Comparing the strength of query types in property testing: The case of testing k-colorability*

Ido Ben-Eliezer † Tali Kaufman ‡ Michael Krivelevich § Dana Ron ¶

Abstract

We study the power of four query models in the context of property testing in general graphs (i.e., with arbitrary edge densities), where our main case study is the problem of testing k-colorability. Two query types, which have been studied extensively in the past, are *pair queries* and *neighbor queries*. The former corresponds to asking whether there is an edge between any particular pair of vertices, and the latter to asking for the i'th neighbor of a particular vertex. We show that while for pair queries, testing k-colorability requires a number of queries that is a monotone decreasing function in the average degree d, the query complexity in the case of neighbor queries remains roughly the same for every density and for large values of k. We also consider a combined model that allows both types of queries, and we propose a new, stronger, query model, related to the field of Group Testing. We give one-sided error upper and lower bounds for all the models, where the bounds are nearly tight for three of the models. In some of the cases our lower bounds extend to two-sided error algorithms.

The problem of testing k-colorability was previously studied in the contexts of dense and sparse graphs, and in our proofs we unify approaches from those cases, and also provide some new tools and techniques that may be of independent interest.

1 Introduction

Property testing [6, 12] deals with the problem of deciding whether a certain object has a prespecified property P or it is far (i.e., differs significantly) from any object that has P. Namely, the algorithm should *accept* objects that have the property, and should *reject* objects that are far from having the property with respect to some predetermined distance measure, where the algorithm is allowed a small probability of failure. The algorithm is given query access to the object, and it should make the decision after observing only a small part of the object. Thus, the main complexity measure studied in the context of property testing is the *query complexity* of the algorithm, which is normally expected to be *sublinear* in the size of the object, and the question is what exact form does this complexity take.

In this work we compare the power of different query types in the context of testing graph properties of general graphs (i.e., with arbitrary edge densities). To this end we focus on the problem of testing k-colorability for $k \geq 3$, and study the query complexity of this problem for different query types as a function of the number of vertices and the average degree of the graph.

*Preliminary version appeared in Proceedings of the 19th ACM-SIAM Symposium on Discrete Algorithms, 2008.

†School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel. E-mail: idobene@post.tau.ac.il.

‡Institute for Advanced Study, Princeton, New Jersey. E-mail:kaufmant@ias.edu.

§School of Mathematical Sciences, Tel Aviv University, Tel Aviv 69978, Israel. E-mail: krivelev@post.tau.ac.il.
Research supported in part by a USA-Israel BSF grant and by a grant from the Israel Science Foundation.

¶School of Electrical Engineering, Tel Aviv University, Tel Aviv 69978, Israel. E-mail: danar@eng.tau.ac.il.
Research supported in part by a grant from the Israel Science Foundation.

2 The Distance Measure and Query Types Studied in this Work

When defining models for property testing of graphs there are two issues to consider: the distance measure between graphs (which determines what graphs should be rejected by the testing algorithm) and the types of queries that the algorithm is allowed to make. Since we study graphs of varying edge densities and vertex degrees, we follow [10, 9] and define our distance measure with respect to the total number of edges in the graph. Namely, if n denotes the number of graph vertices and d denotes the average degree, then we say that a graph is *ϵ-far* from being k-colorable for a given $0 \leq \epsilon \leq 1$, if it is necessary to remove more than ϵdn edges so as to obtain a k-colorable graph.[1]

We consider the following types of queries where the first two have been considered in the past and the third is a new query type we introduce.

- Pair queries. These are queries of the form "Is there an edge between the pair of vertices u and v?".
- Neighbor queries. These are queries of the form: "Who is the i'th neighbor of vertex v?". If v has less than i neighbors then a special symbol is returned, and no assumption is made about the order of the neighbors of a vertex.
- Group queries. We propose a new query type that extends pair queries. These queries are of the form "Is there at least one edge between a vertex u and a set of vertices S?".

The study of group queries is partially motivated by the field of Group Testing (see, e.g., [5]), where similar queries are allowed. Problems of group testing can be found in various fields such as Statistics and Biology. Another motivation for studying group-queries is that lower bounds on the query complexity of algorithms that use group queries also apply (up to poly-logarithmic factors in n) to algorithms that use pair queries and/or neighbor queries, and hence they can be viewed as a tool for obtaining query complexity lower bounds. This follows from the (easily proven) fact that pair queries are a special case of group queries and neighbor queries can be emulated using group queries. To be precise, the emulation is for random neighbor queries (that is, the query is of the form: "Give me a uniformly selected neighbor of vertex v").

For the sake of simplicity of the presentation, we allow algorithms that perform neighbor queries and algorithms that perform group queries to perform degree queries as well. That is, they may ask for the degree of any vertex of their choice. Clearly, a degree query can be emulated by $O(\log n)$ neighbor queries (using a simple binary search). The degree of any vertex can be approximated with high probability within a constant factor using a number of group queries that is poly-logarithmic in n (and such an approximation is sufficient for our purposes).

In what follows, when we refer to the *pair query model* (respectively, *neighbor query model* and *group query model*), we mean that only pair queries (respectively, neighbor queries and group queries) are allowed. When both pair queries and neighbor queries (as well as degree queries) are allowed, then we refer to the resulting model as the *combined model*.

3 Related Work on Testing k-Colorability

Testing k-colorability has previously been studied in the pair query model for the case that the graph is dense, that is, $d = \Theta(n)$. For this case k-colorability is testable using a number of queries

[1] Another well studied distance measure is the fraction of edge modification as a function of n^2. This measure is appropriate for dense graphs (i.e., that satisfy $d = \Theta(n)$). In what can be viewed as the other extreme, where all vertices have bounded degree $d_{\max}$ (in particular, $d_{\max} = O(1)$), distance is measured with respect to $d_{\max}n$).

that is *independent* of the graph size (and polynomial in k and $1/\epsilon$ [6, 3]).[2]

Testing k-colorability has previously been studied in the neighbor query model for the case that $k = 3$ and the graph has constant maximum degree. (that is, $d = O(1)$, and furthermore, the maximum degree $d_{\max}$ is $O(1)$ as well). In this case Bogdanov et. al. [4] proved that is necessary to perform $\Omega(n)$ queries (that is, there is no algorithm with sublinear query complexity).

Testing k-colorability for $k = 2$ (i.e., testing bipartitness) has previously been studied for general graphs in the combined model [9] where it was shown that $\tilde{\Theta}(\min\{\sqrt{n}, n/d\})$ (pair and neighbor) queries are both sufficient and necessary. The proof of the lower bound in [9] implies that if only neighbor queries are allowed then $\Omega(\sqrt{n})$ queries are necessary for every value of d, and if only pair queries are allowed then $\Omega(n/d)$ queries are necessary.[3]

4 Our Results

In this work we study the power of the different types of queries when testing k-colorability of general graphs for a fixed $k \geq 3$. In previous work on testing properties of graphs, the pair query model was studied in the case of dense graphs, the neighbor query model was studied in the case of bounded-degree graphs, and for general graphs, the combined model was considered. Here we are interested in understanding how the query complexity of the problem behaves as a function of the edge density (and the number of vertices) when the algorithm is allowed to perform only one type of query, and whether there is a gain when allowing to combine query types. One motivation for this investigation is that the type of queries allowed depends on the way the graph is represented. Thus, allowing both pair queries and neighbor queries (as in the combined model) assumes that the algorithm has access both to an adjacency matrix representation (that supports pair queries) and to an incidence lists representation (that supports neighbor queries), which is not necessarily the case. Our second motivation is simply complexity theoretic: understanding the strength of each query type separately (and possibly combined) for varying edge densities.

In what follows we say that an algorithm has *one-sided error* if it always accepts graphs that are k-colorable, otherwise it has *two-sided error*. Our results are stated in terms of the dependence on n and d. With a slight abuse of notation, we write $f = \tilde{O}(g)$ (and similarly, $f = \tilde{\Omega}(g)$) if $f(x) = O(g(x)) \cdot \text{polylog}(n)$ for every x, where n is the number of vertices. In all our upper bounds the dependence on both k and $1/\epsilon$ is polynomial. The bounds are for the query complexity in the different models. We note that the running time of our algorithms may be exponential in the number of queries, but the focus of this work is only on the query complexity.

Theorem 1 *The following holds for testing k-colorability in the pair query model:*

1. *There exists a one-sided error tester that performs $\tilde{O}((\frac{n}{d})^2)$ queries.*
2. *Every one-sided error tester must perform $\Omega((\frac{n}{d})^2)$ queries.*

Theorem 2 *The following holds for testing k-colorability in the neighbor query model:*

1. *There exists a one-sided error tester that performs $O(n)$ queries.*

[2]Interestingly, the earlier work of Rödl and Duke [11] implicitly implies that k-colorability is testable using a number of queries that is independent of the graph size, but is a tower function of $1/\epsilon$.

[3]In earlier work [8, 7] it was shown that if only neighbor queries are allowed and the distance measure is with respect to $d_{\max}n$ rather than dn, then $\tilde{\Theta}(\sqrt{n})$ are both necessary and sufficient.

	Pair Queries	Neighbor Queries	Pair&Neighbor Queries	Group Queries
Upper Bound	$\tilde{O}((\frac{n}{d})^2)$	$O(n)$	$\min\{\tilde{O}((\frac{n}{d})^2), O(n)\}$	$\tilde{O}(\frac{n}{d})$
Lower Bound	$\Omega((\frac{n}{d})^2)$	$\Omega(n^{1-\frac{1}{\lceil (k+1)/2 \rceil}})$ $\Omega(n \cdot d^{-\frac{1}{\lceil k/2 \rceil - 1}})$ if $k \geq 6$	$\Omega(\frac{n}{d})$ also for 2-sided error	$\Omega(\frac{n}{d})$ also for 2-sided error

Table 1: Results for one-sided error testing of k-colorability

2. *Every tester must perform* $\Omega\left(\max\{\frac{n}{d}, \sqrt{n}\}\right)$ *queries.*

3. *Every one-sided error tester must perform* $\Omega(n^{1-\frac{1}{\lceil (k+1)/2 \rceil}})$ *queries.*

4. *Every one-sided error tester for* $k \geq 6$ *must perform* $\Omega(n \cdot d^{-\frac{1}{\lceil k/2 \rceil - 1}})$ *queries.*

Observe that for one-sided error testers in the neighbor query model, as k increases, our lower bound approaches our upper bound.

Theorem 3 *The following holds for testing k-colorability in the group query model:*

1. *There exists a one-sided error tester that performs* $\tilde{O}(\frac{n}{d})$ *queries.*

2. *Every tester must perform* $\tilde{\Omega}(\frac{n}{d})$ *queries.*

By combining Theorems 1, 2, 3 and the fact that neighbor queries can be emulated using a logarithmic number of group queries, we get the next corollary.

Corollary 4.1 *The following holds for testing k-colorability in the combined query model:*

1. *There exists a one-sided error tester that performs* $\min((\tilde{O}(\frac{n}{d})^2), O(n))$ *queries.*

2. *Every tester must perform* $\tilde{\Omega}(\frac{n}{d})$ *queries.*

The results are summarized in Table 1 and are illustrated in Figure 1.

Discussion of our results and conclusions. We next discuss the main phenomena we observe in our study of testing k-colorablity in the different query models.

- While the query complexity in the pair query model and the query complexity in the group query model are monotone decreasing functions of d, the query complexity in the neighbor query model remains roughly the same for every value of d (and large values of k).

- When comparing the pair query model to the neighbor query model in more detail we see that the query complexity in the pair query model is higher than in the neighbor query model for $d < \sqrt{n}$, while once d passes $\sqrt{n}$ it becomes lower (and continues decreasing). The extreme case is $d = \Theta(n)$, where in the pair query model the query complexity does not depend on n.[4] This coincides with the intution that neighbor queries are useful for sparse graphs and pairs queries are useful for dense graphs.

[4]In fact, when $d = \Theta(n)$ then many natural properties are testable in the pair query model using a constant number of queries (see e.g. [6, 1, 2]).

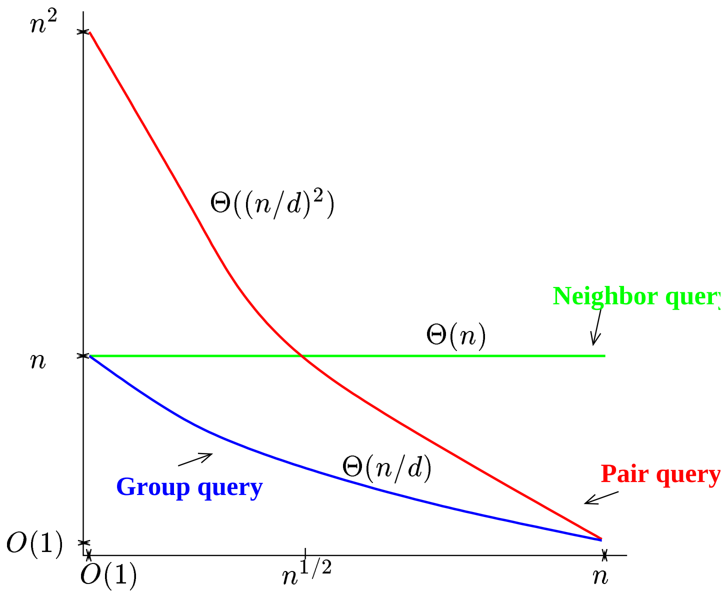

Figure 1: A schematic illustration of the query complexity for the different query types (and one-sided error). For the sake of simplicity we ignore logarithmic factors in the bounds and furthermore, for the neighbor query model we think of k being large. For the combined model we have that the lower bound on the query complexity coincides with the group query model, and the upper bound coincides with the neighbor query model until $d = n^{1/2}$ and from that point on it coincides with the pair query model.

- When comparing the group query model to the other models we observe that the query complexity in the former model is never higher (upto poly-logarithmic factors) than in the latter models. As noted previously, this holds for other graph properties provided that it is sufficient to use random neighor queries in the neighbor query model (since these can be emulated by group queries). When non-random neighbor queries are required in order to perform degree queries, then one should verify whether approximate degree queries are sufficient (since these can be emulated using group queries).

  Moreover, we note that a rather straightforward modification of the arguments of [9] shows that using group queries one can give strictly better bounds for testing bipartiteness than in the combined model. Therefore, there exist properties for which the group query model is stronger than the combined model. We also show that for the case of testing k-colorability and a certain distribution over graphs, the combined model is strictly stronger than the optimum of the pair query and the neighbor query models.

References

[1] N. Alon, E. Fischer, M. Krivelevich, and M. Szegedy. Efficient testing of large graphs. *Combinatorica*, 20:451–476, 2000.

[2] N. Alon, E. Fischer, I. Newman, and A. Shapira. A combinatorial characterization of the testable graph properties: it's all about regularity. In *Proceedings of the 38th ACM STOC*, pages 251–260, 2007.

[3] N. Alon and M. Krivelevich. Testing k-colorability. *SIAM Journal on Discrete Math*, 15(2):211–227, 2002.

[4] A. Bogdanov, K. Obata, and L. Trevisan. A lower bound for testing 3-colorability in bounded degree graphs. In *Proceedings of the 43th IEEE FOCS*, pages 93–102, 2002.

[5] D. Du and F. Hwang. *Combinatorial group testing and its applications.* World Scientific, 1993.

[6] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *JACM*, 45(4):653–750, 1998.

[7] O. Goldreich and D. Ron. A sublinear bipartite tester for bounded degree graphs. *Combinatorica*, 19(3):335–373, 1999.

[8] O. Goldreich and D. Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2):302–343, 2002.

[9] T. Kaufman, M. Krivelevich, and D. Ron. Tight bounds for testing bipartiteness in general graphs. *SIAM Journal on Computing*, 33(6):1441–1483, 2004.

[10] M. Parnas and D. Ron. Testing the diameter of graphs. *Random Structures and Algorithms*, 20(2):165–183, 2002.

[11] V. Rödl and R. Duke. On graphs with small subgraphs of large chromatic number. *Graphs Combin*, 1:91–96, 1985.

[12] R. Rubinfeld and M. Sudan. Robust characterization of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996.