

What does the local structure of a planar graph tell us about its global structure? *

Christian Sohler

Technische Universität Dortmund
christian.sohler@tu-dortmund.de

Abstract. The local k -neighborhood of a vertex v in an unweighted graph $G = (V, E)$ with vertex set V and edge set E is the subgraph induced by all vertices of distance at most k from v . The rooted k -neighborhood of v is also called a k -disk around vertex v . If a graph has maximum degree bounded by a constant d , and k is also constant, the number of isomorphism classes of k -disks is constant as well. We can describe the local structure of a bounded-degree graph G by counting the number of isomorphic copies in G of each possible k -disk. We can summarize this information in form of a vector that has an entry for each isomorphism class of k -disks. The value of the entry is the number of isomorphic copies of the corresponding k -disk in G . We call this vector *frequency vector* of k -disks. If we only know this vector, what does it tell us about the structure of G ?

In this paper we will survey a series of papers in the area of Property Testing that leads to the following result (stated informally): There is a $k = k(\epsilon, d)$ such that for any planar graph G its local structure (described by the frequency vector of k -disks) determines G up to insertion and deletion of at most ϵdn edges (and relabelling of vertices).

1 Introduction

Very large networks like social networks, the web graph, transportation networks and road maps appear in many applications. Analyzing huge networks is already a difficult task and things become even more involved when we want to analyze a collection of very large networks or learn certain concepts from it. An illustrative example is the question, if one can learn from the Facebook graph of a country, whether it is a democracy or a totalitarian state. In order to answer this question one has to design learning algorithms that extract information from very large graphs. One possible approach is to extract features from these graphs and use standard learning methods on the extracted feature vectors. In order to make this approach work, we want to extract many features from a set of very large graphs, i.e. we have a problem that is severely time-constrained. One approach to this problem is to use random sampling to approximately extract features. Such

* This paper has been published in Lecture Notes in Computer Science Volume 8634, 2014, pp. 44-49. The final version is available at http://link.springer.com/chapter/10.1007%2F978-3-662-44522-8_4

random sampling approaches can be studied using the framework of *Property Testing*.

2 Property Testing

Property Testing provides a framework to study sampling approaches to approximately decide if a given object has a property or is far away from it. The notion of "far away" is parametrized by ϵ , which typically measures the fraction of the object's description that has to be modified to obtain an object that has the studied property. The concept of Property Testing has first been formulated by Rubinfeld and Sudan in the context of program checking [12]. It has then been extended to graphs by Goldreich, Goldwasser and Ron [5]. In this paper, we consider Property Testing in the bounded degree graph model, which has been introduced by Goldreich and Ron [7]. In this model, an algorithm is given oracle access to a graph $G = (V, E)$ with vertex set $V = \{1, \dots, n\}$ and edge set E and maximum degree bounded by d . Furthermore, the algorithm is given the values n and d . It can query the oracle about the i -th neighbor of vertex j for $i \in \{1, \dots, d\}$ and $j \in \{1, \dots, n\}$ and the answer is either this neighbor or a special symbol that indicates that such a neighbor does not exist. Next we define the notion of ϵ -far.

Definition 1. *A graph is ϵ -far from a property Π in the bounded degree graph model, if one has to insert or delete more than ϵdn edges to obtain a graph that has property Π and maximum degree at most d .*

A Property Testing algorithm or property tester for a property Π must accept every graph with property Π with probability at least $3/4$ and reject every graph that is ϵ -far from Π with probability $3/4$. If a graph neither has Π nor is ϵ -far from it, the algorithm may answer arbitrarily.

The goal of Property Testing is to find algorithms that approximately decide a property in the above sense without looking at the whole input. In fact, there are many examples for Property Testing algorithms that make only a constant number of queries to the input graph (assuming ϵ to be constant). In order to study these properties we define testable graph properties as follows.

Definition 2. *A graph property Π is called testable, if there exists a function $q(\epsilon, d)$ such that for every n, d and ϵ there is an algorithm $A_{\epsilon, d, n}$ that gets as input a graph G with n vertices, makes at most $q(\epsilon, d)$ queries to G , accepts with probability at least $3/4$, if G has Π and rejects with probability at least $3/4$, if G is ϵ -far from Π .*

Note that the above notion allows to have different property testers for different values of ϵ, d and n . This is required if one wants to obtain results of a generality as presented later in this survey.

3 Property Testing of planar graphs

The first properties shown to be testable in the bounded degree graph model included connectivity, k -connectivity, cycle-freeness and being Eulerian [7]. However, no results of testable classes of properties were known. The first result in this direction was proved by Czumaj, Shapira and Sohler [3] who studied Property Testing when the input graph is restricted to be a planar graph of maximum degree at most d . Under this assumption they could prove that every hereditary property, i.e. any property that is closed under vertex removal, is testable. The property tester is very simple: It samples a set of vertices S and checks whether the subgraph induced by S has the studied property. If it does, the tester accepts and otherwise it rejects. By closedness under vertex removal, the tester accepts every hereditary graph property.

It remains to prove that the tester rejects, if the graph is ϵ -far from Π . The proof exploits the fact that every planar graph has a small vertex separator:

Theorem 1. [10] *Let $G = (V, E)$ be a planar graph with n vertices. Then V can be partitioned into three sets A, B, C such that there is no edge between the sets A and B . Furthermore, $|A|, |B| \leq \frac{2}{3} \cdot n$ and $|C| \leq 2\sqrt{2n}$.*

Repeatedly applying this theorem leads to a set of, say, at most $\epsilon n/2$ vertices whose removal partitions G into connected components of size $O(1/\epsilon^2)$. Since G has degree at most d we can achieve the same effect by removing the at most $\epsilon dn/2$ edges incident to the vertices in the separator. This implies that any graph that is ϵ -far from having a property Π is still $\epsilon/2$ -far from Π after removing this set of edges. This essentially reduces testing a property of a graph G to testing the property in a graph G that only has connected components of constant size. This implies (and here we are simplifying a bit) that there are $\Omega(n)$ connected components that do not have property Π (assuming ϵ to be a constant). If we choose our value of k sufficiently large, i.e. larger than the diameter of the connected components, then sampling a vertex inside such a component will lead to the discovery of a subgraph that contains the component (recall that the sampling is done in the original graph). By closedness under vertex removal this subgraph does not have Π and the tester rejects.

4 Testing planar graph properties

Given that many properties in planar graphs are testable there is the question if something similar holds for general graphs. Benjamini, Schramm and Shapira proved that this is indeed the case [1]. They showed that if the frequency vector of a graph is close to that of a planar graph, then this graph can be partitioned into small components by removing, say, $\epsilon dn/2$ edges. Furthermore, if a graph can be partitioned into such a set of small components, we can use arguments similar to that in [3] to prove that every property that is closed under insertion, deletion and contraction of edges is testable. Such a property is also called minor-closed. This implies that every planar and minor-closed graph property is testable (a

graph property Π is planar, if every graph that has Π is also planar). The arguments can be generalized to prove that every minor-closed graph property is testable [1].

5 Local algorithms to access a partition

Another interesting question one can ask in this context is if it is possible to get local access to a partition of a planar graph into small components that is obtained by removing at most ϵdn edges, i.e. whether for a query vertex v one can locally compute its connected component and the answer is consistent with some global partitioning. Hassidim, Kelner, Nguyen and Onak [6] introduced a remarkably simple local algorithm that gives such access. Their algorithm is a local instantiation of the following global algorithm also given in [6].

GLOBALPARTITIONING (k, ϵ)

Compute a random permutation v_1, \dots, v_n of the vertices in V

$P = \emptyset$

for $i = 1$ **to** n **do**

if v_i is still in the graph **then**

if there is a connected set $S \subset V$ with $v_i \in S$ and that has at most $\epsilon|S|$ edges that leave S **then** Let S be this set

else $S = \{v_i\}$

$P = P \cup \{S\}$

 remove S from G

return P

If we want to locally simulate the algorithm then each vertex computes a random value between 0 and 1 and the permutation is given by sorting the vertices increasingly according to these values. It is proved in [6] that in order to compute the component of a vertex v we typically only need to look at a constant size neighborhood of v and we only need to instantiate the random values for this neighborhood. This new algorithm can also be used to simplify and improve some Property Testing results.

Some more efficient constructions in terms of dependency on $1/\epsilon$ are known for more restricted classes of graphs [4,9].

6 Approximating a planar graph by its local structure

If we consider two planar graphs that are ϵ -close to being isomorphic, i.e. one has to change at most an ϵ fraction of the edges in one graph to obtain an isomorphic copy of the other graph, then their local structures will also be similar (we here think of ϵ being a very small constant). An interesting question is, if the converse is true as well. Based on the previous results, Newman and Sohler proved that this is indeed the case [11]. They proved that if two graphs on n vertices

have the same distribution of local neighborhoods, then they can be partitioned into the same set of connected components by removing at most $\epsilon dn/2$ edges in each graph. This implies that they are ϵ -close to being isomorphic. In the formal statement of the theorem below note that $f_G(k)$ denotes the normalized frequency vector of the k -disks in G , i.e. their distribution. Also, recall that a k -disk around v is the rooted subgraph induced by all vertices of distance at most k from v .

Theorem 2. *Let G_1, G_2 be two planar graphs with maximum degree at most d on n vertices. Then for every ϵ , $0 < \epsilon \leq 1$, there exists $\eta = \eta(\epsilon, \rho, d)$, $k = k(\epsilon, \rho, d)$, such that, if $|f_{G_1}(k) - f_{G_2}(k)| \leq \eta$ then G_1 is ϵ -close to (being an isomorphic copy of) G_2 .*

7 Extensions

Most of the results mentioned above extend to more general classes of graphs, i.e. to all graphs that, for every ϵ , $1 > \epsilon > 0$, can be partitioned into a set of small components of size $f(\epsilon, d)$ by removing at most ϵdn edges.

8 Open problems

There are several interesting open problems in Property Testing for sparse graphs. We will mention three of the most interesting ones.

Query complexity of planarity testing. Currently, the best Property Testing algorithm for planarity testing is from [9] and has a query complexity of $(1/\epsilon)^{O(\log 1/\epsilon)}$. Can this be improved to a polynomial? In order to prove such a result one may need to develop improved local partitioning algorithms or a different way to approach the problem.

Testable properties in expander graphs. We do not know much about the testability of properties that contain expander graph. It would be very nice to find a characterization of a large set of testable properties that contain expander graphs.

Testable properties in bounded average degree graphs. If we do not have a degree bound, most of the techniques presented in this survey do not work any more. It would be interesting to prove similar results as in the bounded degree graph model. So far, we know that bipartiteness is testable in arbitrary planar graphs [2]. Furthermore, it is known that one can test for isomorphism [8].

9 Acknowledgements

The author acknowledges the support of ERC grant 307696.

References

1. I. Benjamini, O. Schramm and A. Shapira. Every Minor-Closed Property of Sparse Graphs is Testable. *Advances in Mathematics* 223, pp. 2200-2218, 2010.
2. A. Czumaj, M. Monemizadeh, K. Onak, C. Sohler. Planar Graphs: Random Walks and Bipartiteness Testing. *FOCS*, pp. 423-432, 2011.
3. A. Czumaj, A. Shapira, C. Sohler. Testing Hereditary Properties of Nonexpanding Bounded-Degree Graphs. *SIAM Journal on Computing* 38(6): 2499-2510, 2009.
4. Edelman A, Hassidim A, Nguyen H N, Onak K (2011) An Efficient Partitioning Oracle for Bounded-Treewidth Graphs. *APPROX-RANDOM*, pp. 530-541.
5. O. Goldreich, S. Goldwasser and D. Ron. Property Testing and its Connection to Learning and Approximation. *Journal of the ACM* 45(4): 653-750, 1998.
6. A. Hassidim, J.A. Kelner, H.N. Nguyen and K. Onak. Local Graph Partitions for Approximation and Testing. *FOCS*, pp. 22-31, 2009.
7. O. Goldreich and D. Ron. Property Testing in Bounded Degree Graphs. *Algorithmica* 32(2): 302-343, 2002.
8. M. Kusumoto, Y. Yoshida. Testing Forest-Isomorphism in the Adjacency List Model. *ICALP*, pp. 763-774, 2014.
9. R. Levi, D. Ron. A Quasi-Polynomial Time Partition Oracle for Graphs with an Excluded Minor. *ICALP*, pp. 709-720, 2013.
10. R.J. Lipton and R.E. Tarjan. A separator theorem for planar graphs. *SIAM Journal on Applied Mathematics*, 36(2): 177-189, 1979.
11. I. Newman and C. Sohler. Every Property of Hyperfinite Graphs Is Testable. *SIAM Journal on Computing* 42(3): 1095-1112, 2013.
12. R. Rubinfeld, M. Sudan. Robust Characterizations of Polynomials with Applications to Program Testing. *SIAM Journal on Computing* 25(2): 252-271 , 1996.