Erasure-Resilient Sublinear-Time Graph Algorithms

Amit Levi
David R. Cheriton School of Computer Science, University of Waterloo, Canada
amit.levi@uwaterloo.ca

Ramesh Krishnan S. Pallavoor
Department of Computer Science, Boston University, MA, USA
rameshkp@bu.edu

Sofya Raskhodnikova
Department of Computer Science, Boston University, MA, USA
sofya@bu.edu

Nithin Varma
Department of Computer Science, University of Haifa, Israel
nvarma@bu.edu

Abstract

We investigate sublinear-time algorithms that take partially erased graphs represented by adjacency lists as input. Our algorithms make degree and neighbor queries to the input graph and work with a specified fraction of adversarial erasures in adjacency entries. We focus on two computational tasks: testing if a graph is connected or ε-far from connected and estimating the average degree. For testing connectedness, we discover a threshold phenomenon: when the fraction of erasures is less than ε, this property can be tested efficiently (in time independent of the size of the graph); when the fraction of erasures is at least ε, then a number of queries linear in the size of the graph representation is required. Our erasure-resilient algorithm (for the special case with no erasures) is an improvement over the previously known algorithm for connectedness in the standard property testing model and has optimal dependence on the proximity parameter ε. For estimating the average degree, our results provide an "interpolation" between the query complexity for this computational task in the model with no erasures in two different settings: with only degree queries, investigated by Feige (SIAM J. Comput. '06), and with degree queries and neighbor queries, investigated by Goldreich and Ron (Random Struct. Algorithms '08) and Eden et al. (ICALP '17). We conclude with a discussion of our model and open questions raised by our work.

2012 ACM Subject Classification Theory of computation → Streaming, sublinear and near linear time algorithms; Mathematics of computing → Approximation algorithms

Keywords and phrases Graph property testing, Computing with incomplete information, Approximating graph parameters

Digital Object Identifier 10.4230/LIPIcs.ITCS.2021.80

Related Version The full version of this article is available at https://arxiv.org/abs/2011.14291 [20].

Funding *Amit Levi*: Part of this work was done while the author was visiting Boston University.
Ramesh Krishnan S. Pallavoor: The work of this author was partially supported by NSF award CCF-1909612 and was done in part while the author was visiting the Simons Institute for the Theory of Computing.
Sofya Raskhodnikova: The work of this author was partially supported by NSF award CCF-1909612 and was done in part while the author was visiting the Simons Institute for the Theory of Computing.
Nithin Varma: The work of this author was partially supported by ISF grant 497/17 and Israel PBC Fellowship for Outstanding Postdoctoral Researchers from India and China. This work was done in part while the author was a student at Boston University.

Acknowledgements We thank Talya Eden for useful discussions that led to simplification of analysis in Section 3.1.

12th Innovations in Theoretical Computer Science Conference (ITCS 2021).
Editor: James R. Lee; Article No. 80; pp. 80:1–80:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

The goal of this work is to model and investigate sublinear-time algorithms that run on graphs with incomplete information. Typically, sublinear-time models assume that algorithms have query or sample access to an input graph. However, this assumption does not accurately reflect reality in some situations. Consider, for example, the case of a social network where vertices represent individuals and edges represent friendships. Individuals might want to hide their friendship relations for privacy reasons. When input graphs are represented by their adjacency lists, such missing information can be modeled as *erased* entries in the lists. In this work, we initiate an investigation of sublinear-time algorithms whose inputs are graphs represented by the adjacency lists with some of the entries adversarially erased.

In our erasure-resilient model of sublinear-time graph algorithms, an algorithm gets a parameter $\alpha \in [0,1]$ and query access to the adjacency lists of a graph with at most an α fraction of the entries in the adjacency lists erased. We call such a graph *α-erased* or, when α is clear from the context, *partially erased*. Algorithms access partially erased graphs via degree and neighbor queries. The answer to a degree query v is the degree of the vertex v. A neighbor query is of the form (v, i), and the answer is the i^{th} entry in the adjacency list of v. If the i^{th} entry is erased[1], the answer is a special symbol $\perp$. A *completion* of a partially erased graph G is a valid graph represented by adjacency lists (with no erasures) that coincide with the adjacency lists of G on all nonerased entries. We formulate our computational tasks in terms of valid completions of partially erased input graphs and analyze the performance of our erasure-resilient algorithms in the *worst case* over all α-erased graphs. We investigate representative problems from two fundamental classes of computational tasks in our model: graph property testing and estimating a graph parameter.

In the context of graph property testing [15], we study the problem of testing whether a partially erased graph is connected. Our model is a generalization of the *general graph model* of Parnas and Ron [23] (which is in turn a generalization of the *bounded degree model* of Goldreich and Ron [16]) to the setting with erasures. A partially erased graph G has property $\mathcal{P}$ (in our case, is connected) if there exists a completion of G that has the property. For $\varepsilon \in (0,1)$, such a graph with m edges (more precisely, $2m$ entries in its adjacency lists) is ε-far from $\mathcal{P}$ (in our case, from being connected) if every completion of G is different in at least εm edges from every graph with the property. The goal of a testing algorithms is to distinguish, with high probability, α-erased graphs that have the property from those that are ε-far. For testing connectedness in our erasure-resilient model, we discover a threshold phenomenon: when the fraction of erasures is less than ε, this property can be tested efficiently (in time independent of the size of the graph); when the fraction of erasures is at least ε, then a number of queries linear in the size of the graph is required to test connectedness. Additionally, when there are no erasures, our tester has better query complexity than the best previously known standard tester for connectedness [23, 5], also mentioned in the book on property testing by Goldreich [14]. Our tester has optimal dependence on ε, as evidenced by a recent lower bound in [21] for this fundamental property.

Next, we study erasure-resilient algorithms for estimating the average degree of a graph. The problem of estimating the average degree of a graph, in the case with no erasures, was studied by Feige [13], Goldreich and Ron [17], and Eden et al. [9, 10]. Feige designed an algorithm that, for all $\varepsilon > 0$, makes $O(\sqrt{n}/\varepsilon)$ degree queries to an n-node graph and outputs, with high probability, an estimate that is within a factor of $2 + \varepsilon$ of the average

[1] One can consider a more general model where the degrees of some vertices can also be erased. Our algorithms continue to work in this model, since one can determine the degree of a vertex using $O(\log n)$ neighbor queries (irrespective of whether these queries are made to erased adjacency entries).

degree. He also showed that to get a 2-approximation, one needs $\Omega(n)$ degree queries. Goldreich and Ron proved that if an algorithm can make uniformly random neighbor queries (that is, obtain a uniformly random neighbor of a specified vertex) then, for all $\varepsilon > 0$, the average degree can be estimated to within a factor of $1 + \varepsilon$ using $O(\sqrt{n} \cdot \text{poly}(\log n, 1/\varepsilon))$ queries. Eden et al. proved a tighter bound of $O(\sqrt{n} \cdot \log \log n \cdot \text{poly}(1/\varepsilon))$ on the query complexity of this problem and provided a simpler analysis. We describe an algorithm that estimates the average degree of α-erased graphs to within a factor of $1 + \min(2\alpha, 1) + \varepsilon$ using $O(\sqrt{n} \cdot \log \log n \cdot \text{poly}(1/\varepsilon))$ queries. Our result can be thought of as an interpolation between the results in [13] and [17, 9, 10]. In particular, when there are no erasures, that is, when $\alpha = 0$, we get a $(1 + \varepsilon)$-approximation; when all adjacency entries are erased, and only the degree queries are useful, that is, when $\alpha = 1$, we obtain a $(2 + \varepsilon)$-approximation. We also show that our result cannot be improved significantly: to get a $(1 + \alpha)$-approximation, $\Omega(n)$ queries are necessary.

Discussion of our model

For the case of graph property testing, our model is an adaptation of the erasure-resilient model for testing properties of functions by Dixit et al. [7]. Dixit et al. designed erasure-resilient testers for many properties of functions, including monotonicity, the Lipschitz property, and convexity. The conceptual difference between the two models is that the adjacency lists representation of a graph cannot be viewed as a function. (This is not the case for the adjacency matrix representation.) For a function, erased entries can be filled in arbitrarily and, as a result, they never contribute to the distance to the property. For the adjacency lists representation, this is not the case: erasures have to be filled so that the resulting completion is a valid graph. The restrictions on how they can be filled may result in some contribution to the distance coming from the erased entries[2]. For example, consider the property of bipartiteness. Let B be a complete balanced bipartite graph $(U, V; E)$, and let B' be obtained from B by adding an erased entry to the adjacency list of every vertex in U. Then, in every completion of B', all formerly erased entries have to be changed to make the graph bipartite.

Furthermore, Dixit et al. [7] gave results only on property testing in the erasure-resilient model. We go beyond property testing in our exploration of erasure-resilient algorithms by considering more general computational tasks.

Finally, our model opens up many new research directions, some of which are discussed in Section 4.

1.1 The Model

We consider simple undirected graphs $G = (V, E)$ represented by adjacency lists, where some entries in the adjacency lists could be adversarially erased (these entries are denoted by $\perp$).

▶ **Definition 1.1** (α-erased graph; completion). *Let $\alpha \in [0, 1]$ be a parameter. An* α-erased graph *on a vertex set V is a concatenation of the adjacency lists of a simple undirected graph (V, E) with at most an α fraction of all entries (that is, at most $2\alpha|E|$ entries) in the lists erased. A* completion *of an α-erased graph G is the adjacency lists representation of a simple undirected graph G' that coincides with G on all nonerased entries.*

[2] Because of this, we make an adjustment to the model of Dixit et al. [7]: we measure the distance to the property as a fraction of the completion representation that needs to be changed, as opposed to the fraction of the nonerased representation that needs to be changed.

By definition, every partially erased graph has a completion, because it was obtained by erasing entries in a valid graph.

Given a partially erased graph G over a vertex set V, we use n to denote $|V|$ and m to denote the number of edges in any completion of G, that is, half the sum of lengths of the adjacency lists of all the vertices in G. The average degree, that is, $2m/n$, is denoted by $\overline{d}$. For $u \in V$, we use $\mathsf{Adj}(u)$ to denote the adjacency list of u. The degree u, denoted $\deg(u)$, is the length of $\mathsf{Adj}(u)$.

▶ **Definition 1.2** (Nonerased and half-erased edges). *Let G be a partially erased graph over a vertex set V. For vertices $u, v \in V$, the set $\{u, v\}$ is a* nonerased edge *in G if u is present in* $\mathsf{Adj}(v)$ *and vice versa. The set $\{u, v\}$ is a* half-erased edge *if u is in* $\mathsf{Adj}(v)$ *but v is not in* $\mathsf{Adj}(u)$*, or vice versa.*

Our algorithms make two types of queries: *degree queries* and *neighbor queries*. A degree query specifies a vertex v, and the answer is $\deg(v)$. A neighbor query specifies (v, i), and the answer is the i^{th} entry in $\mathsf{Adj}(v)$.

▶ **Definition 1.3** (Distance to a property; erasure-resilient property tester). *Let $\alpha \in [0, 1]$, $\varepsilon \in (0, 1)$ be parameters. An α-erased graph G satisfies a property $\mathcal{P}$ if there exists a completion of G that satisfies $\mathcal{P}$. An α-erased graph G is ε-far from a property $\mathcal{P}$ if every completion G' of G is different in at least εm edges from every graph that satisfies $\mathcal{P}$.*

An α-erasure-resilient ε-tester for a property $\mathcal{P}$ gets parameters $\alpha \in [0, 1], \varepsilon \in (0, 1)$ and query access to an α-erased graph G. The tester accepts, with probability at least $2/3$, if G satisfies $\mathcal{P}$. The tester rejects, with probability at least $2/3$, if G is ε-far from $\mathcal{P}$.

1.2 Our Results

In this section, we state our main results for the erasure-resilient model of sublinear-time algorithms.

1.2.1 Testing Connectedness

The problem of testing connectedness in the *general graph model* (that we further generalize to the erasure-resilient setting) was studied by Parnas and Ron [23]. The results on this fundamental problem are described in Section 10.2.1 in [14]. The best tester for this problem to date, due to [5], had query complexity $O\big(\frac{1}{(\varepsilon \overline{d})^2}\big)$.

We give two erasure-resilient testers for connectedness: one for small values of α and another for intermediate values of α. Both testers work for all[3] values of the proximity parameter, ε. We first give a tester that works for all $\alpha < \varepsilon/2$. (This tester is presented in Section 2.1.)

▶ **Theorem 1.4.** *There exists an α-erasure-resilient ε-tester for connectedness of graphs with the average degree $\overline{d}$ that has $O\big(\min\big\{\frac{1}{((\varepsilon-2\alpha)\overline{d})^2}, \frac{1}{\varepsilon-2\alpha} \log \frac{1}{(\varepsilon-2\alpha)\overline{d}}\big\}\big)$ query and time complexity and works for every $\varepsilon \in (0, 2/\overline{d})$ and $\alpha \in [0, \varepsilon/2)$. The tester has 1-sided error. When the average degree $\overline{d}$ of the input graph is unknown, α-erasure-resilient ε-testing of connectedness (with 1-sided error) has query and time complexity $O(\frac{1}{\varepsilon-2\alpha} \log \frac{1}{\varepsilon-2\alpha})$.*

[3] For $\varepsilon \geq 2/\overline{d}$, we have $\varepsilon m \geq n$. Then testing connectedness is trivial, since every graph can be made connected by adding at most $n - 1$ edges.

Importantly, when the input adjacency lists have no erasures (i.e., when $\alpha = 0$), our tester has better query complexity than the previously known best (standard) tester for connectedness, which was due to [5]. We present a standalone algorithm for this important special case in the full version of this article [20]. By substituting $\alpha = 0$ in Theorem 1.4, we get $O\big(\min\big\{\frac{1}{(\varepsilon\overline{d})^2}, \frac{1}{\varepsilon}\log\frac{1}{\varepsilon\overline{d}}\big\}\big)$ query complexity for the case when $\overline{d}$ is known and $O(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ query complexity when $\overline{d}$ is unknown. For the case with no erasures, the improvement in query complexity as a function of ε is from $O(\frac{1}{\varepsilon^2})$ to $O(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$. The latter is optimal, as evidenced by an $\Omega(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ lower bound for testing connectedness of graphs of degree 2 in [21]. We note that Berman et al. [5] already proved that testing connectedness of graphs (with no erasures) in the bounded degree graph model of [16] has query complexity $O(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon D})$ where D denotes the degree bound. Our result shows that the same query complexity (with D replaced by $\overline{d}$) is attainable in the general graph model.

Our first tester looks for small connected components that do not have any erasures. When $\alpha \in [\varepsilon/2, \varepsilon)$, some α-erased graphs that are ε-far from connected may not have any connected component that is free of erasures. Consequently, our first tester fails to reject such graphs. We give a different algorithm (presented in Section 2.2) which works by looking for a subset of vertices that has at most one erasure and gets completed to a unique connected component in every completion of the partially erased graph. (In the beginning of Section 2.2, we give an explanation, illustrated by Figure 1, of why two erasures in a witness may render it not detectable from a local view obtained by a sublinear algorithm.)

▶ **Theorem 1.5.** *There exists an α-erasure-resilient ε-tester for connectedness of graphs with the average degree $\overline{d}$ that has $O\big(\frac{1}{(\varepsilon-\alpha)^2\cdot\overline{d}}\cdot\min\big\{\frac{1}{(\varepsilon-\alpha)\cdot\overline{d}^2}, 1\big\}\big)$ query and time complexity and works for every $\varepsilon \in (0, 2/\overline{d})$ and $\alpha \in [0, \varepsilon)$. The tester has 1-sided error.*

Finally, we show that when $\alpha \geq \varepsilon$, the task of α-erasure-resilient ε-testing of connectedness requires examining a linear portion of the graph representation. That is, we discover a phase transition in the complexity of this problem when the fraction of erasures α reaches the proximity parameter ε.

▶ **Theorem 1.6.** *For all $\varepsilon \in (0, 1/7]$, every ε-erasure-resilient ε-tester for connectedness that makes only degree and neighbor queries requires a number of queries linear in the size of the graph representation.*

To prove this theorem, we construct (in Section 2.3) a family of partially erased graphs for which it is hard to distinguish connected graphs from graphs that are far from connected. The average degree of the graphs in our constructions is constant. So, the lower bound for this graph family is $\Omega(n) = \Omega(m)$.

1.2.2 Estimating the Average Degree

In Section 3.1, we give an erasure-resilient algorithm for estimating the average degree by generalizing the algorithm of Eden et al. [9, 10] to work for the case with erasures.

▶ **Theorem 1.7.** *Let $\alpha \in [0, 1]$ and $\varepsilon \in (0, 1/2)$. There exists an algorithm that makes $O(\sqrt{n}\cdot\log\log n\cdot\mathrm{poly}(1/\varepsilon))$ degree queries and uniformly random neighbor queries to an α-erased input graph of average degree $\overline{d} \geq 1$ and outputs, with probability at least $2/3$, an estimate $\widetilde{d}$ satisfying $(1-\varepsilon)\cdot\overline{d} < \widetilde{d} < (1 + 2\min(\alpha, \frac{1}{2}) + \varepsilon)\cdot\overline{d}$. The running time of the algorithm is the same as its query complexity.*

For graphs with no erasures, a good estimate of the number of edges gives a good estimate of the average degree. Feige's algorithm [13] (that has access only to degree queries) counts some edges twice and gets an estimate of the average degree that is within a factor of $2 + \varepsilon$. Goldreich and Ron [17] and Eden et al. [9, 10] avoid the issue of double-counting by ranking vertices according to their degrees and estimating, within a factor of $1 + \varepsilon$, the number of edges going from lower-ranked to higher-ranked vertices. These algorithms use degree queries and uniformly random neighbor queries. Having erasures in the adjacency lists is, in a rough sense, equivalent to not having access to *some* of the neighbor queries. This results in the additional 2α error term in the approximation guarantee. Consequently, when the fraction of erasures approaches 1/2, all the "relevant" entries in the adjacency lists of the input graph could be erased, and we enter the regime of having access only to degree queries.

In Section 3.2, we show that, for any fraction $\alpha \in (0, 1]$, estimating the average degree of an α-erased graph to within a factor of $(1 + \alpha)$ requires $\Omega(n)$ queries. In other words, the approximation ratio of our erasure-resilient algorithm for estimating the average degree cannot be improved significantly.

▶ **Theorem 1.8.** *Let $\alpha \in (0, 1]$ be rational. For all $\gamma < \alpha$, at least $\Omega(n)$ queries are necessary for every algorithm that makes degree and neighbor queries to an α-erased graph with the average degree $\overline{d}$ and outputs, with probability at least 2/3, an estimate $\widetilde{d} \in \left[\overline{d}, (1 + \gamma)\overline{d}\right]$.*

1.3 Research Directions and Further Observations

There are numerous research questions that arise from our work. In Section 4, we discuss some of them and also give additional observations about variants of our model. We mention open questions about another (weaker) threshold in erasure-resilient testing of connectedness, about erasure-resilient testing of monotone graph properties, about the relationship between testing with erasures and testing with errors, and about the variant of our model that allows only symmetric erasures. We show that some of the questions we discuss are open in our model, but easy in the bounded-degree version of our model.

1.4 Related Work

Erasure-resilient sublinear-time algorithms, in the context of testing properties of functions, were first investigated by Dixit et al. [7], and further studied by Raskhodnikova et al. [25], Pallavoor et al. [22], and Ben-Eliezer et al. [3].

Property testing in the general graph model was first studied by Parnas and Ron [23], who considered a relaxed version of the problem of testing whether the input graph has small diameter. Kaufman et al. [19] studied the problem of testing bipartiteness in the general graph model and obtained tight upper and lower bounds on its complexity.

Sublinear-time algorithms for estimating various graph parameters have also received significant attention. There are sublinear-time algorithms for estimating the weight of a minimum weight spanning tree [6], the number of connected components [6, 4], the average degree [13, 17], the average pairwise distance [17], moments of the degree distribution [18, 9], and subgraph counts [18, 8, 11, 12, 1, 2].

2 Erasure-Resilient Testing of Connectedness

In this section, we present our results on erasure-resilient testing of connectedness in graphs.

2.1 An Erasure-Resilient Connectedness Tester for $\alpha < \varepsilon/2$

In this section, we present our connectedness tester for small α and prove Theorem 1.4. The tester looks for witnesses to disconnectedness in the form of connected components with no erasures. It repeatedly performs a breadth first search (BFS) from a random vertex until it finds a witness to disconnectedness or exceeds a specified query budget.

A simple counting argument shows that if a partially erased graph is far from connected then it has many small witnesses to disconnectedness. Moreover, the size of the average witness among them is at most some bound b (that we calculate later). Our tester uses BFS to detect a witness to disconnectedness of size at most b.

The best tester for connectedness to date, by Berman et al. [5], uses a technique called the *work investment strategy*. Specifically, their algorithm repeatedly samples a uniformly random vertex v, guesses the size of the witness to disconnectedness $C_{(v)}$ containing v, and then performs a BFS from v for $|C_{(v)}|^2$ queries. Clearly, $|C_{(v)}|^2$ queries are enough to detect $C_{(v)}$. Using the fact that the expected size of a witness is b, they argue that their algorithm has complexity $O(b^2)$.

The new idea in our connectedness tester is to perform the BFS from a uniformly random vertex v for $|C_{(v)}| \cdot \deg(v)/2$ queries. The expected value of the latter quantity is bounded by $E_{(v)}$, where $E_{(v)}$ denotes the number of edges in the witness containing v, and the expectation is over the choice of a uniformly random vertex from $C_{(v)}$. That is, in expectation, the number of queries that we *invest* into the BFS from v is enough to detect $C_{(v)}$. We show that, overall, the expected complexity of this algorithm is $\widetilde{O}(b \cdot \overline{d})$, which is smaller than $O(b^2)$ when $b > \overline{d}$.

Our erasure-resilient tester is Algorithm 1, with a small standard modification to ensure that the stated complexity bounds hold in the worst case (not just in expectation). It is obtained by running the algorithm of Berman et al. (generalized to handle erasures) when $b < \overline{d}$ and running the above algorithm otherwise.

Before stating the algorithm, we formalize the notion of the witness to disconnectedness and argue that partially erased graphs that are far from being connected have many witnesses to disconnectedness.

▶ **Definition 2.1** (Witness to disconnectedness). *A set C of vertices is a witness to disconnectedness in a partially erased graph G if the adjacency lists of vertices in C have no erasures, and C forms a connected component in every completion of G.*

▶ **Observation 2.2.** *Let $\varepsilon \in (0, 2/\overline{d})$ and G' be an m-edge graph (with no erasures) that is ε-far from connected. Then G' has at least $\varepsilon m + 1$ connected components.*

Next, in Claim 2.3, we argue that if the fraction of erasures is *small*, *many* of the connected components present in a completion G' are also present as witnesses to disconnectedness in G.

▷ Claim 2.3. Let $\varepsilon \in (0, 2/\overline{d})$ and $\alpha \in [0, \varepsilon/2)$. The number of witnesses to disconnectedness in an α-erased graph G that is ε-far from connected is at least $(\varepsilon - 2\alpha)m$.

Proof. By Observation 2.2, every completion G' of G has at least $\varepsilon m+1$ connected components. The number of connected components in G' with at least one erased entry in the union of its adjacency lists (with respect to G) is at most $2\alpha m$. Hence, the number of connected components in G' that do not have any erased entry in the union of its adjacency lists (with respect to G) is at least $\varepsilon m - 2\alpha m = (\varepsilon - 2\alpha)m$. The claim follows. ◁

Let $b = 2/((\varepsilon - 2\alpha) \cdot \overline{d})$. By Claim 2.3, the size of the average witness to disconnectedness is at most b. Now we are ready to state Algorithm 1.

Algorithm 1 Erasure-Resilient Connectedness Tester for $\alpha < \varepsilon/2$.

```
   input  : The average degree $\overline{d}$, parameters $\varepsilon \in (0, 2/\overline{d}), \alpha \in [0, \varepsilon/2)$; query access to
            an $\alpha$-erased graph $G$.
1  Let $b \leftarrow 2/((\varepsilon - 2\alpha) \cdot \overline{d})$.        // the average size of a witness is at most $b$
2  for $i \in [\lceil \log(4b) \rceil]$ do
3      repeat $\lceil \frac{4b \ln 6}{2^i} \rceil$ times
4          Sample a vertex $v$ uniformly and independently at random.
5          if $b \le \overline{d} \log b$ then
6              Run a BFS from $v$ until it encounters an erased entry or $(2^i + 1)$ vertices.
           else
7              Query $\deg(v)$;
8              Run a BFS from $v$ until it encounters an erased entry or $(2^{i-1} \cdot \deg(v) + 1)$
                edges.
9          if the BFS explored an entire connected component and didn't encounter an
            erasure then reject.
10 Accept.
```

Clearly, Algorithm 1 accepts all connected partially erased graphs.

▶ **Lemma 2.4.** *Let $\varepsilon \in (0, 2/\overline{d})$ and $\alpha \in [0, \varepsilon/2)$. Let G be an α-erased graph that is ε-far from connected. Then Algorithm 1 rejects G with probability at least $5/6$.*

Proof. Let V be the vertex set of G. We start by defining the quality of a vertex $v \in V$. The definition is different for the two cases, corresponding to the two stopping conditions Algorithm 1 uses for BFS. First, we consider the case when $b \le \overline{d} \cdot \log b$, that is, when Algorithm 1 runs the version of BFS specified in Step 6.

▶ **Definition 2.5** (Quality of a vertex when $b \le \overline{d} \cdot \log b$). *The quality of a vertex v, denoted $q(v)$, is defined as follows. If v belongs to a witness to disconnectedness in G then $q(v) = 1/|C_{(v)}|$, where $C_{(v)}$ denotes the witness to disconnectedness that v belongs to. Otherwise, $q(v) = 0$.*

The important feature of $q(v)$ is that, for a witness C to disconnectedness, $\sum_{v \in C} q(v) = 1$.

Next, we define the quality of a vertex for the case when $b > \overline{d} \cdot \log b$, that is, when Algorithm 1 runs the version of BFS specified in Step 8.

▶ **Definition 2.6** (Quality of a vertex when $b > \overline{d} \cdot \log b$). *Fix a completion G' of G. For a vertex $v \in V$, let $C_{(v)}$ denote the connected component (in G') containing v, and let $E_{(v)}$ denote the number of edges in $C_{(v)}$. The quality of a vertex v, denoted $q(v)$, is defined as*

$$q(v) = \begin{cases} 0 & \text{if } C_{(v)} \text{ contains at least one erased entry in } G, \\ \frac{\deg(v)}{2E_{(v)}} & \text{if } E_{(v)} > 0, \\ 1 & \text{if } E_{(v)} = 0. \end{cases}$$

Like for $q(v)$ from Definition 2.5, for a witness C to disconnectedness, $\sum_{v \in C} q(v) = 1$.

The rest of the proof of Lemma 2.4 is the same for both cases. We analyze the expected quality of a uniformly random vertex $v \in V$. Since $\sum_{v \in C} q(v) = 1$, by Claim 2.3,

$$\mathop{\mathbb{E}}_{v \in V}[q(v)] = \frac{1}{n} \sum_{v \in V} q(v) = \frac{1}{n} \sum_{\substack{C: C \text{ is a witness} \\ \text{to disconnectedness}}} 1 \geq \frac{(\varepsilon - 2\alpha)m}{n} = \frac{1}{b}.$$

Finally, we apply the following work investment strategy lemma [5, Lemma 2.5].

▶ **Lemma 2.7** ([5]). *Let X be a random variable that takes values in $[0,1]$. Suppose $\mathbb{E}[X] \geq \beta$, and let $t = \lceil \log(4/\beta) \rceil$. For all $i \in [t]$, let $p_i = \Pr[X \geq 2^{-i}]$ and $k_i = \frac{4 \ln 6}{2^i \beta}$. Then $\prod_{i=1}^{t}(1 - p_i)^{k_i} \leq \frac{1}{6}$.*

We apply Lemma 2.7 with X equal to $q(v)$ for a uniformly random $v \in V$. Set $\beta = 1/b$ and $t = \lceil \log(4/\beta) \rceil$. For $i \in [t]$, set p_i to be the probability that a vertex v sampled uniformly at random belongs to a witness to disconnectedness of G that has at most (i) 2^i vertices, when $b \leq \overline{d} \cdot \log b$; (ii) $2^{i-1} \cdot \deg(v)$ edges, otherwise. That is, $p_i = \Pr[X \geq 2^{-i}]$. Similarly, for $i \in [t]$, let $k_i = \frac{4 \ln 6}{2^i \beta}$. Then the probability that Step 9 of the tester does not reject is $\prod_{i=1}^{t}(1 - p_i)^{k_i}$. By Lemma 2.7, this step rejects with probability at least 5/6. ◀

Proof of Theorem 1.4. We start by analyzing the query and time complexity of Algorithm 1.
Case 1: When $b \leq \overline{d} \cdot \log b$, the query and time complexity of Algorithm 1 is

$$\sum_{i \in [\lceil \log(4b) \rceil]} \left\lceil \frac{4b \ln 6}{2^i} \right\rceil \cdot 2^{2i} = O\left(b^2\right) = O(\min\{b^2, b\overline{d} \cdot \log b\}).$$

Case 2: When $b > \overline{d} \cdot \log b$, the expected query and time complexity of Algorithm 1 is

$$\sum_{i \in [\lceil \log 4b \rceil]} \left\lceil \frac{4b \ln 6}{2^i} \right\rceil \cdot 2^i \cdot \mathop{\mathbb{E}}_{s \in V}[\deg(s)] = O(b\overline{d} \log b) = O(\min\{b^2, b\overline{d} \cdot \log b\}).$$

Substituting the value of b, we get

$$O(\min\{b^2, b\overline{d} \cdot \log b\}) = O\left(\min\left\{\frac{1}{((\varepsilon - 2\alpha)\overline{d})^2}, \frac{1}{\varepsilon - 2\alpha} \log \frac{1}{(\varepsilon - 2\alpha)\overline{d}}\right\}\right).$$

The final tester is obtained by running Algorithm 1 and then aborting and accepting if the number of queries exceeds six times its expectation. The final tester then has the query complexity and the running time stated in Theorem 1.4.

The final tester never rejects a connected partially erased graph. However, a partially erased graph that is ε-far from connected can get accepted incorrectly if Algorithm 1 accepts it or if the final algorithm aborts. The probability of the former event is at most 1/6, by Lemma 2.4. The probability of aborting is also at most 1/6, by Markov's inequality. By a union bound, the final algorithm accepts incorrectly with probability at most 1/3, completing the proof of the theorem for the case when $\overline{d}$ is given to the algorithm.

We can adjust the algorithm to work without access to the average degree at a small cost in query and time complexity. The details appear in the full version [20]. ◀

2.2 Our Erasure-Resilient Connectedness Tester for $\alpha \in [\varepsilon/2, \varepsilon)$

In this section, we prove Theorem 1.5. We describe and analyze a 1-sided error α-erasure-resilient ε-tester for connectedness that can work with more erasures in the input graph than Algorithm 1 can handle. Specifically, the tester works for all $\alpha < \varepsilon$. However, it has better performance than Algorithm 1 only for $\alpha \in [\varepsilon/2, \varepsilon)$.

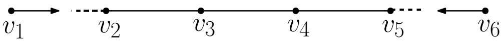

Figure 1 An example of a component with two erasures, where a BFS from any vertex fails to detect that this component is disconnected from the rest of the graph.

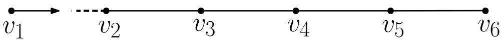

Figure 2 An example of a generalized witness to disconnectedness, where only a BFS from v_1 (but not from any other vertex) detects the generalized witness.

A dotted line represents an erasure in the adjacency list of the corresponding vertex. An arrow pointing from a vertex a in the direction of a vertex b represents that $b \in \mathsf{Adj}(a)$, but $a \notin \mathsf{Adj}(b)$.

When $\alpha > \varepsilon/2$, an α-erased graph that is ε-far from being connected may not contain any witnesses to disconnectedness as defined in Section 2.1. Specifically, every set C of nodes that gets completed to a connected component could have an erasure in the union of the adjacency lists of the nodes in C. To get around this issue, our tester looks for a *generalized witness to disconnectedness*, which is, intuitively, a connected component with at most one erasure. Observe that a component with two erasures could have a unique completion, but impossible to certify as a separate connected component from the local view from any of its vertices. Figure 1 shows an example of a small component, where a BFS from any vertex will be unable to certify that the graph is disconnected.

Our tester repeatedly performs a BFS from a random vertex until it detects a generalized witness to disconnectedness, or exceeds a specified query budget. We show, by a counting argument, that every partially erased graph that is far from connected has several *small* generalized witnesses to disconnectedness. The correctness of the tester is ensured by the observation that each such witness C contains at least one vertex from which all the other vertices in C are reachable. (It is possible to have *exactly* one vertex in C from which all the other vertices are reachable. Figure 2 shows an example of a connected component, where a BFS can detect the generalized witness to disconnectedness only if started at vertex v_1, but will fail to do so from all other vertices.)

Before we state our tester, we formalize the notion of generalized witnesses.

▶ **Definition 2.8** (Generalized witness to disconnectedness). *Given a partially erased graph G over a vertex set V, a set $C \subset V$ is a* generalized witness to disconnectedness *of G if*

1. *there is at most one erased entry ($\perp$) in $\bigcup_{v \in C} \mathsf{Adj}(v)$,*
2. *every nonerased entry in $\bigcup_{v \in C} \mathsf{Adj}(v)$ is a vertex from C,*
3. *if $\perp \in \mathsf{Adj}(u)$ for some $u \in C$ then $u \in \mathsf{Adj}(v)$ but $v \notin \mathsf{Adj}(u)$ for some $v \in C$; moreover, each node in C is reachable via a BFS from v.*

Definition 2.8 implies that the only erasure, if any, in the union of the adjacency lists of the nodes in C is part of a half-erased edge within C, and that C forms a connected component in every completion of G.

Let $b = 4/((\varepsilon - \alpha)\overline{d})$. Our tester is presented in Algorithm 2. In the rest of the section, we analyze the correctness and complexity of the tester.

▶ **Definition 2.9** (Small and big sets). *Let G be a partially erased graph and let $\varepsilon^\star \in (0, 2/\overline{d})$ be a parameter. The* representation length *of a set C of nodes is the sum of lengths of the adjacency lists of nodes in C. The set C is $\varepsilon^\star$-*small *if either*

- *$\varepsilon^\star \geq 4/\overline{d}^2$ and C contains at most $4/(\varepsilon^\star \cdot \overline{d})$ vertices, or*
- *$\varepsilon^\star < 4/\overline{d}^2$ and C has representation length at most $4/\varepsilon^\star$.*

*The set C is $\varepsilon^\star$-*big *otherwise.*

Claim 2.10 shows that a partially erased graph that is far from connected has sufficiently many small generalized witnesses to disconnectedness.

Algorithm 2 Erasure-Resilient Connectedness Tester for $\alpha \in [\varepsilon/2, \varepsilon)$.

```
input : The average degree $\overline{d}$, parameters $\varepsilon \in (0, 2/\overline{d}), \alpha \in [0, \varepsilon)$; query access to an
        $\alpha$-erased graph $G$.
1 Let $b \leftarrow 4/((\varepsilon - \alpha)\overline{d})$.
2 repeat $\lceil b \ln 3 \rceil$ times
3     Sample a vertex $s$ uniformly and independently at random.
4     Run a BFS starting from $s$ using at most $\min\{b^2, b \cdot \overline{d}\}$ neighbor queries.
5     if Step 4 detected a generalized witness to disconnectedness then
6         Reject.
7 Accept.
```

▷ Claim 2.10. Let $\varepsilon \in (0, 2/\overline{d}), \alpha \in [0, \varepsilon)$. Let G be an α-erased graph that is ε-far from connected. The number of $(\varepsilon - \alpha)$-small generalized witnesses to disconnectedness of G is at least $(\varepsilon - \alpha)m/2$.

Proof. We first argue that there are many small connected components in every completion G' of G and then prove that many of these are generalized witnesses in G.

Consider a completion G' of G. If $\varepsilon - \alpha \geq 4/\overline{d}^2$, the number of $(\varepsilon - \alpha)$-big connected components in G' is at most $n/b = (\varepsilon - \alpha)m/2$. If $\varepsilon - \alpha < 4/\overline{d}^2$, the number of $(\varepsilon - \alpha)$-big connected components in G' is at most $2m/(b \cdot \overline{d}) = (\varepsilon - \alpha)m/2$, since the representation length of the vertex set V of G is $2m$. By Observation 2.2, the total number of connected components in G' is at least $\varepsilon m + 1$. Hence, the number of $(\varepsilon - \alpha)$-small connected components in G' is at least $(\varepsilon + \alpha)m/2$.

Let $C \subset V$ denote the set of vertices corresponding to an $(\varepsilon - \alpha)$-small connected component in G'. If $\bigcup_{v \in C} \mathsf{Adj}(v)$ has no erasures, then C is a generalized witness to disconnectedness of G. Next, assume that $\bigcup_{v \in C} \mathsf{Adj}(v)$ has exactly one erasure. We show that the set C is a generalized witness to disconnectedness of G. Condition 1 is satisfied by definition. Condition 2 is true since C forms a connected component in G'. To see that Condition 3 holds, let $u \in C$ be the vertex with $\perp \in \mathsf{Adj}(u)$. Since C is a connected component in G', this erased entry was completed with the label of another vertex $v \in C$. Moreover, every vertex in C is reachable by a BFS from v, since C forms a connected component in G', and the erased entry is not needed for these searches because it would lead back to v. Therefore, C is a generalized witness to disconnectedness of G if $\bigcup_{v \in C} \mathsf{Adj}(v)$ has exactly one erasure.

Among the $(\varepsilon - \alpha)$-small connected components in G', at most αm have at least 2 erased entries in the union of their adjacency lists. Hence, the number of $(\varepsilon - \alpha)$-small generalized witnesses to disconnectedness of G is at least $((\varepsilon + \alpha)m/2) - \alpha m = (\varepsilon - \alpha)m/2$. ◁

Lemma 2.11 below implies Theorem 1.5.

▶ **Lemma 2.11.** *For every $\varepsilon \in (0, 2/\overline{d})$ and $\alpha \in [0, \varepsilon)$, Algorithm 2 is an α-erasure-resilient ε-tester for connectedness of graphs with the average degree $\overline{d}$. It has $O(b^2\overline{d} \cdot \min\{b/\overline{d}, 1\})$ query and time complexity.*

Proof. Consider an α-erased graph G over a vertex set V. Assume that G is connected, that is, there exists a connected completion G' of G. Consider an arbitrary $C \subset V$. There exist vertices $u \in C$ and $v \in V \setminus C$ such that $\mathsf{Adj}(u)$ in G' contains v. Hence, C is not a generalized witness to disconnectedness of G. Therefore, the tester accepts G.

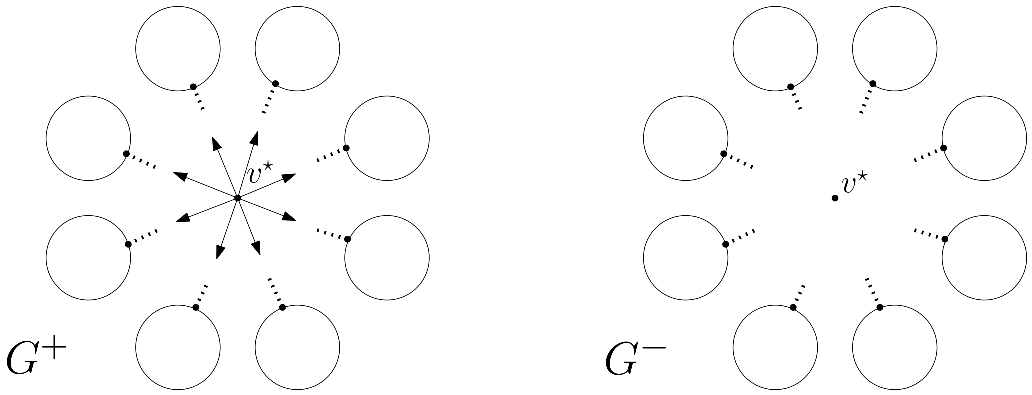

Figure 3 The partially erased graphs G^+ and G^- described in the proof of Theorem 1.6. The dotted lines represent erased entries in the adjacency lists of the corresponding vertices. In G^+, the directed edges from $v^\star$ point to the vertices in its adjacency list. The circles represent cycles.

Next, assume that G is ε-far from connected. Let $\mathcal{W}$ denote the family of all $(\varepsilon-\alpha)$-small generalized witnesses to disconnectedness of G. Let $C \subset V$ be an element of $\mathcal{W}$. If $\varepsilon - \alpha \geq 4/\overline{d}^2$, the representation length of C is at most $b^2 \leq b \cdot \overline{d}$. If $\varepsilon - \alpha < 4/\overline{d}^2$, the representation length of C is at most $b \cdot \overline{d} < b^2$. Hence, the representation length of C is at most $\min\{b^2, b \cdot \overline{d}\}$. If $\bigcup_{v \in C} \mathsf{Adj}(v)$ has no erasures then every vertex in C is reachable from every other vertex in C. Otherwise, the vertex v in Condition 3 of Definition 2.8 is such a vertex. If Algorithm 2 performs a BFS from v, it will detect a generalized witness to disconnectedness after at most $\min\{b^2, b \cdot \overline{d}\}$ queries and reject. Since $|\mathcal{W}| \geq (\varepsilon - \alpha)m/2$ and each generalized witness in $\mathcal{W}$ has at least one vertex from which the generalized witness is detectable by a BFS, a single iteration of Algorithm 2 rejects with probability at least $|\mathcal{W}|/n = 1/b$. Hence, Algorithm 2 rejects with probability at least $1 - (1 - (1/b))^{\lceil b \ln 3 \rceil} \geq 1 - \exp(-\ln 3) = 2/3$.

Step 4 of Algorithm 2 makes at most $\min\{b^2, b\overline{d}\}$ queries. Thus, the query complexity of Algorithm 2 is $O(b \cdot \min\{b^2, b\overline{d}\})$, which simplifies to the claimed expression. Checking (in Step 5) whether a set C is a generalized witness to disconnectedness can be done with a constant number of passes over the adjacency lists of vertices in C. Since the algorithm queried all entries in them, its running time is asymptotically equal to its query complexity. ◀

2.3 A Lower Bound for Erasure-Resilient Connectedness Testing

In this section, we prove Theorem 1.6. We note that hard graphs in our construction have constant average degree. That is, for those graphs, our lower bound is $\Omega(n) = \Omega(m)$.

Proof of Theorem 1.6. We apply Yao's minimax principle, as stated in [26]. Specifically, we construct distributions $\mathcal{D}^+$ and $\mathcal{D}^-$, the former over connected graphs and the latter over graphs that are ε-far from connected, such that every deterministic ε-erasure-resilient ε-tester for connectedness makes $\Omega(m)$ queries to distinguish the two distributions.

Without loss of generality, assume that $t = (1-\varepsilon)/(2\varepsilon)$ is an integer. Observe that $t \geq 3$ as $\varepsilon \leq 1/7$. Let k be an even number and $n = kt + 1$. We first construct two partially erased n-node graphs G^+ and G^-, depicted in Figure 3. The vertices of G^+ are partitioned into $k+1$ sets. Each of the first k sets induces a t-node cycle. Exactly one node in each cycle has degree 3 and has an erasure in its adjacency list, in addition to its two neighbors on the cycle. The last set contains a single node $v^\star$ of degree k. Its adjacency list contains the labels of the degree-3 vertices in the cycles. The graph G^- is the same as G^+, except that in G^-, we have that $\mathsf{Adj}(v^\star)$ is empty, that is, $v^\star$ is isolated.

We can obtain a connected completion of G^+ by connecting the vertex $v^\star$ to all the degree-3 vertices. In contrast, at least $k/2$ edges need to be added to every completion of G^- to make it connected. Hence, the distance from G^- to connectedness is $(k/2)/(kt+k/2) = 1/(2t+1) = \varepsilon$.

The fraction of erased entries in the adjacency lists of G^+ and G^- are $1/(2t+2)$ and $1/(2t+1)$, respectively. That is, G^+ and G^- are both α-erased graphs for $\alpha = 1/(2t+1) = \varepsilon$.

The distributions $\mathcal{D}^+$ and $\mathcal{D}^-$ are uniform over the sets of all partially erased graphs isomorphic to G^+ and G^-, respectively. Each partially erased graph sampled from $\mathcal{D}^+$ is connected. Each partially erased graph sampled from $\mathcal{D}^-$ is ε-far from connected.

▷ Claim 2.12. Every deterministic algorithm A has to make $\Omega(n)$ queries to distinguish $\mathcal{D}^+$ and $\mathcal{D}^-$ with probability at least $2/3$.

Proof. Let q denote the number of queries made by A and assume $q \leq n/6$. In this proof, we use $v^\star$ as a shorthand for the vertex from the singleton set in the construction of $\mathcal{D}^+$ and $\mathcal{D}^-$, as opposed to the label of that vertex. Since $\mathcal{D}^+$ and $\mathcal{D}^-$ differ only on $v^\star$, it is important to understand when A gets any information about $v^\star$.

▶ **Definition 2.13** (Node status). *Given a sequence of queries made by A and answers it has received so far, a node v is* known *if it has been queried (via a degree or neighbor query) or received as an answer to a (neighbor) query; otherwise, it is* unknown.

The node $v^\star$ is *unknown* before A makes its first query. Since $v^\star$ cannot be received as an answer to a query for the graphs in the support of $\mathcal{D}^+$ and $\mathcal{D}^-$, it can become *known* only if A queries an *unknown* node that happens to be $v^\star$. At most two new nodes become *known* per query. So, the probability (over the distribution $\mathcal{D}^+$ or $\mathcal{D}^-$) that a specific *unknown* node queried by A turns out to be $v^\star$ is at most $1/(n-2q)$. Let p denote the probability that $v^\star$ becomes *known* by the end of an execution of A. By a union bound over all queries made by A, we have, $p \leq \frac{q}{n-2q} \leq \frac{n/6}{n-n/3} = \frac{1}{4}$.

If $v^\star$ is *unknown* by the end of a particular execution then the view of the partially erased graph obtained by A in that execution arises with the same probability under $\mathcal{D}^+$ and under $\mathcal{D}^-$. Such an execution of A can distinguish $\mathcal{D}^+$ and $\mathcal{D}^-$ with probability at most $1/2$. Therefore, the probability that A distinguishes $\mathcal{D}^+$ and $\mathcal{D}^-$ is at most $p+(1-p)\cdot\frac{1}{2} = \frac{1}{2}+\frac{p}{2} < \frac{2}{3}$. ◁

In our construction, $m = \Theta(n)$. Thus, every ε-erasure-resilient ε-tester for connectedness that uses only degree and neighbor queries must make $\Omega(m)$ queries in the worst case over the input graph, completing the proof of Theorem 1.6. ◀

3 Estimating the Average Degree of a Graph

In this section, we present our results on erasure-resilient estimation of the average degree of graphs.

3.1 An Algorithm for Estimating the Average Degree

In this section, we describe and analyze the algorithm (claimed in Theorem 1.7) for estimating the average degree of (or, equivalently, the number of edges in) a partially erased graph. Our algorithm is a generalization of the algorithm for counting the number of edges in graphs by Eden et al. [9, 10] to the case of partially erased graphs. We first give an algorithm (Algorithm 3) that takes a crude estimate of the average degree as input and outputs a more accurate estimate. Using a standard technique similar to the binary search, our final algorithm uses Algorithm 3 as a subroutine to gradually refine its estimate of the average degree. The final algorithm and the complete proof of Theorem 1.7 appear in the full version.

Algorithm 3, like the algorithm of Eden et al. [9, 10], works by empirically estimating a random variable whose expectation is close to the number of edges in the graph. We first rank vertices according to their degrees, breaking ties arbitrarily. Then we orient the nonerased edges of the graph from lower-ranked to higher-ranked endpoints. This orientation allows us to attribute each nonerased edge to its lower-ranked endpoint in order to avoid double-counting the edge. Since the number of edges between high-degree vertices is small, we ignore such edges. Algorithm 3 samples low-degree vertices uniformly at random and estimates, via sampling, the number of edges "credited" to them.

The crucial difference in the behavior of the algorithm in the case of partially erased graphs is the following. When we sample an erased entry from the adjacency list of a low-degree vertex u, we assume that it gets completed to a vertex ranked higher than u and, therefore, attribute the corresponding edge to u. Consequently, some erased edges get counted twice. This results in the additional term depending on the fraction of erasures in the approximation guarantee.

The ranking of (or the total ordering on) the vertices of a graph is defined below.

▶ **Definition 3.1** (Total ordering $\prec$). *In a partially erased graph G, for any two vertices u, v, we write $u \prec v$ if either $\deg(u) < \deg(v)$, or $\deg(u) = \deg(v)$ and u is lexicographically smaller than v.*

Algorithm 3 Erasure-Resilient Algorithm for Improving an Estimate of Average Degree.

```
input : Parameters $\varepsilon \in (0, 1/2), \delta \in (0, 1/3)$; query access to a partially erased
        graph $G$ on $n$ nodes; a crude estimate $\widehat{d}$ of the average degree of $G$.
Set $s \leftarrow \left\lceil 660 \ln(2/\delta) \sqrt{\frac{n}{\varepsilon^5 \cdot \widehat{d}}} \right\rceil$.
for $i = 1$ to $s$ do
    Sample a node $u$ from $V$ uniformly at random and query its degree, $\deg(u)$.
    Query a uniformly random entry from $\mathsf{Adj}(u)$ and let $v$ be the answer.
    If $v \neq \perp$ then query its degree, $\deg(v)$.
    if $\deg(u) \leq 4\sqrt{n\widehat{d}/\varepsilon}$ and either $v = \perp$ or $u \prec v$ then
        $\chi_i \leftarrow \deg(u)$
    else
        $\chi_i \leftarrow 0$
return $\widetilde{d} = 2 \cdot \frac{1}{s} \sum_{i=1}^{s} \chi_i$ .
```

▶ **Lemma 3.2.** *Let G be an α-erased n-node graph with the average degree $\overline{d} \geq 1$. Let $\widehat{d}$ be a crude estimate of the average degree, given as an input to Algorithm 3. Then the output $\widetilde{d}$ of Algorithm 3 satisfies the following:*

1. *If $\widehat{d} \geq \frac{\overline{d}}{8}$ then, with probability at least $3/4$, we have $\widetilde{d} \leq 8\overline{d}$.*
2. *Furthermore, if $\frac{\overline{d}}{8} \leq \widehat{d} \leq 8\overline{d}$ then with probability at least $1 - \delta$,*

$$(1 - \varepsilon) \cdot \overline{d} < \widetilde{d} < (1 + \varepsilon + 2\min(\alpha, \tfrac{1}{2})) \cdot \overline{d}.$$

The query complexity of the algorithm is $\Theta\left(\sqrt{\frac{n}{\varepsilon^5 \cdot \widehat{d}}} \cdot \log \frac{1}{\delta}\right)$.

Proof. The algorithm makes at most two degree queries and one neighbor query in each iteration, and it runs for $\Theta\Big(\sqrt{\frac{n}{\varepsilon^5 \cdot \widehat{d}}} \cdot \log \frac{1}{\delta}\Big)$ iterations. Hence, the bound on its query complexity is as claimed in the lemma.

To prove the guarantees on the output estimate $\widetilde{d}$, we first show that for all $i \in [s]$, the expected value of χ_i is a good estimate to the average degree of the partially erased graph, where s is the number of samples taken by Algorithm 3. We then apply Markov's inequality and Chernoff bound to prove parts 1 and 2 of the lemma, respectively. For all $i \in [s]$, the random variables χ_i set by the algorithm are mutually independent and identically distributed. Hence, it suffices to bound $\mathbb{E}[\chi_1]$.

▷ Claim 3.3. If $\widehat{d} \geq \frac{\overline{d}}{8}$ then

$$\left(1 - \frac{\varepsilon}{2}\right) \cdot \frac{\overline{d}}{2} < \mathbb{E}[\chi_1] \leq \left(1 + 2\min\left(\alpha, \frac{1}{2}\right)\right) \cdot \frac{\overline{d}}{2}.$$

Proof. Let $m = n\overline{d}/2$ denote the total number of edges in the graph, and

$$\mathcal{H} = \left\{ u \in V \;\middle|\; \deg(u) > 4\sqrt{n\widehat{d}/\varepsilon} \right\}$$

denote the set of high degree vertices. Let $\widehat{m} = n\widehat{d}/2$ be the number of edges in the graph estimated from the input parameter $\widehat{d}$. Since $\widehat{d} \geq \overline{d}/8$, we have $\widehat{m} \geq m/8$. Hence,

$$|\mathcal{H}| < \frac{2m}{4\sqrt{n\widehat{d}/\varepsilon}} = \frac{m}{2\sqrt{2\widehat{m}/\varepsilon}} \leq \frac{m}{\sqrt{m/\varepsilon}} = \sqrt{\varepsilon m}, \tag{1}$$

where the first inequality holds because the sum of degrees of high-degree vertices is at most $2m$, and the second inequality follows from $\widehat{m} \geq m/8$.

The following quantity, $d^+(u)$, was defined in [10] for (standard) graphs. We extend their definition to partially erased graphs.

▶ **Definition 3.4.** *For a vertex u in a partially erased graph G, let $N(u)$ denote the set of (nonerased) neighbors present in* $\mathsf{Adj}(u)$. *Let $d^+(u) = |\{v \in N(u) \mid u \prec v\}|$ denote the number of nonerased neighbors of u that are higher than u w.r.t. the ordering on vertices (as in Definition 3.1).*

Roughly, $d^+(u)$ denotes the number of nonerased neighbors of u with the degree higher than that of u. The following fact is based on an observation by [10].

▶ **Fact 3.5.** *For a partially erased graph G over a vertex set V, the sum $\sum_{u \in V} d^+(u) \leq m$. The inequality can be replaced with equality when G has no erasures.*

The fact holds because each nonerased and half-erased edge in G is counted exactly once and at most once, respectively, in the sum $\sum_{u \in V} d^+(u)$.

Let $u_1, u_2, \ldots, u_{|\mathcal{H}|}$ be a labeling of the the high degree vertices such that $u_1 \prec u_2 \prec \ldots \prec u_{|\mathcal{H}|}$. For each $j \in [|\mathcal{H}|]$, observe that $d^+(u_j) \leq |\mathcal{H}| - j$, as $d^+(u_j)$ is at most the number of vertices that are higher than u_j in the ordering. Hence,

$$\sum_{u \in \mathcal{H}} d^+(u) \leq \sum_{j=1}^{|\mathcal{H}|} (|\mathcal{H}| - j) = \sum_{k=0}^{|\mathcal{H}|-1} k < \frac{|\mathcal{H}|^2}{2} < \frac{\varepsilon m}{2}, \tag{2}$$

where the last inequality follows from (1).

Let $d^{\perp}(u)$ denote the number of erased entries in $\mathsf{Adj}(u)$. The expectation

$$\mathbb{E}[\chi_1] = \frac{1}{n}\sum_{u\in V\setminus\mathcal{H}} \frac{d^+(u)+d^{\perp}(u)}{\deg(u)}\cdot\deg(u) = \frac{1}{n}\sum_{u\in V\setminus\mathcal{H}}(d^+(u)+d^{\perp}(u)) \tag{3}$$

since the degree of the sampled vertex u is assigned to χ_1 if and only if

1. $\deg(u)\le 4\sqrt{n\widehat{d}/\varepsilon}$, i.e., $u\in V\setminus\mathcal{H}$; and
2. the queried entry from $\mathsf{Adj}(u)$ is either a vertex $v\succ u$ or $\perp$.

We now bound the quantity on the right hand side of (3) from below and above. Let G' be an arbitrary completion of G, and let $d^+_{G'}(\cdot)$ denote the quantity defined in Definition 3.4 with respect to G' (instead of G). For each $u\in V$, observe that $d^+(u)+d^{\perp}(u)\ge d^+_{G'}(u)$. Note that the upper bound in (2) still holds if we replace $d^+(\cdot)$ with $d^+_{G'}(\cdot)$. Hence, by (3),

$$\mathbb{E}[\chi_1] \ge \frac{1}{n}\sum_{u\in V\setminus\mathcal{H}} d^+_{G'}(u) = \frac{1}{n}\left(m-\sum_{u\in\mathcal{H}} d^+_{G'}(u)\right) > \left(1-\frac{\varepsilon}{2}\right)\frac{m}{n}. \tag{4}$$

On the other hand, by (3),

$$\mathbb{E}[\chi_1] \le \frac{1}{n}\sum_{u\in V}(d^+(u)+d^{\perp}(u)) \le (1+2\alpha)\frac{m}{n}, \tag{5}$$

where the last inequality uses Fact 3.5 and $\sum_{u\in V} d^{\perp}(u)\le 2\alpha m$. Since $d^+(u)+d^{\perp}(u)\le\deg(u)$ for all $u\in V$, by (3),

$$\mathbb{E}[\chi_1] \le \frac{1}{n}\sum_{u\in V}\deg(u) = \frac{2m}{n}. \tag{6}$$

This completes the proof of Claim 3.3 because, using (4),(5) and (6), we get

$$\left(1-\frac{\varepsilon}{2}\right)\cdot\frac{m}{n} < \mathbb{E}[\chi_1] \le \left(1+2\min\left(\alpha,\frac{1}{2}\right)\right)\cdot\frac{m}{n}. \qquad \lhd$$

Let random variable $\chi=\frac{1}{s}\sum_{i=1}^{s}\chi_i$ denote the mean of χ_i's calculated in Step 9 of Algorithm 3. Since all χ_i's are independent and identically distributed, $\mathbb{E}[\chi]=\mathbb{E}[\chi_1]$. Furthermore, the output $\widetilde{d}$ of the algorithm is 2χ and hence, $\mathbb{E}[\widetilde{d}]=2\,\mathbb{E}[\chi]$. By Claim 3.3, if $\widehat{d}\ge\overline{d}/8$ then $\mathbb{E}[\widetilde{d}]\le 2\overline{d}$. By Markov's inequality, $\Pr[\widetilde{d}>8\overline{d}]\le\Pr[\widetilde{d}>4\,\mathbb{E}[\widetilde{d}]]\le 1/4$. This completes the proof of part 1 of Lemma 3.2.

Now consider the case when $\frac{\overline{d}}{8}\le\widehat{d}\le 8\overline{d}$. Observe that $0\le\chi_i\le 4\sqrt{n\widehat{d}/\varepsilon}$ for all $i\in[s]$ by Step 6. Hence, by an application of the Hoeffding bound,

$$\Pr\left[|\chi-\mathbb{E}[\chi]|\ge\frac{\varepsilon}{2}\cdot\mathbb{E}[\chi]\right]\le 2\exp\left(-\frac{\varepsilon^2/4}{2+\varepsilon/2}\cdot\frac{s\,\mathbb{E}[\chi]}{4}\sqrt{\frac{\varepsilon}{n\widehat{d}}}\right)<\delta,$$

where we used $\varepsilon<1/2$ and $\widehat{d}\le 8\overline{d}$ in the simplification. Hence, with probability at least $1-\delta$, we have, $\left(1-\frac{\varepsilon}{2}\right)\cdot\mathbb{E}[\chi_1]<\chi<\left(1+\frac{\varepsilon}{2}\right)\cdot\mathbb{E}[\chi_1]$. Since $\widetilde{d}=2\chi$, by Claim 3.3, we get that with probability at least $1-\delta$,

$$\left(1-\frac{\varepsilon}{2}\right)\left(1-\frac{\varepsilon}{2}\right)\cdot\overline{d}<\widetilde{d}<\left(1+\frac{\varepsilon}{2}\right)\left(1+2\min\left(\alpha,\frac{1}{2}\right)\right)\cdot\overline{d},$$

proving part 2 of Lemma 3.2. ◀

The rest of the proof of Theorem 1.7 appears in the full version of this article [20].

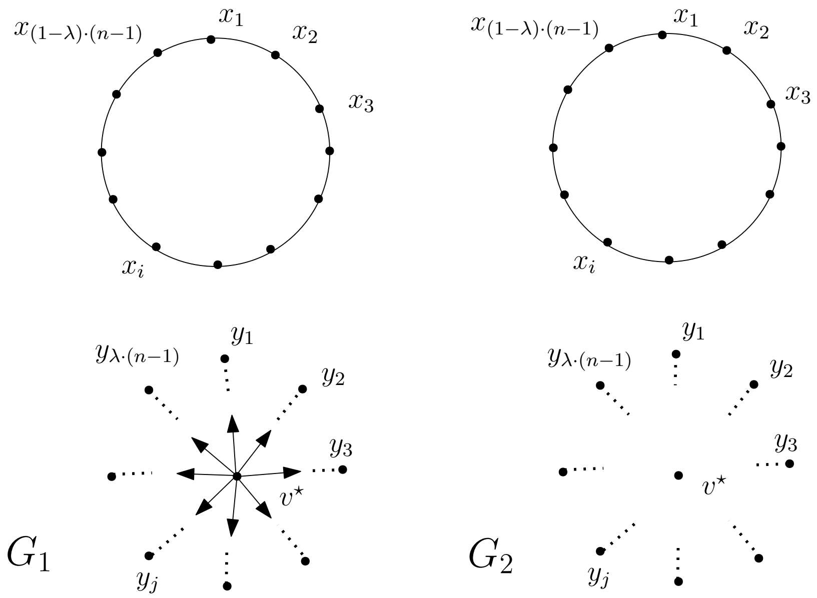

Figure 4 The partially erased graphs G_1 and G_2 described in the proof of Theorem 1.8. The dotted lines represent erased entries in the adjacency lists of corresponding vertices. The lines with arrows indicate that the entry corresponds to the vertex to which the arrow points to. The circles represent the $(1-\lambda)(n-1)$-cycles.

3.2 A Lower Bound for Estimating the Average Degree

In this section, we prove Theorem 1.8.

Proof of Theorem 1.8. Fix $\lambda = \frac{2\alpha}{1+\alpha}$. Note that $\lambda \in (0,1]$ since $\alpha \in (0,1]$. Consider any integer n such that $\lambda(n-1)$ is an even integer. Since α is rational, there are infinitely many such n. We define two n-node graphs, G_1 and G_2 (see Figure 4). Both graphs contain a cycle consisting of $(1-\lambda)(n-1)$ vertices. Of the remaining $\lambda(n-1)+1$ vertices, both graphs have $\lambda(n-1)$ vertices of degree 1, with the only entry in the adjacency list of each such vertex erased. The last vertex, called $v^\star$, is where G_1 and G_2 differ. In G_1, we have that $\mathsf{Adj}(v^\star)$ consists of the labels of the $\lambda(n-1)$ degree-1 vertices. In contrast, in G_2, the vertex $v^\star$ is isolated.

The graph G_1 can only be completed to a graph consisting of two components: a cycle of length $(1-\lambda)(n-1)$ and a star consisting of $\lambda(n-1)$ edges. The graph G_2 can only be completed to a graph consisting of a cycle of length $(1-\lambda)(n-1)$, one isolated vertex, and a matching of size $\lambda(n-1)/2$. Hence, the total lengths of the adjacency lists of G_1 and G_2 are $2(n-1)$ and $(2-\lambda)(n-1)$, respectively. The number of entries erased in both graphs is $\lambda(n-1)$. So, the fraction of erased entries in the adjacency lists of G_1 and G_2 are $\frac{\lambda}{2}$ and $\frac{\lambda}{2-\lambda}$, respectively. Hence, both G_1 and G_2 are α-erased, as $\frac{\lambda}{2-\lambda} = \alpha$. The average degree of G_1 and G_2 are $\frac{2(n-1)}{n}$ and $\frac{(2-\lambda)(n-1)}{n}$, respectively. The ratio of the average degrees is $\frac{2}{2-\lambda} = 1+\alpha$.

The rest of the proof is similar to that of Theorem 1.6. We define two distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ as the uniform distributions over the set of all graphs isomorphic to G_1 and G_2, respectively. To differentiate between the two distributions, any tester must necessarily query $v^\star$ which requires $\Omega(n)$ queries. The ratio of the average degrees of the two distributions is $1+\alpha$. Hence, to approximate the average degree within a factor of $(1+\gamma)$, where $\gamma < \alpha$, any tester must query $\Omega(n)$ vertices. ◀

4 Conclusion and Open Questions

In this work, we initiate the study of sublinear-time algorithms for problems on partially erased graphs. Our investigation opens up a plethora of research directions and possibilities for future work. Next, we discuss several specific open questions arising from our work.

Phase Transitions in the Complexity of Erasure-Resilient Connectedness Testing

As shown in Section 2, there is a phase transition in the complexity of connectedness testing at $\alpha = \varepsilon$ from time independent of the size of the graph to $\Omega(n)$. Our upper bound on the complexity of this problem exhibits another, less drastic phase transition at $\alpha = \varepsilon/2$, when the asymptotic dependence of the running time on ε and α changes. We conjecture that this second phase transition is inherent (and not an artifact of our techniques). It would be interesting to investigate whether connectedness testing when $\alpha \in [\varepsilon/2, \varepsilon)$ is fundamentally different from the same problem when $\alpha \in [0, \varepsilon/2)$.

Erasure-Resilient Testing of Monotone Properties in the Bounded-Degree Model

A property of a graph is *monotone* if it is preserved under deletion of edges and vertices. That is, if G satisfies a monotone property then so does every subgraph of G. Many important graph properties, including bipartiteness, 3-colorability, and triangle-freeness, are monotone.

In the bounded-degree property testing model [16], an n-node graph G with the degree bound D is represented as a concatenation of n adjacency lists, each of length D. For a vertex $v \in G$ and an index $i \in [D]$, a neighbor query (v, i) returns a valid vertex in the graph if $i \leq \deg(v)$ and a special symbol, say $\sqcup$, if $i > \deg(v)$. The graph G is ε-far from satisfying a property $\mathcal{P}$ if at least εnD entries in the adjacency lists of G need to be modified to make it satisfy $\mathcal{P}$.

Bounded-degree property testing can be generalized in a natural way to account for erased entries in adjacency lists. A bounded-degree graph is α-erased if at most αnD entries of its adjacency lists are erased. We observe that a tester for a monotone property of bounded-degree graphs can be made erasure-resilient via a simple transformation.

▶ **Observation 4.1.** *Let $\mathcal{P}$ be a monotone property of graphs. Suppose there exists an ε-tester for $\mathcal{P}$ in the bounded-degree model that makes $q(\varepsilon, n, D)$ queries. Then there exists an α-erasure-resilient ε-tester for $\mathcal{P}$ in the bounded-degree model that makes at most $D^2 \cdot q(\varepsilon - 2\alpha, n, D)$ queries and works for all $\alpha \in (0, \varepsilon/2)$.*

This transformation is not efficient for general graphs, as the maximum degree of a graph can be $n - 1$. It is interesting to understand how much erasure-resilience affects query complexity of testing monotone properties in our erasure-resilient model for general graphs.

Erasure-Resilient vs. Tolerant Testing of Graphs

For $0 \leq \varepsilon_1 < \varepsilon_2 < 1$, an $(\varepsilon_1, \varepsilon_2)$-*tolerant tester* for a property $\mathcal{P}$ must accept, with high probability, if the input is ε_1-close[4] to $\mathcal{P}$ and reject, with high probability, if the input is ε_2-far from $\mathcal{P}$ [24]. Dixit et al. [7] observed that, for properties of functions, erasure-resilient testing is no harder than tolerant testing. Specifically, a tolerant tester for a property of functions can be easily converted to an erasure-resilient tester with the same complexity.

[4] An object is ε_1-close to a property $\mathcal{P}$ if it is not ε_1-far from $\mathcal{P}$.

The new tester can run the tolerant tester, filling in the queried erasures with arbitrary values. However, this argument fails in the case of testing properties of graphs represented as adjacency lists, since the erased entries have to be filled in so that the resulting completion is a valid graph. In the bounded-degree model, we can use a $(2\alpha, \varepsilon - 2\alpha)$-tolerant tester for a property $\mathcal{P}$ to obtain an α-erasure-resilient ε-tester for $\mathcal{P}$ with an overhead $O(D^2)$ in query complexity via a transformation similar to the one explained in our discussion of monotone properties. It is an important open question to understand the relationship between erasure-resilient and tolerant testing in the general graph model.

Symmetric vs. Asymmetric Erasures

Our definition of partially erased graphs is general in the sense that erased entries may be *asymmetric*: an edge (u, v) can be erased in $\mathsf{Adj}(u)$, but not in $\mathsf{Adj}(v)$. A partially erased graph has only *symmetric* erasures if it has no half-erased edges, that is, $u \in \mathsf{Adj}(v)$ iff $v \in \mathsf{Adj}(u)$ for any two nodes u, v. It is an interesting direction to investigate which computational tasks are strictly easier in the model with symmetric erasures compared to the model with asymmetric erasures.

References

1 Maryam Aliakbarpour, Amartya Shankha Biswas, Themis Gouleakis, John Peebles, Ronitt Rubinfeld, and Anak Yodpinyanee. Sublinear-time algorithms for counting star subgraphs via edge sampling. *Algorithmica*, 80(2):668–697, 2018. doi:10.1007/s00453-017-0287-3.

2 Sepehr Assadi, Michael Kapralov, and Sanjeev Khanna. A simple sublinear-time algorithm for counting arbitrary subgraphs via edge sampling. In *Proc. of Innovations in Theoretical Computer Science (ITCS)*, pages 6:1–6:20, 2019. doi:10.4230/LIPIcs.ITCS.2019.6.

3 Omri Ben-Eliezer, Eldar Fischer, Amit Levi, and Ron D. Rothblum. Hard properties with (very) short PCPPs and their applications. In *Proc. of Innovations in Theoretical Computer Science (ITCS)*, pages 9:1–9:27, 2020. doi:10.4230/LIPIcs.ITCS.2020.9.

4 Petra Berenbrink, Bruce Krayenhoff, and Frederik Mallmann-Trenn. Estimating the number of connected components in sublinear time. *Inf. Process. Lett.*, 114(11):639–642, 2014. doi:10.1016/j.ipl.2014.05.008.

5 Piotr Berman, Sofya Raskhodnikova, and Grigory Yaroslavtsev. L_p-testing. In *Proc. of ACM Symposium on Theory of Computing (STOC)*, pages 164–173, 2014. doi:10.1145/2591796.2591887.

6 Bernard Chazelle, Ronitt Rubinfeld, and Luca Trevisan. Approximating the minimum spanning tree weight in sublinear time. *SIAM J. Comput.*, 34(6):1370–1379, 2005. doi:10.1137/S0097539702403244.

7 Kashyap Dixit, Sofya Raskhodnikova, Abhradeep Thakurta, and Nithin Varma. Erasure-resilient property testing. *SIAM J. Comput.*, 47(2):295–329, 2018. doi:10.1137/16M1075661.

8 Talya Eden, Amit Levi, Dana Ron, and C. Seshadhri. Approximately counting triangles in sublinear time. *SIAM J. Comput.*, 46(5):1603–1646, 2017. doi:10.1137/15M1054389.

9 Talya Eden, Dana Ron, and C. Seshadhri. Sublinear time estimation of degree distribution moments: The degeneracy connection. In *Proc. of Intl. Colloquium on Automata, Languages and Programming (ICALP)*, pages 7:1–7:13, 2017. doi:10.4230/LIPIcs.ICALP.2017.7.

10 Talya Eden, Dana Ron, and C. Seshadhri. Extremely simple algorithm for estimating the number of edges. Personal Communication, 2019.

11 Talya Eden, Dana Ron, and C. Seshadhri. On approximating the number of k-cliques in sublinear time. *SIAM J. Comput.*, 49(4):747–771, 2020. doi:10.1137/18M1176701.

12 Talya Eden and Will Rosenbaum. Lower bounds for approximating graph parameters via communication complexity. In *Proc. of Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX-RANDOM)*, pages 11:1–11:18, 2018. doi:10.4230/LIPIcs.APPROX-RANDOM.2018.11.

13 Uriel Feige. On sums of independent random variables with unbounded variance and estimating the average degree in a graph. *SIAM J. Comput.*, 35(4):964–984, 2006. doi:10.1137/S0097539704447304.

14 Oded Goldreich. *Introduction to Property Testing*. Cambridge University Press, 2017. doi:10.1017/9781108135252.

15 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998. doi:10.1145/285055.285060.

16 Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2):302–343, 2002. doi:10.1007/s00453-001-0078-7.

17 Oded Goldreich and Dana Ron. Approximating average parameters of graphs. *Random Struct. Algorithms*, 32(4):473–493, 2008. doi:10.1002/rsa.20203.

18 Mira Gonen, Dana Ron, and Yuval Shavitt. Counting stars and other small subgraphs in sublinear-time. *SIAM J. Discrete Math.*, 25(3):1365–1411, 2011. doi:10.1137/100783066.

19 Tali Kaufman, Michael Krivelevich, and Dana Ron. Tight bounds for testing bipartiteness in general graphs. *SIAM J. Comput.*, 33(6):1441–1483, 2004. doi:10.1137/S0097539703436424.

20 Amit Levi, Ramesh Krishnan S. Pallavoor, Sofya Raskhodnikova, and Nithin Varma. Erasure-resilient sublinear-time graph algorithms. *CoRR*, abs/2011.14291, 2020. arXiv:2011.14291.

21 Ramesh Krishnan S. Pallavoor, Sofya Raskhodnikova, and Nithin Varma. Improved bounds for k-connectedness testing. Unpublished manuscript, 2020.

22 Ramesh Krishnan S. Pallavoor, Sofya Raskhodnikova, and Erik Waingarten. Approximating the distance to monotonicity of Boolean functions. In *Proc. of ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1995–2009, 2020. doi:10.1137/1.9781611975994.123.

23 Michal Parnas and Dana Ron. Testing the diameter of graphs. *Random Struct. Algorithms*, 20(2):165–183, 2002. doi:10.1002/rsa.10013.

24 Michal Parnas, Dana Ron, and Ronitt Rubinfeld. Tolerant property testing and distance approximation. *J. Comput. Syst. Sci.*, 72(6):1012–1042, 2006. doi:10.1016/j.jcss.2006.03.002.

25 Sofya Raskhodnikova, Noga Ron-Zewi, and Nithin Varma. Erasures vs. errors in local decoding and property testing. In *Proc. of Innovations in Theoretical Computer Science (ITCS)*, pages 63:1–63:21, 2019. doi:10.4230/LIPIcs.ITCS.2019.63.

26 Sofya Raskhodnikova and Adam D. Smith. A note on adaptivity in testing properties of bounded degree graphs. *Electronic Colloquium on Computational Complexity (ECCC)*, 13(089), 2006. URL: http://eccc.hpi-web.de/eccc-reports/2006/TR06-089/index.html.