \ln \frac{3k}{p} + \alpha \cdot |\partial_{G_i}(I_i)|\right) \tag{A.11}$$

$$= O\left(\frac{1}{\gamma} \ln\left(\frac{1}{\gamma}\right) \ln \frac{\ln 1/\gamma}{p} + \sum_{i=0}^{k} \alpha \cdot |\partial_{G_i}(I_i)|\right) \tag{A.12}$$

$$= O\left(\frac{1}{\gamma} \ln\left(\frac{1}{\gamma}\right) \ln \frac{\ln 1/\gamma}{p} + \left(\sqrt{\log n} + \frac{n}{m}\right) \cdot \ln\left(\frac{m}{r}\right) \cdot |\partial_G(I)|\right) \tag{A.13}$$

$$= O\left(\frac{1}{\gamma} \ln\left(\frac{1}{\gamma}\right) \ln \frac{\ln 1/\gamma}{p} + \left(\sqrt{\log n} + \frac{1}{\gamma}\right) \cdot \ln\left(\frac{1}{\gamma}\right) \cdot |\partial_G(I)|\right) \tag{A.14}$$

$$= \widetilde{O}\left(\frac{1}{\gamma}\right) \cdot O\left(\ln \frac{1}{p} + \sqrt{\log n} \cdot \left|\partial_G(I)\right|\right). \tag{A.15}$$

This concludes the proof. ■

A.3. Expansion parameters gracefully degrade: proof of Lemma 11

In this section we prove Lemma 11, thus showing that throughout subsequent successful iterations the degradation of our parameters of interest is tolerable. Our proof requires three ingredients. The first is monotonicity of the external boundary of any set throughout the iterations.

Lemma 12 (Monotonicity of ∂) *For any graph $G = (V, E)$ and $A, B \subseteq V$ we have:*

$$\partial_{G \setminus B}(A \setminus B) \subseteq \partial_G(A)\,.$$

Proof Let $v \in \partial_{G\setminus B}(A \setminus B)$. Then, by definition,

$$v \in (V \setminus B) \setminus (A \setminus B) = (V \setminus B) \setminus A \subseteq V \setminus A \tag{A.16}$$

and moreover there exists $u \in A \setminus B \subseteq A$ such that $\{u, v\} \in E(G \setminus B) \subseteq E(G)$. Hence $v \in \partial_G(A)$. ■

The second ingredient is a statement that shows how given a graph G, if we remove a set S, the external boundary of other sets decreases at most by a factor comparable to the frontier of S. For a graph G and any positive $t < |V(H)|$, define

$$\partial_G(t) \coloneqq \min_{\substack{Y \subseteq V(G) \\ t \leqslant |Y| \leqslant |V(G)| - t}} |\partial_G(Y)|\,. \tag{A.17}$$

Lemma 13 (Bounding the change in the frontier) *Let $G = (V, E)$ be any graph. For every $V^*, S \subseteq V$ and $t \geqslant 0$, writing $G^* = G[V^*]$,*

$$\partial_{G^* \setminus S}(t) \geqslant \partial_{G^*}(t) - |\partial_G(S)|\,. \tag{A.18}$$

Proof Let $G^* = (V^*, E^*)$ and $S^* = S \cap V^*$. For any $X \subseteq V^* \setminus S^*$ let $Y_X = X \cup S^*$. Now observe:

$$\begin{aligned}
\partial_{G^*}(t) &= \min_{\substack{Y \subseteq V^* \\ t \leqslant |Y| \leqslant |V^*| - t}} |\partial_{G^*}(Y)| && \text{by definition of } \partial && \text{(A.19)}\\
&\leqslant \min_{\substack{X \subseteq V^* \setminus S^* \\ t \leqslant |X| \leqslant |V^*| - t - |S^*|}} |\partial_{G^*}(Y_X)| && \text{by taking } \min(\cdot) \text{ over a subset} && \text{(A.20)}\\
&= \min_{\substack{X \subseteq V^* \setminus S^* \\ t \leqslant |X| \leqslant |V^* \setminus S^*| - t}} |\partial_{G^*}(Y_X)| && \text{as } S^* \subseteq V^*\,. && \text{(A.21)}
\end{aligned}$$

Notice moreover that

$$|\partial_{G^*}(Y_X)| \leqslant |\partial_{G^* \setminus S^*}(X)| + |\partial_{G^* \setminus X}(S^*)| \leqslant |\partial_{G^* \setminus S^*}(X)| + |\partial_G(S)|\,, \tag{A.22}$$

where the second inequality holds by applying Lemma 12 to $|\partial_{G^* \setminus X}(S^*)|$ with $A = S$ and $B = V(G) \setminus (V(G^*) \setminus X)$, and noting that $S \setminus B = S^* \setminus X = S^*$. Applying this bound to Equation (A.21), using $V^* \setminus S^* = V(G^* \setminus S^*)$, and noting that $G^* \setminus S^* = G^* \setminus S$ yields:

$$\partial_{G^*}(t) \leqslant \min_{\substack{X \subseteq V(G^*) \setminus S^* \\ t \leqslant |X| \leqslant |V(G^*) \setminus S^*| - t}} \Big(|\partial_{G^* \setminus S^*}(X)| + |\partial_G(S)| \Big) \tag{A.23}$$

$$= \partial_{G^* \setminus S^*}(t) + |\partial_G(S)| = \partial_{G^* \setminus S}(t) + |\partial_G(S)| \,. \tag{A.24}$$

This completes the proof. ■

The third ingredient is a lemma showing that the poor-expansion properties of the residual corrupted set I_i degrade gracefully with i, provided the initial set I_0 is sufficiently poorly expanding in G_0 and the application of Lemma 10 is successful at every iteration.

Lemma 14 (Poor expansion of I_i) *Suppose $I_0 = I$ is $(\varepsilon, \eta m, m)$-expanding in $G_0 = G$, where $m = m_0 = |I| \geqslant \gamma n \geqslant 3$ and $n = n_0 = |V(G_0)|$, for some $\eta < \frac{\gamma}{3}$ and $\gamma \leqslant \frac{c_2}{2} \leqslant \frac{1}{2}$, see Lemma 10. Suppose moreover that we perform i successful iterations, where α satisfies Lemma 10 at every iteration, and that $|I_i| \geqslant \gamma n$. Then $|I_i| \leqslant \frac{n_i}{2}$ and I_i is $(\varepsilon_i, \eta m, m_i)$-expanding, where $m_i = |I_i|$ and:*

$$\varepsilon_i < \frac{n_i}{m_i} \cdot \left(\frac{\varepsilon}{\eta - 2i\varepsilon\alpha} \right) \tag{A.25}$$

Proof To see that $|I_i| \leqslant \frac{n_i}{2}$, observe that, by Lemma 10, at every successful iteration the vertices removed from I_{i-1} are no less than those removed from $V_{i-1} \setminus I_{i-1}$. As $|I_0| = m \leqslant \frac{n}{2}$ this proves the claim. To prove that I_i is $(\varepsilon_i, \eta m, m_i)$-expanding, it suffices to show:

$$\frac{\phi_{G_i^*}(\eta m)}{\phi_{G_i}(I_i)} > \frac{m_i}{n_i} \cdot \left(\frac{\eta}{\varepsilon} - 2i\alpha \right) \tag{A.26}$$

First, by definition of ϕ, and since $\eta m \leqslant n_i^*$, we have:

$$\phi_{G_i^*}(\eta m) = \min_{\substack{\emptyset \neq S \subset V(G_i^*) \\ \eta m \leqslant |S| \leqslant n_i^* - \eta m}} \frac{|\partial_{G_i^*}(S)|}{|S| \cdot |V_i^* \setminus S|} \geqslant \frac{\partial_{G_i^*}(\eta m)}{(n_i^*/2)^2} \tag{A.27}$$

Note that the $\min$ is over a nonempty domain. Indeed, $n_i^* \geqslant m_i \geqslant \gamma n$, and together with $\eta < \frac{\gamma}{3}$ this implies:

$$(n_i^* - \eta m) - \eta m = n_i^* - 2\eta m > n_i^* - \frac{2}{3}\gamma m \geqslant m_i - \frac{2}{3}\gamma n \geqslant \frac{\gamma}{3} n \geqslant 1 \tag{A.28}$$

Moreover, again by definition of ϕ:

$$\phi_{G_i}(I_i) = \frac{\partial_{G_i}(I_i)}{m_i(n_i - m_i)} \tag{A.29}$$

Overall we thus obtain:

$$\frac{\phi_{G_i^*}(\eta m)}{\phi_{G_i}(I_i)} \geqslant \frac{\partial_{G_i^*}(\eta m)}{\partial_{G_i}(I_i)} \cdot \frac{m_i(n_i - m_i)}{(n_i^*/2)^2} \tag{A.30}$$

Since $n_0^* \geqslant \frac{n_0}{2}$ and in every successful iteration the majority of removed vertices are from I, then $m_i \leqslant \frac{n_i}{2}$. Since moreover $n_i^* \leqslant n_i$:

$$\frac{\partial_{G_i^*}(\eta m)}{\partial_{G_i}(I_i)} \cdot \frac{m_i(n_i - m_i)}{(n_i^*/2)^2} \geqslant \frac{\partial_{G_i^*}(\eta m)}{\partial_{G_i}(I_i)} \cdot \frac{m_i\,(n_i/2)}{n_i^2/4} = \frac{\partial_{G_i^*}(\eta m)}{\partial_{G_i}(I_i)} \cdot 2\frac{m_i}{n_i} \tag{A.31}$$

We shall now bound the ratio $\frac{\partial_{G_i^*}(\eta m)}{\partial_{G_i}(I_i)}$. More precisely, we bound $\partial_{G_i^*}(\eta m)$ from below using $\partial_{G_i}(I_i)$. To begin with, iterate Lemma 13 to obtain:

$$\partial_{G_i^*}(\eta m) \geqslant \partial_{G_0^*}(\eta m) - \sum_{j=0}^{i-1} \left|\partial_{G_j}(S_j)\right| \tag{A.32}$$

$$\geqslant \partial_{G_0^*}(\eta m) - \sum_{j=0}^{i-1} \alpha \cdot \left|\partial_{G_j}(I_j)\right| \qquad \text{by Lemma 10} \tag{A.33}$$

$$\geqslant \partial_{G_0^*}(\eta m) - i\alpha \cdot \left|\partial_{G_0}(I_0)\right| . \tag{A.34}$$

The last inequality holds since for every $j = 1, \dots, i$ we have $\partial_{G_j}(I_j) \subseteq \partial_{G_0}(I_0)$, as given by an application of Lemma 12 to $\partial_{G_0}(I_0)$ using $B = S_0 \cup \cdots \cup S_{i-1}$. Next, we shall bound the two terms of Equation (A.34) in terms of $\phi_{G_i}(I_i)$. For the first term:

$$\partial_{G_0^*}(\eta m) \geqslant \phi_{G_0^*}(\eta m) \cdot (\eta m)(n^* - \eta m) \tag{A.35}$$

$$\geqslant \phi_{G_0^*}(\eta m) \cdot (\eta m)\frac{n-m}{2} \qquad \text{as } n^* \geqslant \frac{n}{2} \text{ and } \eta m \leqslant \frac{1}{2}m \tag{A.36}$$

$$= \phi_{G_0^*}(\eta m) \cdot \frac{\eta}{2} m(n-m) \tag{A.37}$$

$$> \phi_{G_0}(I_0) \cdot \frac{\eta}{2\varepsilon} m(n-m) \qquad \text{as } I_0 \text{ is } (\varepsilon, \eta m, m)\text{-expanding in } G_0 . \tag{A.38}$$

For the second term:

$$|\partial_{G_0}(I_0)| = \phi_{G_0}(I_0) \cdot m(n-m) \tag{A.39}$$

We can then use Equation (A.34) to obtain:

$$\partial_{G_i^*}(\eta m) \geqslant \phi_{G_0}(I_0) \cdot m(n-m) \cdot \left(\frac{\eta}{2\varepsilon} - i\alpha\right) \qquad \text{by Equations (A.38) and (A.39)} \tag{A.40}$$

$$= \partial_{G_0}(I_0) \cdot \left(\frac{\eta}{2\varepsilon} - i\alpha\right) \qquad \text{definition of } \phi_{G_0}(I_0) \tag{A.41}$$

$$\geqslant \partial_{G_i}(I_i) \cdot \left(\frac{\eta}{2\varepsilon} - i\alpha\right) \qquad \text{as } \partial_{G_0}(I_0) \supseteq \partial_{G_i}(I_i) \tag{A.42}$$

Finally, by using Equation (A.42) in Equation (A.31), we obtain:

$$\frac{\phi_{G_i^*}(\eta m)}{\phi_{G_i}(I_i)} > \frac{m_i}{n_i} \cdot \left(\frac{\eta}{\varepsilon} - 2i\alpha\right) \tag{A.43}$$

which yields Equation (A.27) and concludes the proof. ■

Taken together, these results allow us to prove Lemma 11.

Proof of Lemma 11 We show that, if the hypotheses hold, then I_i satisfies the assumptions of Lemma 10 with m, n, η, α respectively replaced by:

$$m_i = |I_i|,\ n_i = |V(G_i)|,\ \eta_i = \eta_0 \frac{m_0}{m_i},\ \alpha_i = \alpha_0 . \tag{A.44}$$

Let us verify each assumption in turn.

(1) Trivially $m_i \leqslant n_i$.

(2) Since $n_i \leqslant n_0$ and $m_i \geqslant \gamma n_0 \geqslant \gamma m_0$, and by the assumptions on $\alpha_i = \alpha_0$,

$$\alpha_i = \alpha_0 \geqslant \frac{1}{\gamma}\frac{1}{c_1}\left(\sqrt{\log n_0} + \frac{n_0}{m_0}\right) \geqslant \frac{1}{c_1}\left(\sqrt{\log n_i} + \frac{n_i}{m_i}\right) . \tag{A.45}$$

(3) By the definition of η_i, the choice of η_0, the fact that $m_0 \geqslant \gamma n_0$, and the assumption $\gamma_0 < \frac{c_2}{2}$,

$$\eta_i = \eta\frac{m_0}{m_i} = \frac{c_2\,\gamma_0}{4\lg(1/\gamma_0)}\frac{n}{m} \leqslant \frac{c_2}{4\lg(1/\gamma_0)} < \frac{c_2}{2} \leqslant \frac{1}{2}, \tag{A.46}$$

(4) Trivially $m_i \leqslant |I_i| \leqslant \frac{m_i}{c_4}$ since $m_i = |I_i|$ and $c_4 \leqslant 1$.

(5) Theorem 14 shows that, if the first i iterations are successful, then I_i is $(\varepsilon_i, \eta_0 m_0, m_i)$-expanding in G_i for the value:

$$\varepsilon_i = \frac{n_i}{m_i} \cdot \left(\frac{\varepsilon_0}{\eta_0 - 2i\varepsilon_0\alpha_0}\right) \tag{A.47}$$

Before dealing with ε_i, note that $\eta_0 m_0 = \eta_i m_i$, thus I_i is $(\varepsilon_i, \eta_i m_i, m_i)$-expanding in G_i. To satisfy the hypothesis it then remains to show that $\varepsilon_i \leqslant \frac{\eta_i^2}{\alpha_0}$, that is,

$$\frac{n_i}{m_i} \cdot \left(\frac{\varepsilon_0}{\eta_0 - 2i\varepsilon_0\alpha_0}\right) \leqslant \frac{\eta_i^2}{\alpha_0} = \frac{\eta_0^2}{\alpha_0}\left(\frac{m_0}{m_i}\right)^2 \tag{A.48}$$

First, let us simplify the right-hand side. To this end observe that, by the assumption on i and the choice of ε_0, α_0 and η_0:

$$2i \cdot \varepsilon_0\alpha_0 \leqslant \frac{4\,\lg^2(1/\gamma)}{\gamma^2} \cdot \frac{c_2\,\gamma^3}{8\lg^3(1/\gamma)} = \frac{c_2^3\,\gamma}{2\lg(1/\gamma)} \leqslant \frac{1}{2}\eta_0 \tag{A.49}$$

Hence, $\eta_0 - 2i\varepsilon_0\alpha_0 \geqslant \eta_0/2$, and to prove Equation (A.48) it suffices to show:

$$\frac{2\,n_i\varepsilon_0}{m_i\eta_0} \leqslant \frac{\eta_0^2}{\alpha_0}\left(\frac{m_0}{m_i}\right)^2 \tag{A.50}$$

which, by rearranging terms, is equivalent to:

$$2\varepsilon_0\alpha_0 \leqslant \eta_0^3\frac{m_0^2}{m_i n_i} . \tag{A.51}$$

By further substituting the values of ε_0 and η_0, and cancelling out terms, we need to show:

$$\frac{2\,c_2^3\gamma^3}{8\,\lg^3(1/\gamma)} \leqslant \frac{c_2^3\,\gamma^3}{\lg^3(1/\gamma)}\frac{n_0^3}{m_0 m_i^2} \tag{A.52}$$

which always holds since $m_0, m_i \leqslant n_0$.

The proof is complete. ■

Appendix B. Unbalanced vertex expansion

In this section we prove Theorem 4. We then use it to obtain Lemma 10 in Appendix B.3.

Theorem *[Restatement of Theorem 4] There exists a randomized polynomial-time algorithm that, given an n-vertex graph G and $0 < m \leqslant n/2$, returns $S \subset V$ satisfying*

(i) $\phi_G(S) \leqslant \phi_m(G) \cdot O\big(\sqrt{\log n} + \frac{n}{m}\big)$,

(ii) $\min\{|S|, |V \setminus S|\} = \Omega(m)$.

Theorem 4 can be seen as an extension of Feige et al. (2005) and relies on the sum-of-squares framework. We devote most of the section to it. Necessary background on sum-of-squares can be found in Appendix C. The algorithm behind Theorem 4 consists of two steps. In the first, we obtain a degree-4 pseudo-distribution that is consistent with a specific, natural relaxation of our vertex expansion problem. In the second, we carefully round this pseudo-distribution into an integral solution.

Relaxation for vertex expansion. Let $G = (V, E)$ be an n-vertex graph and let $0 < m \leqslant n/2$ be an integer. A vertex separator $U \subseteq V$ is a set such that $V \setminus U$ results into two non-empty disconnected pieces $S \subseteq V \setminus U$ and $V \setminus (S \cup U)$. We adopt the convention that $|S| \leqslant |V \setminus (U \cup S)|$. Notice that by construction

$$\phi(S) \coloneqq \frac{|U|}{|S| \cdot |V \setminus S|} \,. \tag{B.1}$$

We consider the system of polynomial inequalities $\mathcal{P}_{\bar{m}}(G)$ below, which captures the problem of finding a set of minimum vertex expansion with a given fixed size. For every $i \in V$ we introduce two variables x_i, y_i ; these should be understood as indicators of whether i is in S or in $V \setminus (U \cup S)$, or none of the two. We also assume $m \leqslant \bar{m} \leqslant n/2$ to be the size of the set—or of its complement—with minimum vertex expansion among all sets of size in $[m, n-m]$. Since guessing the at most n possible values for $\bar{m}$ and picking the best one can increase the running time by at most a linear multiplicative factor, this assumption can be made without loss of generality.

$$\min \sum_{i \in V} 1 - x_i^2 - y_i^2 \qquad \text{s.t.} \qquad \left\{ \begin{aligned} \forall i \in V \,, \quad & x_i^2 = x_i \\ \forall i \in V \,, \quad & y_i^2 = y_i \\ \forall i \in V \,, \quad & x_i y_i = 0 \\ \forall ij \in E \,, \quad & x_i y_j = 0 \\ & \sum_{ij \in V} (x_i - x_j)^2 = \bar{m}(n - \bar{m}) \end{aligned} \right\} =: \qquad (\mathcal{P}_{\bar{m}}(G))$$

The constraints $\{x_i^2 = x_i\}$ and $\{y_i^2 = y_i\}$ enforce feasible solutions to be Boolean. The constraint $\{x_i y_i = 0\}$ is used to enforce that no vertex is both in S and $V \setminus (U \cup S)$. As for every edge $ij \in E$ we have the constraint $\{x_i y_j\} = 0$, in any feasible solution the set of vertices i with

$x_i = y_i = 0$ must be a vertex separator in G. Finally, the last constraint controls the size of S and its complement.

Crucially, any degree-4 pseudo-distribution μ satisfying $\mathcal{P}_{\bar{m}}(G)$ defines a pseudo-metric d_μ of negative type over V such that $d_\mu(i,j) = \widetilde{\mathbb{E}}_\mu\big[(x_i - x_j)^2\big]$ (see Appendix C). Hence, our rounding algorithm leverages the structure of this pseudo-metric, building on previous work of Feige et al. (2005) and on the celebrated structure theorem of Arora et al. (2009).

B.1. Large sets with small vertex expansion

As remarked above, any degree-4 pseudo-distribution μ satisfying $\mathcal{P}_{\bar{m}}(G)$ defines a pseudo-metric d_μ of negative type over V such that $d_\mu(i,j) = \widetilde{\mathbb{E}}_\mu\big[(x_i - x_j)^2\big]$ (see Appendix C). For sets $X, Y \subseteq V$ we let $d_\mu(X,Y) = \min_{i\in X, j\in Y} \widetilde{\mathbb{E}}_\mu\big[(x_i - x_j)^2\big]$ by a slight abuse of notation.

We introduce a notion of well-separated sets in a metric space, which we will use for the pseudo-metric defined by μ.

Definition 15 (Well-separated sets) *Given a finite pseudo-metric space (V,d). We say that $X, Y \subseteq V$ are Δ-separated w.r.t. d if $\Delta \leqslant d(X,Y) \leqslant 2$.*

We let $r_\mu \coloneqq \frac{1}{n^2}\sum_{j,j'\in V} d_\mu(j,j')$ and we also write $\frac{d_\mu}{r_\mu} : V\times V \to \mathbb{R}$ for the pseudo-metric obtained rescaling d_μ by $1/r_\mu$. We let $B_\mu(i,q)$ be the ball of radius q around $i \in V$ induced by d_μ. From this point forward, we drop the subscripts from d_μ and r_μ when the context is clear.

Our first ingredient is the following generalization of (Feige et al., 2005, Proposition 3.10), which relates the distance in the pseudo-metric with the objective value of the solution.

Lemma 16 *Let μ be a degree-4 pseudo-distribution consistent with $\mathcal{P}_{\bar{m}}(G)$. Then, for every $ij \in E$*

$$d(i,j) \leqslant 2\widetilde{\mathbb{E}}\big[1 - x_i^2 - x_j^2\big] + 2\widetilde{\mathbb{E}}\big[1 - y_i^2 - y_j^2\big]\,.$$

Proof Observe that

$$\begin{aligned}
d(i,j) &= \widetilde{\mathbb{E}}\Big[(x_i - x_j)^2\Big] && \text{(B.2)}\\
&= \widetilde{\mathbb{E}}\Big[((x_i + y_i - 1) - (x_j + y_i - 1))^2\Big] && \text{(B.3)}\\
&\leqslant 2\widetilde{\mathbb{E}}\Big[(x_i + y_i - 1)^2 + (x_j + y_i - 1)^2\Big] && \text{(B.4)}\\
&= 2\widetilde{\mathbb{E}}\big[x_i^2 + y_i^2 + 1 - 2x_i - 2y_i + x_j^2 + y_i^2 + 1 - 2x_j - 2y_i\big] && \text{(B.5)}\\
&= 2\widetilde{\mathbb{E}}\big[2 - x_i^2 - x_j^2 - 2y_i^2\big]\,. && \text{(B.6)}
\end{aligned}$$

Similarly we have

$$d(i,j) \leqslant 2\widetilde{\mathbb{E}}\big[2 - x_i^2 - x_j^2 - 2y_j^2\big]\,. \tag{B.7}$$

Combining the two we get

$$d(i,j) \leqslant \widetilde{\mathbb{E}}\big[4 - 2x_i^2 - 2y_i^2 - 2x_j^2 - 2y_j^2\big] = 2\widetilde{\mathbb{E}}\big[2 - x_i^2 - y_i^2 - x_j^2 - y_j^2\big] \tag{B.8}$$

as desired. ■

Our second ingredient is a characterization of pseudo-distributions consistent with $\mathcal{P}_{\bar{m}}(G)$, which relies on the following notion of well-spreadness.

Definition 17 (Well-spread) *Let μ be a degree-4 pseudo-distribution consistent with $\mathcal{P}_{\bar{m}}(G)$. We say μ is* well-spread *if there exists $i \in V$ such that,*

$$\sum_{j,j' \in B(i,2r)} d(j,j') \geqslant \frac{r \cdot n^2}{16} = \sum_{j,j' \in V} \frac{d(j,j')}{16} \,.$$

Whenever μ is well-spread, after an appropriate rescaling of the pseudo-metric induced by μ, we can use it to find sets X, Y that are $O(1/\sqrt{\log n})$-separated.

Lemma 18 *Let μ be a degree-4 pseudo-distribution consistent with $\mathcal{P}_{\bar{m}}(G)$. If μ is well-spread then there exists $X, Y \subseteq V$ satisfying:*

(i) $\min\{|X|, |Y|\} \geqslant c \cdot n$ *, for some universal constant* $c > 0$ *,*

(ii) $2 \geqslant \frac{d}{r}(X,Y) \geqslant c'/\sqrt{\log n}$ *, for some universal constant* $c' > 0$ *.*

Moreover, there exists a Las Vegas polynomial-time algorithm that finds X, Y.

Lemma 18 is a consequence of the structure theorem of Arora et al. (2009). We present a proof in Appendix B.2 and directly use it here. Whenever the given pseudo-distribution is not well-spread, its induced pseudo-metric must have a small ball containing a large fraction of the elements in V.

Lemma 19 *Let μ be a degree-4 pseudo-distribution consistent with $\mathcal{P}_{\bar{m}}(G)$. If μ is not well-spread then there exists $i \in V$ such that $|B(i, r/4)| \geqslant n/4$.*

Proof Suppose that no such i exists. Pick $k \in V$ such that $\frac{1}{n}\sum_{j \in V} d(j,k) \leqslant r$. Such k must exists by definition of r . By Markov's inequality then $|B(k,2r)| \geqslant n/2$. It follows that

$$\sum_{i,j \in B(k,2r)} d(i,j) \geqslant \sum_{i \in B(k,2r)} \frac{r}{4} |B(k,2r) \setminus B(i,r/4)| > \sum_{i \in B(k,2r)} \frac{r}{4} \cdot \frac{n}{4} \geqslant \frac{r \cdot n^2}{16} \tag{B.9}$$

which is a contradiction since μ is not well-spread. ■

Our third ingredient is the rounding algorithm (Algorithm 1) below.

Our last ingredient is the following classic theorem on vertex connectivity.

Theorem 20 (Menger, 1927) *A graph G contains at least k vertex-disjoint paths between two non-adjacent vertices $i, j \in V(G)$ if and only if every vertex cut that separates i from j has size at least k.*

Algorithm 1 Rounding

Input: Graph G, degree-4 pseudo-distribution μ consistent with $\mathcal{P}_{\bar{m}}(G)$.

Output: Set S.

if μ *is well-spread* **then**
  Let X, Y be the $(c'r/\log n)$-separated sets found by the algorithm of Lemma 18.
  Let $\Delta = c'r/\sqrt{\log n}$.
else
  Let $i^* = \operatorname{argmax}_i |B(i, r/4)|$ and $X = B(i^*, r/4)$.
  Let $\Delta = 1$.
end
for $i \in V$ *with* $d(i, X) < \Delta$ **do**
  Find the minimum vertex separator U_i between $\hat{X}$ and $V \setminus \hat{X}$.
  Let (S_i, U_i, R_i) be the resulting partition where $|S_i| \leqslant |R_i|$.
  if $|S_i| < c\bar{m}/4$ **then**
    Let S_i be the largest set between $\hat{X}$ and $V \setminus \hat{X}$.
end
Return the S_i minimizing the vertex expansion among all candidates.

Finally, we are ready to prove Theorem 4 by combining the technical results presented thus far.

Proof of Theorem 4 Given our input graph G, we may assume we know $\bar{m}$ since there are less than n possible values to try. Then we can compute in polynomial time a degree-4 pseudo-distribution μ of minimum cost consistent with $\mathcal{P}_{\bar{m}}(G)$.

It remains to analyze the rounding. To this end suppose step (a) of Algorithm 1 is replaced as follows: pick t be chosen uniformly at random from the interval $[0, \Delta)$, where Δ is defined at step (1), and let $\hat{X} := \{j \in V \mid d(j, X) \leqslant t\}$. Let S be the set returned at step (c). We will then relate the analysis under this modification with that of Algorithm 1.

Suppose first that μ is well-spread. By Lemma 18, we only need to show that Algorithm 1 finds a set S of size $m \leqslant |S| \leqslant n - m$ and with vertex expansion $O(1/\Delta)\phi_m(G)$. Since $\min\{|X|, |Y|\} \geqslant c \cdot \bar{m}$ for some constant $0 < c \leqslant 1$ and since the two sets are $\Delta = (c'r/\sqrt{\log n})$-separated w.r.t. the pseudo-metric d we must have $\min\{|\hat{X}|, |V \setminus \hat{X}|\} \geqslant c \cdot \bar{m}$. So let (S, U, R) be the partition obtained in step (b). If $|S| \leqslant c\bar{m}/4$ then $|U| \geqslant 3c\bar{m}/4$ and it follows that the largest between $\hat{X}$ and $V \setminus \hat{X}$ has expansion bounded from above by

$$\frac{4|U|}{c\bar{m}(n - c\bar{m})} \leqslant \frac{O(|U|)}{\bar{m}(n - \bar{m})}. \tag{B.10}$$

Else, we have

$$\phi(S) \leqslant \frac{|U_i|}{\frac{c\bar{m}}{4}(n - \frac{c\bar{m}}{4})} \leqslant \frac{O(|U|)}{\bar{m}(n - \bar{m})}. \tag{B.11}$$

To conclude the proof under the well-spreadness assumption, it remains to argue that $\mathbb{E}_t[|U|] \leqslant O(1) \sum_{i \in V} \widetilde{\mathbb{E}}\big[1 - x_i^2 - y_i^2\big]$. So order the vertices in increasing distance from X, breaking ties arbitrarily. Let $\mathcal{E}_i$ be the event that $d(i, X) \leqslant t \leqslant d(i+1, X)$ and let U_i be the vertex separator found by

the algorithm when $\mathcal{E}_i$ is verified. We show that for all $i \in V$, $\mathbb{P}(\mathcal{E}_i) \cdot |U_i| \leqslant 4 \sum_{i \in V} \widetilde{\mathbb{E}}\big[1 - x_i^2 - y_i^2\big]$ which implies the desired bound in expectation.

Suppose $\mathcal{E}_i$ is verified. Because U_i is a minimum vertex-separator, by Theorem 20 there must exist exactly $|U|$ ordered pairs (j, j') such that $j < j'$ and $j \in \hat{X}$ but $j' \notin \hat{X}$. Moreover, we must have $d(j, X) \leqslant d(i, X) \leqslant d(i+1, X) \leqslant d(j', X)$. Let P_i be the set of such pairs. We have

$$\begin{aligned}
\mathbb{P}(\mathcal{E}_i) \cdot |U_i| &= \sum_{(j,j') \in P_i} \mathbb{P}(\mathcal{E}_i) && \text{(B.12)} \\
&\leqslant \sum_{(j,j') \in P_i} \frac{1}{\Delta} |d(i, X) - d(i+1, X)| && \text{(B.13)} \\
&\leqslant \sum_{(j,j') \in P_i} \frac{1}{\Delta} \big|d(j, X) - d(j', X)\big| && \text{(B.14)} \\
&\leqslant \sum_{(j,j') \in P_i} \frac{d(j, j')}{\Delta} && \text{(B.15)} \\
&\leqslant \frac{4}{\Delta} \sum_{j \in V} \widetilde{\mathbb{E}}\big[1 - x_j^2 - y_j^2\big] \,, && \text{(B.16)}
\end{aligned}$$

where we used Theorem 16 in the last step. The claim for the original algorithm follows since there must be a choice of i for which $|U_i| \leqslant \mathbb{E}_t[|U|]$.

Consider now the case in which μ is not well-spread. A similar analysis as before shows that $\mathbb{E}_t[|U|] \leqslant 4 \sum_{j \in V} \widetilde{\mathbb{E}}\Big[1 - x_j^2 - y_j^2\Big]$. Hence if $|S| \geqslant c\bar{m}/4$ for some constant $c > 0$ then the argument is identical to the previous case.

Conversely, suppose the algorithm finds a partition (S, U, R) with $|S| \leqslant c\bar{m}/4$. By Lemma 19 we must have $|\hat{X}| \geqslant |X| \geqslant \Omega(n)$ and hence we only need to argue that $|V \setminus \hat{X}| \geqslant \Omega(\bar{m})$ since then the analysis may proceed as above. To this end notice that for any j , $\mathbb{P}\Big(j \notin \hat{X}\Big) = d(j, X)$. Therefore it holds

$$\mathbb{E}|\hat{X}|(n - |\hat{X}|) \geqslant |X| \sum_{j \in V} d(j, X) \,. \tag{B.17}$$

By triangle inequality

$$r \leqslant \frac{1}{n^2} \sum_{j,j' \in V} \big(d(j, X) + d(j', X)\big) + \frac{r}{2} = \frac{r}{2} + \frac{2}{n} \sum_{j \in V} d(j, X) \,. \tag{B.18}$$

This implies $|X| \sum_{j \in V} d(j, X) \geqslant \frac{r \cdot n \cdot |X|}{4} \geqslant \Omega(r \cdot n^2) = \Omega(\bar{m}(n - \bar{m}))$. To conclude the proof, it is enough to observe that by Markov's inequality $\mathbb{P}\big(|U| \leqslant z \cdot \mathbb{E}_t[|U|] \cdot \sqrt{\log n}\big) \geqslant 1 - 1/(z \cdot \sqrt{\log n})$ and by the Paley-Zygmund inequality $\mathbb{P}\Big(|V \setminus \hat{X}| \geqslant \Omega(\bar{m})\Big) \geqslant \bar{m}/n$ which means the two events have non-empty intersection for $z \geqslant n/(\bar{m} \cdot \sqrt{\log n})$ as desired. ■

B.2. Well-spread sets must be well-separated: proof of Lemma 18

Lemma 18 appears implicitly in Arora et al. (2009) and can be seen as a corollary of the following structure theorem. We direct the unfamiliar reader to Appendix C.

Lemma 21 (Arora et al., 2009) *Let d be a pseudo-metric of negative type over V satisfying:*

1. $\exists j \in V$ *such that* $\forall i \in V\,,\, d(i,j) \leqslant 2\,,$
2. $\sum_{ij \in V} d(i,j) \geqslant \Omega(n^2)\,.$

Then there exists $X, Y \subseteq V$ satisfying:

(i) $\min\{|X|\,,|Y|\} \geqslant \Omega(n)\,,$

(ii) X, Y *are* Δ*-separated for* $\Delta \geqslant O(1/\sqrt{\log n})\,.$

Moreover, there exists a Las Vegas polynomial time algorithm that finds X, Y.

We present next a proof of Lemma 18.

Proof of Lemma 18 For any degree-4 pseudo-distribution μ consistent with $\left\{x_i^2 = x_i\,, \forall i \in V\right\}$ the pseudo-metric induced by $\frac{d_\mu(i,j)}{r_\mu} = \frac{1}{r_\mu}\widetilde{\mathbb{E}}_\mu\left[(x_i - x_j)^2\right]$ is of negative type by Corollary 27. If μ is well-spread, then there exists $i \in V$ such that $\sum_{j,j' \in B(i,2r)} \frac{d(j,j')}{r} \geqslant n^2/16$ and $|B(i,2r)| \geqslant \Omega(n)\,.$ Hence the metric space $(B(i,2r), \frac{d}{r})$ satisfies the hypotheses of Lemma 21. ■

B.3. Poorly expanding sets must have large intersection: proof of Lemma 10

To prove Lemma 10 we use the following key consequence of Theorem 4.

Lemma 22 *Let G be an n-vertex graph, let $0 < m(n) \leqslant \frac{n}{2}$ and $0 < \eta(n) < 1$. Let $0 < \varepsilon'(n) < 1$ and $\alpha(n) \geqslant 1$ such that $\varepsilon'(n)\alpha(n) \leqslant 1$. Let $I \subset V(G)$ be an $(\varepsilon', \eta m, m)$-expanding set and let $S \subseteq [n]$ be a set satisfying*

(i) $\min\{|S|\,,|V \setminus S|\} \geqslant \Omega(m)$

(ii) $\phi_G(S) \leqslant \phi_m(G) \cdot \alpha$

Then $\max\left\{\frac{|S \cap I|}{|S|}\,,\frac{|(V \setminus S) \cap I|}{|V \setminus S|}\right\} \geqslant 1 - \max\left\{\sqrt{\varepsilon' \cdot \alpha}\,, O(\eta)\right\}.$

Proof Let $G^* = G[V \setminus I]$. Furthermore, let $c > 0$ be such that $\min\{|S|\,,|V \setminus S|\} \geqslant m/c$. By definition of I we have $\phi_G(I) < \phi_{\eta m}(G^*) \cdot \varepsilon'$. We consider two cases.

First, suppose $\min\left\{\frac{|S \setminus I|}{|S|}\,,\frac{|(V \setminus S) \setminus I|}{|V \setminus S|}\right\} < c\eta$. Then the claim immediately follows as

$$\max\left\{\frac{|S \cap I|}{|S|}\,,\frac{|(V \setminus S) \cap I|}{|V \setminus S|}\right\} = 1 - \min\left\{\frac{|S \setminus I|}{|S|}\,,\frac{|(V \setminus S) \setminus I|}{|V \setminus S|}\right\} > 1 - c\eta\,. \tag{B.19}$$

Second, suppose $\min\left\{\frac{|S\setminus I|}{|S|}, \frac{|(V\setminus S)\setminus I|}{|V\setminus S|}\right\} \geqslant c\eta$. In particular, it holds that

$$\eta m \leqslant c\eta|S| \leqslant |S\setminus I| = |V(G^*)| - |(V\setminus S)\setminus I| \leqslant |V(G^*)| - c\eta|V\setminus S| \leqslant |V(G^*)| - \eta m\,, \quad \text{(B.20)}$$

where we used the fact that $\min\{|S|, |V\setminus S|\} \geqslant m/c$. Thus $|V(G^*)| \geqslant 2\eta m$, implying that $\phi_{\eta m}(G^*)$ is well defined. Since $|S\setminus I| \geqslant c\eta|S|$ and $|S| \geqslant m/c$, we get that $\phi_{G^*}(S\setminus I) \geqslant \phi_{\eta m}(G^*)$. We can then show the following key inequality:

$$\frac{\phi_G(S)}{\phi_{G^*}(S\setminus I)} \leqslant \frac{\phi_m(G)\cdot\alpha}{\phi_{G^*}(S\setminus I)} \leqslant \frac{\phi_G(I)\cdot\alpha}{\phi_{G^*}(S\setminus I)} \leqslant \frac{\phi_{\eta m}(G^*)\cdot\alpha\cdot\varepsilon'}{\phi_{G^*}(S\setminus I)} \leqslant \alpha\cdot\varepsilon'\,, \quad \text{(B.21)}$$

where the second and third steps follows because I is $(\varepsilon', \eta m, m)$-expanding, and the last step follows from the fact that $\eta m \leqslant |S\setminus I| \leqslant |V(G^*)| - \eta m$. Then,

$$\phi_G(S) = \frac{|\partial_G(S)|}{|S|\cdot|V\setminus S|} \geqslant \frac{|\partial_{G^*}(S\setminus I)|}{|S|\cdot|V\setminus S|} = \frac{|\partial_{G^*}(S\setminus I)|}{|S\setminus I|\cdot|(V\setminus S)\setminus I|}\cdot\frac{|S\setminus I|\cdot|(V\setminus S)\setminus I|}{|S|\cdot|V\setminus S|} \quad \text{(B.22)}$$

$$= \phi_{G^*}(S\setminus I)\cdot\frac{|S\setminus I|\cdot|(V\setminus S)\setminus I|}{|S|\cdot|V\setminus S|}\,. \quad \text{(B.23)}$$

Applying Eq. (B.21) we obtain

$$\alpha\cdot\varepsilon' \geqslant \frac{\phi_G(S)}{\phi_{G^*}(S\setminus I)} \geqslant \frac{|S\setminus I|\cdot|(V\setminus S)\setminus I|}{|S|\cdot|V\setminus S|} \geqslant \min\left\{\frac{|S\setminus I|}{|S|}, \frac{|(V\setminus S)\setminus I|}{|V\setminus S|}\right\}^2, \quad \text{(B.24)}$$

which concludes the proof. ■

We are now ready to prove Lemma 10.

Proof of Lemma 10 Let S be the subset of $V(G)$ found by Algorithm 1. By Theorem 4, S satisfies *(ii)* and

$$|\partial_G(S)| \leqslant \alpha\phi_m(G)\cdot O(m\cdot(n-m)) \leqslant \alpha\phi_G(I)\cdot O(m\cdot(n-m)) \leqslant O(\alpha)|\partial_G(I)|, \quad \text{(B.25)}$$

where we used the assumption that $m \leqslant |I| \leqslant n/2$. So S also satisfies *(i)*. Now, by Lemma 22 either $|S\setminus I| \leqslant O(\eta)|S|$ or $|V\setminus(S\cup I)| \leqslant O(\eta)|V\setminus S|$. We may assume now that $C\cdot\eta < 1/10$ for a sufficiently large universal constant, since otherwise *(iii)* is trivially satisfied and the result follows by simply returning S. If $\min\{|S|, |V\setminus S|\} \leqslant 10\log p$ then we can deterministically check which side of the partition satisfies *(iii)*. Otherwise, notice that because $|I| \leqslant n/2$, we cannot have $\min\left\{\frac{|S\cap I|}{|S|}, \frac{|(V\setminus S)\cap I|}{|V\setminus S|}\right\} \geqslant \frac{1-C\cdot\eta}{3}$. Hence it suffices to distinguish between these two cases. For any subset $S^* \subseteq S$ chosen uniformly at random from S, we have by standard concentration bounds, for any $p \leqslant 1/2$,

$$\mathbb{P}\left(\left||S^*\cap I| - |S^*|\frac{|S\cap I|}{|S|}\right| \geqslant \sqrt{3|S^*|\frac{|S\cap I|}{|S|}\log(1/2p)}\right) \leqslant p\,. \quad \text{(B.26)}$$

Hence picking some random subset of S of size $10\log p$ we have that with probability $1 - p/2$ we can correctly decide whether $|S\cap I| \geqslant (1 - C\cdot\eta)|S|/2$. We may do the same for $V\setminus S$ and we can then correctly find the side maximizing $\left\{\frac{|S\cap I|}{|S|}, \frac{|(V\setminus S)\cap I|}{|V\setminus S|}\right\}$ with probability $1 - p$.

If such set is S, the proof follows by simply returning S. Otherwise, let $S' = S \cup \{v \in \partial_G(S) \,|\, v \notin I\}$. Notice that we can construct S' in linear time by asking $|\partial_G(S)| \leqslant \alpha|\partial_G(I)|$ queries to $\mathcal{O}_G$. We claim that $V \setminus S'$ now satisfies *(i),(ii),(iii)*. Indeed, by construction

$$|\partial_G(V \setminus S')| \leqslant |\partial_G(S)| + |\partial_G(I)| \leqslant (\alpha + 1)|\partial_G(I)| \tag{B.27}$$

and

$$|V \setminus S'| \geqslant |(V \setminus S) \cap I| \geqslant (1 - O(\eta))|V \setminus S| \geqslant \Omega(m). \tag{B.28}$$

Finally, *(iii)* follows since $(V \setminus S') \cap I = (V \setminus S) \cap I$. ■

Appendix C. Sum-of-squares background

We present here necessary background about the sum-of-squares framework. See Fleming et al. (2019) for proofs and more details.

Let $x = (x_1, x_2, \ldots, x_n)$ be a tuple of n indeterminates and let $\mathbb{R}[x]$ be the set of polynomials with real coefficients and indeterminates $x_1, \ldots, x_n$. In a *polynomial feasibility problem*, we are given a system of polynomial inequalities $\mathcal{A} = \{f_1 \geqslant 0, \ldots, f_m \geqslant 0\}$, and we would like to know if there exists a point $x \in \mathbb{R}^n$ satisfying $f_i(x) \geqslant 0$ for all $i \in [m]$. This task is easily seen to be NP-hard.

Given a polynomial system $\mathcal{A}$, the *sum-of-squares (sos) algorithm* computes a *pseudo-distribution* of solutions to $\mathcal{A}$ if one exists. Pseudo-distributions are generalizations of probability distributions, therefore the sos algorithm solves a relaxed version of the feasibility problem. The search for a pseudo-distribution can be formulated as a semidefinite program (SDP).

There is strong duality between *pseudo-distributions* and *sum-of-squares proofs*: the sos algorithm will either find a pseudo-distribution satisfying $\mathcal{A}$, or a refutation of $\mathcal{A}$ inside the sum-of-squares proof system. When using sos for algorithm design as we do here, we work in the former case and our goal is to design a rounding algorithm that transforms a pseudo-distribution into an actual point x that satisfies or nearly satisfies $\mathcal{A}$.

The side of the sum-of-squares algorithm which computes a pseudo-distribution is summarized into the following theorem (we will not need the side that computes a sum-of-squares refutation). The full definitions of these objects will be presented momentarily.

Theorem 23 *Fix a parameter $\ell \in \mathbb{N}$. There exists an $(n+m)^{O(\ell)}$-time algorithm that, given an explicitly bounded and satisfiable polynomial system $\mathcal{A} = \{f_1 \geqslant 0, \ldots, f_m \geqslant 0\}$ in n variables with bit complexity $(n+m)^{O(1)}$, outputs a degree-ℓ pseudo-distribution that satisfies $\mathcal{A}$ approximately.*

Pseudo-distributions. We can represent a discrete (i.e., finitely supported) probability distribution over $\mathbb{R}^n$ by its probability mass function $\mu \colon \mathbb{R}^n \to \mathbb{R}$ such that $\mu \geqslant 0$ and $\sum_{x \in \mathrm{supp}(\mu)} \mu(x) = 1$. A pseudo-distribution relaxes the constraint $\mu \geqslant 0$ and only requires that μ passes certain low-degree non-negativity tests.

Concretely, a *degree-ℓ pseudo-distribution* is a finitely-supported function $\mu : \mathbb{R}^n \to \mathbb{R}$ such that $\sum_{x \in \mathrm{supp}(\mu)} \mu(x) = 1$ and $\sum_{x \in \mathrm{supp}(\mu)} \mu(x) f(x)^2 \geqslant 0$ for every polynomial f of degree at most $\ell/2$. A straightforward polynomial interpolation argument shows that every degree-∞ pseudo-distribution satisfies $\mu \geqslant 0$ and is thus an actual probability distribution.

A pseudo-distribution μ can be equivalently represented through its *pseudo-expectation operator* $\widetilde{\mathbb{E}}_\mu$. For a function f on $\mathbb{R}^n$ we define the pseudo-expectation $\widetilde{\mathbb{E}}_\mu f(x)$ as

$$\widetilde{\mathbb{E}}_\mu f(x) = \sum_{x \in \mathrm{supp}(\mu)} \mu(x) f(x) \ . \tag{C.1}$$

We are interested in pseudo-distributions which satisfy a given system of polynomials $\mathcal{A}$.

Definition 24 (Satisfying constraints) *Let μ be a degree-ℓ pseudo-distribution over $\mathbb{R}^n$. Let $\mathcal{A} = \{f_1 \geqslant 0, f_2 \geqslant 0, \dots, f_m \geqslant 0\}$ be a system of polynomial inequalities. We say that μ* is consistent with $\mathcal{A}$ at level r, *denoted $\mu \models_r \mathcal{A}$, if for every $S \subseteq [m]$ and every polynomial h with $2 \deg h + \sum_{i \in S} \max\{\deg f_i, r\} \leqslant \ell$,*

$$\widetilde{\mathbb{E}}_\mu h^2 \cdot \prod_{i \in S} f_i \geqslant 0 \, .$$

We say μ satisfies $\mathcal{A}$ and write $\mu \models \mathcal{A}$ if the case $r = 0$ holds.

We remark that $\mu \models \{1 \geqslant 0\}$ is equivalent to μ being a valid pseudo-distribution, and if μ is an actual (discrete) probability distribution, then we have $\mu \models \mathcal{A}$ if and only if μ is supported on solutions to the constraints $\mathcal{A}$.

The pseudo-expectations of all polynomials in the variables x with degree at most ℓ can be packaged into the list of *pseudo-moments* $\widetilde{\mathbb{E}}_\mu x^S$ for all monomials x^S, $|S| \leqslant \ell$. Since we will be entirely concerned with polynomials up to degree ℓ, as in Definition 24, we can treat a degree-ℓ pseudo-distribution as being equivalently specified by the list of pseudo-moments up to degree ℓ. Thus we will view the output of the degree-ℓ sos algorithm as being the list of all pseudo-moments up to degree ℓ which has size $O(n^\ell)$.

To design an algorithm based on sos, our task is to utilize the pseudo-moments in order to find a solution point x. The sos framework extends linear programming and semidefinite programming, which conceptually use only the degree-1 or degree-2 moments respectively. Taking sos to higher degree enforces additional constraints on all of the moments, coming from higher-degree sum-of-squares proofs as we will see next.

Sum-of-squares proofs. We say that a polynomial $p \in \mathbb{R}[x]$ is a *sum-of-squares (sos)* if there are polynomials $q_1, \dots, q_r \in \mathbb{R}[x]$ such that $p = q_1^2 + \cdots + q_r^2$. Let $f_1, f_2, \dots, f_m, g \in \mathbb{R}[x]$. A *sum-of-squares proof* that the constraints $\{f_1 \geqslant 0, \dots, f_m \geqslant 0\}$ imply the constraint $\{g \geqslant 0\}$ consists of sum-of-squares polynomials $(p_S)_{S \subseteq [m]}$ such that

$$g = \sum_{S \subseteq [m]} p_S \cdot \Pi_{i \in S} f_i \, . \tag{C.2}$$

We say that this proof has *degree* ℓ if for every set $S \subseteq [m]$, the polynomial $p_S \Pi_{i \in S} f_i$ has degree at most ℓ. When a set of inequalities $\mathcal{A}$ implies $\{g \geqslant 0\}$ with a degree ℓ SoS proof, we write:

$$\mathcal{A} \vdash_{\ell} \{g \geqslant 0\} . \tag{C.3}$$

A sum-of-squares *refutation* of $\mathcal{A}$ is a proof $\mathcal{A} \vdash_{\ell} \{-1 \geqslant 0\}$.

Duality. Degree-ℓ pseudo-distributions and degree-ℓ sum-of-squares proofs exhibit strong duality. In proof theoretic terms, degree-ℓ sum-of-squares proofs are sound and complete when degree-ℓ pseudo-distributions are taken as models.

Soundness, or weak duality, states that every sum-of-squares proof enforces a constraint on every valid pseudo-distribution.

Fact 25 (Weak duality/soundness) *If $\mu \models_{r} \mathcal{A}$ for a degree-ℓ pseudo-distribution μ and there exists a sum-of-squares proof $\mathcal{A} \vdash_{r'} \mathcal{B}$, then $\mu \models_{r \cdot r' + r'} \mathcal{B}$.*

There is a degree-4 proof of the ℓ_2^2 triangle inequality, which implies that every degree-4 pseudo-distribution satisfies the ℓ_2^2 triangle inequality.

Lemma 26 (ℓ_2^2 triangle inequality) *It holds that*

$$\{x_i^2 = x_i\}_{i \in [n]} \vdash_{4} \left\{(x_i - x_j)^2 \leqslant (x_i - x_k)^2 + (x_k - x_j)^2\right\}_{i,j,k \in [n]} .$$

Proof

$$\begin{aligned}
(x_i - x_k)^2 + (x_j - x_k)^2 - (x_i - x_j)^2 &= 2x_k^2 + 2x_i x_j - 2x_j x_k - 2x_i x_k && \text{(C.4)} \\
&= 2(x_k - x_i)(x_k - x_j) && \text{(C.5)} \\
&= 2x_k + 2x_i x_j - 2x_j x_k - 2x_i x_k + 2(x_k^2 - x_k) . && \text{(C.6)}
\end{aligned}$$

One can verify by truth table that $(x_k - x_i)(x_k - x_j)$ takes values in $\{0, 1\}$ for Boolean $x_i, x_j, x_k \in \{0, 1\}$. Therefore its multilinear interpolation $f(x) := x_k + x_i x_j - x_j x_k - x_i x_k$ is the same as that of its square i.e., $f(x) = f(x)^2 + p_i \cdot (x_i^2 - x_i) + p_j \cdot (x_j^2 - x_j) + p_k \cdot (x_k^2 - x_k)$ for some polynomials p_i, p_j, p_k with degree $\leqslant 2$. This is a degree-4 sos proof of $f(x) \geqslant 0$. ■

Corollary 27 *For any degree-4 pseudo-expectation $\widetilde{\mathbb{E}}_{\mu}$ satisfying the constraints $\{x_i^2 = x_i\}_{i \in [n]}$, for all $i, j, k \in [n]$,*

$$\widetilde{\mathbb{E}}_{\mu}(x_i - x_j)^2 \leqslant \widetilde{\mathbb{E}}_{\mu}(x_i - x_k)^2 + \widetilde{\mathbb{E}}_{\mu}(x_j - x_k)^2 .$$

Although we will not need it in our analysis, strong duality a.k.a (refutational) completeness conversely shows that for a given set of axioms, there always exists either a degree-ℓ pseudo-distribution or a degree-ℓ sos refutation.

Fact 28 (Strong duality/refutational completeness) *Suppose $\mathcal{A}$ is a collection of polynomial constraints such that $\mathcal{A} \vdash_{\ell - r} \{\sum_{i=1}^{n} x_i^2 \leqslant B\}$ for some finite B. If there is no degree-ℓ pseudo-distribution μ such that $\mu \models_{r} \mathcal{A}$, then there is a sum-of-squares refutation $\mathcal{A} \vdash_{\ell - r} \{-1 \geqslant 0\}$.*

Negative type metrics. Let (V, d) be a finite pseudo-metric space. (V, d) is of negative type if and only if $(V, \sqrt{d})$ is an Euclidean pseudo-metric. More precisely, for any negative-type metric, there is a map $\psi : V \to \mathbb{R}^n$ such that $\|\psi(i) - \psi(j)\|^2 = d(i, j)$, for every $i, j \in V$. Let μ be degree-4 pseudo-distribution consistent with $\{x_i^2 = x_i \,, \forall i \in V\}$ and consider the function $d : V \times V \to \mathbb{R}$ given by $d(i, j) = \widetilde{\mathbb{E}}[(x_i - x_j)^2]$ for all $i, j \in V$. By Corollary 27 (V, d) is a pseudo-metric. Furthermore, the mapping $\psi : V \to \mathbb{R}$ can be constructed taking the Gram vectors of the matrix $\widetilde{\mathbb{E}}_\mu [xx^\top]$.

Implementation of sos. The sum-of-squares algorithm can be implemented as a semidefinite program (SDP) which can then be solved using, for example, the ellipsoid method. Associated with a degree-ℓ pseudo-distribution μ is the *moment tensor* which is the tensor $\widetilde{\mathbb{E}}_\mu (1, x_1, x_2, \ldots, x_n)^{\otimes \ell}$. When ℓ is even, this tensor can be flattened into the *moment matrix*, which has rows and columns indexed by multisets of $[n]$ with size at most $\ell/2$ and whose (I, J) entry is $\widetilde{\mathbb{E}}_\mu x^I x^J$. Moment matrices can now be characterized as positive semidefinite matrices with simple symmetry constraints from flattening.

Fact 29 *A matrix Λ with rows and columns indexed by multisets of $[n]$ with size at most ℓ is a moment matrix of a degree-2ℓ pseudo-distribution if and only if:*

(i) $\Lambda \succeq 0$

(ii) $\Lambda_{I,J} = \Lambda_{I',J'}$ *whenever* $I \cup J = I' \cup J'$ *as multisets*

(iii) $\Lambda_{\{\},\{\}} = 1$

The above characterization of pseudo-distributions in terms of the cone of positive semidefinite matrices is a formulation of the sos algorithm as an SDP.

We can deduce Theorem 23 from the general theory of convex optimization Grötschel et al. (2012). The above fact leads to an $n^{O(\ell)}$-time weak separation oracle for the convex set of all moment tensors of degree-ℓ pseudo-distributions over $\mathbb{R}^n$. By the results of Grötschel et al. (1981), we can optimize over the set of pseudo-distributions in time $n^{O(\ell)}$, assuming numerical conditions.

The first numerical condition is that the bit complexity of the input to the sos algorithm is polynomial. The second numerical condition is that we assume an upper bound on the norm of feasible solutions. This is guaranteed if the input polynomial system $\mathcal{A}$ is *explicitly bounded*, meaning that it contains a constraint of the form $\|x\|^2 \leqslant M$ for some $M \geqslant 0$ with polynomial bit length, or if $\mathcal{A} \vdash_\ell \{\|x\|^2 \leqslant M\}$. For example, Boolean constraints satisfy this since

$$\{x_i^2 = x_i\}_{i \in [n]} \vdash_2 \{\|x\|^2 \leqslant n\} \,. \tag{C.7}$$

Due to finite numerical precision, the output of the sos algorithm can only be computed approximately, not exactly. For a pseudo-distribution μ, we say that $\mu \models_r \mathcal{A}$ holds *approximately* if the inequalities in Definition 24 are satisfied up to an error of $2^{-n^\ell} \cdot \|h\| \cdot \prod_{i \in S} \|f_i\|$, where $\|\cdot\|$ denotes the Euclidean norm of the coefficients of a polynomial in the monomial basis.[6] In our analysis,

6. The choice of norm is not important here because the factor 2^{-n^ℓ} swamps the effect of choosing another norm.

the approximation error is so minuscule that it can be ignored and we will simply assume that the pseudo-distribution μ computed by the sos algorithm satisfies $\mathcal{A}$ without error.