Tel Aviv University
The Raymond and Beverly Sackler Faculty of Exact Sciences
School of Computer Science

Graph Property Testing and Related Problems

Thesis Submitted for the Degree of

"Doctor of Philosophy"

by

Asaf Shapira

Submitted to the Senate of Tel Aviv University
August 2006

Thesis Prepared Under the Supervision of

Prof. Noga Alon

Acknowledgements

First and foremost I would like to thank my advisor Prof. Noga Alon. I have greatly benefited and enjoyed our meetings and joint works in the past 5 years. I am indebted to Noga for teaching me everything I know about doing research.

I would also like to thank my other coauthors Eldar Fischer, Ilan Newman and Benny Sudakov for their collaboration in some of the results of this thesis. Special thanks also goes to Michael Krivelevich who found time to answer many of my questions and to Oded Goldreich for many (electronic) conversations on property testing. I would also like to thank my fellow graduate students Vera Asodi, Dan Hefetz, Yossi Richter, Liam Roditty and Oded Schwartz for many valuable discussions.

Special thanks to the Clore Scholars Programme for generously supporting me during the last two years of my studies and to IBM for their support during the third year of my studies.

Finally, thanks to my dear parents for their perspective on life and to my beloved wife Ravit for making it all worthwhile.

Abstract

Property testers are fast randomized algorithms for distinguishing between objects satisfying a certain property from those that are ϵ-far from satisfying it. The focus of this thesis is in testing properties of graphs. This thesis is composed of three parts:

In the first part of this thesis we study general testability results without much care how large the involved constants are as a function of the error parameter ϵ. In the first chapter we show that the entire family of hereditary properties can be tested with one-sided error. This result contains as a special case several previous results and also implies the testability of many well studied properties, which were not previously known to be testable. A few examples are the properties of being Perfect, Chordal, Interval and Ramsey. More interestingly, we use this result in order to give a characterization of the (natural) graph properties that can be tested with one sided error. One of the main open problems in the area of property testing, which was raised in the 1996 paper [75] of Goldreich, Goldwasser and Ron that initiated the study of graph property-testing, was to characterize the graph properties that can be tested with a constant number of queries. The second chapter resolves this open problem by giving a combinatorial characterization of testable properties. In the third chapter we study the relation between uniform and non-uniform property testers. We prove that there are (relatively) natural graph properties that can be tested by non-uniform testers but cannot be tested by uniform testers.

In the second part we take a "closer" look at testing certain types of properties, and try to classify the properties that can be tested with a small number of queries. We first study properties defined by a forbidden *induced* graph H. In the second chapter (of the second part) we consider the property of not containing a copy of a given fixed *directed* graph D. We also consider the question of whether allowing two-sided error testers can improve the query complexity of testing the above problems. In both cases we give (nearly) complete characterizations of the graphs H (and digraphs D) for which the corresponding problems can be tested with a small number of queries. We also show that two-sided error testers cannot be efficient in the cases were there is no efficient one-sided error tester. The results of this part resolve several open problems that were raised by Alon [1].

In the third part we study algorithmic results that have some connection to the area of property testing. More specifically we study the following meta problem: given a graph G how well can we approximate the number of edges that need to be removed from G in order to make it satisfy a monotone graph property $\mathcal{P}$. We first show that for any monotone property $\mathcal{P}$ and for every $\epsilon > 0$, this quantity can be approximated in linear time to within an additive error of ϵn^2. A natural question is whether it is possible to obtain a better approximation in polynomial time. The second result gives a precise characterization of the monotone properties for which one can approximate the number of necessary edge deletions within an additive error of $n^{2-\delta}$. This characterization asserts that if there is a bipartite graph that does not satisfy $\mathcal{P}$ then such an algorithm is trivial, and in the other case the problem is NP-hard. This characterization resolves (in a strong form) a question raised by Yannakakis [115] in 1981.

Contents

Introduction

The meta problem in the area of property testing is the following: given a combinatorial structure S, distinguish between the case that S satisfies some property $\mathcal{P}$ and the case that S is ϵ-far from satisfying $\mathcal{P}$. Roughly speaking, a combinatorial structure is said to be ϵ-far from satisfying some property $\mathcal{P}$ if an ϵ-fraction of its representation should be modified in order to make S satisfy $\mathcal{P}$. The main goal is to design randomized algorithms, which look at a very small portion of the input, and using this information distinguish with high probability between the above two cases, namely, algorithms that will accept graphs satisfying $\mathcal{P}$ with high probability and will also reject those that are ϵ-far from satisfying it with high probability. Such algorithms are called *property testers* or simply *testers* for the property $\mathcal{P}$. Preferably, a tester should look at a portion of the input whose size is a function of ϵ only. Blum, Luby and Rubinfeld [32] were the first to formulate a question of this type, and the general notion of property testing was first formulated by Rubinfeld and Sudan [109], who were interested in studying various algebraic properties such as linearity of functions.

The main focus of this thesis is the testing of properties of graphs[1]. More specifically, we focus on property testing in the dense graph model as defined in [75]. In this case a graph G is said to be *ϵ-far* from satisfying a property $\mathcal{P}$, if one needs to add/delete at least ϵn^2 edges to G in order to turn it into a graph satisfying $\mathcal{P}$. A tester for $\mathcal{P}$ should distinguish with high probability, say 2/3, between the case that G satisfies $\mathcal{P}$ and the case that G is ϵ-far from satisfying $\mathcal{P}$. Here we assume that the tester can query some oracle whether a pair of vertices, i and j, are adjacent in the input graph G. In what follows we will say that a tester for a graph property $\mathcal{P}$ has *one-sided error* if it accepts every graph satisfying $\mathcal{P}$ with probability 1 (and still rejects those that are ϵ-far from $\mathcal{P}$ with probability at least 2/3). If the tester may reject graphs satisfying $\mathcal{P}$ with non-zero probability then it is said to have *two-sided error*. The following notion of efficient testing will be the main focus of this thesis:

Definition (Testable) *A graph property $\mathcal{P}$ is* testable *if there is a randomized algorithm T, that can distinguish with probability* 2/3 *between graphs satisfying $\mathcal{P}$ and graphs that are ϵ-far from satisfying $\mathcal{P}$, while making a number of edge queries which is bounded by some function $q(\epsilon)$ that is independent of the size of the input.*

[1] A property of n-vertex graphs is simply a family of n-vertex graphs that is closed under isomorphism

Results with applications in property-testing date back to the 70's. It was implicitly proved by Ruzsa and Szemerédi [110] that triangle-freeness is testable, and Rödl and Duke [105] implicitly showed that k-colorability is testable. The modern study of the notion of testability for combinatorial structures, and mainly the dense graph model, was introduced in the seminal paper of Goldreich, Goldwasser and Ron [75]. In that paper it was shown that several well studied graph properties such as k-colorability, having a large cut and having a large clique are all testable. Graph property testing has also been studied in the *bounded-degree* model [76], and the newer *general density* model [101]. We note that in these models a property is usually said to be testable if the number of queries is $o(n)$. Following [75, 32, 109] property testing was studied in various other contexts such as boolean functions [8, 62, 63, 103], geometric objects [3, 46] and algebraic structures [32, 68, 30]. See the surveys [59, 107] for additional results and references.

We finally note that throughout this thesis we will frequently deal with two types of graph properties.

Definition (Monotone Graph Properties) *Graph property $\mathcal{P}$ is* monotone *if it is closed under removal of vertices and edges. Equivalently, $\mathcal{P}$ is closed under taking subgraphs.*

Definition (Hereditary Graph Properties) *Graph property $\mathcal{P}$ is* hereditary *if it is closed under removal of vertices. Equivalently $\mathcal{P}$ is closed under taking induced subgraphs.*

Standard examples of monotone properties are k-colorability and being H-free for some fixed graph H (e.g. triangle). Clearly any monotone property is also hereditary. Standard examples of hereditary (non-monotone) properties are being Perfect, Chordal and induced H-free for some fixed graph H.

Part I: General Testability Results

In the first part of the thesis we aim at giving general testability results without much care about the number of queries the algorithm performs, as long as it is bounded by a function of ϵ as required by the definition of a testable property. The main focus of property testing and in particular graph property testing is in identifying the testable graph properties. Obtaining a characterization of the testable graph properties was considered the main open problem of graph property testing.

A natural strategy toward obtaining a characterization of the testable graphs was to either prove the testability/non-testability of general families of graph properties or to obtain characterizations for special cases of testers. The main result of [75] was that a general family of so called "partition-problems" are all testable. These include the properties of being k-colorable, having a large cut and having a large clique. Goldreich and Trevisan [77] gave a characterization of the partition-problems that can be tested with *1-sided* error. Czumaj and Sohler [47] studied property testing via the framework of *abstract combinatorial programs* and gave certain characterizations of the testable properties that fit this framework.

Alon, Fischer, Krivelevich and Szegedy [6] obtain general testability results in terms of *logical* properties of a language. More specifically, it was shown in [6] that every first order graph-property of type $\exists\forall$ (see [6]) is testable, while there are first-order graph properties of type $\forall\exists$ that are not testable. The main technical result of [6] was that certain abstract colorability properties are all testable. These results were generalized in [60]. Finally, [77] following [6], proved that a tester may be assumed to be non-adaptive (see Theorem 2.14), and [64] proved that if a graph property is testable then it is also possible to estimate how far is a given graph from satisfying the property (see Theorem 2.27). These last two results are key ingredients in this part of the thesis.

Given the previous mentioned general testability results, a natural question is what makes a combinatorial property testable. In particular, characterizing the testable graph properties was considered one of the main open problems in the area of property testing, and was raised already in the 1996 paper of Goldreich, Goldwasser and Ron [75], see also [74], [31] and [77]. As many of the partition problems that were shown to be testable are closed under removal of edges, a natural possibility was to show that any graph property that is closed under removal of edges is testable. Further supporting evidence of this fact was given by the (implicit) result of [4] that for any fixed graph H the property of being H-free, which is also closed under removal of edges, is testable. Regretfully, it was shown in [77] that there are graph properties that are closed under removal of edges and cannot be tested with $o(n^2)$ queries.

Hereditary Properties and Testing with One-Sided Error (Chapter 1)

Our first result in this chapter identifies a large and natural family of properties that are all testable, by showing that any hereditary graph property is testable. This result is obtained using a novel application of Szemerédi's regularity-lemma. This general testability result contains as a special case many of the previous results about testing graph properties with one-sided error. These include the results of [75] and [7] about testing k-colorability, the characterization of [77] of the graph-partitioning problems that are testable with one-sided error, the induced vertex colorability properties of [6], the induced edge colorability properties of [60], as well as a transformation from two-sided to one-sided error testing [77]. More importantly, as a special case of the main result, we infer that some of the most well studied graph properties, both in graph theory and computer science, are testable with one-sided error. Some of these properties are the well known graph properties of being Perfect, Chordal, Interval, Comparability, Permutation and more. None of these properties was previously known to be testable.

The second result in this chapter is a solution of a problem closely related to that of characterizing the testable graph properties; call a property tester *oblivious* if its decisions are independent of the size of the input graph. We show that a graph property $\mathcal{P}$ has an oblivious one-sided error tester, **if and only if** $\mathcal{P}$ is (semi) *hereditary*. We stress that any "natural" property that can be tested (either with one-sided or with two-sided error) can be tested by an oblivious tester. In particular, all the testers studied thus far in the literature were oblivious. This result can thus be considered as a precise characterization

of the natural graph properties, which are testable with one-sided error.

References: The results of this chapter appeared as:

- N. Alon and A. Shapira, Every monotone graph property is testable, Proc. of the 37^{th} Annual Symp. on Theory of Computing **(STOC)**, 2005, 128-137. Also, **SIAM J. on Computing, Special Issue of STOC'05**, to appear.

- N. Alon and A. Shapira, A characterization of the (natural) graph properties testable with one-sided error, Proc. of the 46^{th} Annual IEEE Symp. on Foundations of Computer Science **(FOCS)** 2005, 429-438. Also, **SIAM J. on Computing, Special Issue of FOCS'05**, to appear.

Szemerédi Partitions and Two-Sided Error Testers (Chapter 2)

The results of the previous chapter give a nearly complete characterization of the properties that can be tested with one-sided error. In this chapter we consider the most general notion of property testers, namely those that may have two-sided error, and obtain a characterization of the testable graph properties. We thus resolve an open problem which was first raised in the 1996 paper of Goldreich, Goldwasser and Ron [75] that initiated the study of graph property-testing. A common thread in all the recent results concerning the testing of dense graphs, including the results of the previous chapter, is the use of Szemerédi's regularity lemma and some of its variants. The characterization we obtain in this chapter shows that in some sense this is not a coincidence. Our first result is that the property defined by having any given Szemerédi-partition is testable with a constant number of queries. Our second and main result is a purely combinatorial characterization of the graph properties that are testable with a constant number of queries. This characterization (roughly) says that a graph property $\mathcal{P}$ can be tested with a constant number of queries **if and only if** testing $\mathcal{P}$ can be reduced to testing the property of satisfying one of finitely many Szemerédi-partitions. This means that in some sense, testing for Szemerédi-partitions is as hard as testing any testable graph property. This characterization also gives an intuitive explanation as to what makes a graph property testable.

References: The results of this chapter appeared as:

- N. Alon, E, Fischer, I. Newman and A. Shapira, A combinatorial characterization of the testable graph properties: it's all about regularity, Proc. of the 38^{th} Annual Symp. on Theory of Computing **(STOC)** 2006, 251-260. Also, invited to **SIAM J. on Computing, Special Issue of STOC'06**.

Uniform vs Non-uniform Property Testing (Chapter 3)

In this chapter we consider the following seemingly rhetorical question: Is it crucial for a property-tester to know the error parameter ϵ in advance? Previous papers dealing with

various testing problems, suggest that the answer may be no, as in these papers there was no loss of generality in assuming that ϵ is given as part of the input, and is not known in advance. The main result in this chapter, however, is that it is possible to separate a natural model of property testing in which ϵ is given as part of the input from the model in which ϵ is known in advance (without making any hardness-type assumptions). To this end, we construct a graph property $\mathcal{P}$ which satisfies the following:

(i) There is no tester for $\mathcal{P}$ accepting ϵ as part of the input, whose number of queries depends only on ϵ.

(ii) For any fixed ϵ, there is a tester for $\mathcal{P}$ (that works only for that specific ϵ), which makes a constant number of queries.

Interestingly, we manage to construct a separating property $\mathcal{P}$, which is combinatorially natural as it can be expressed in terms of forbidden subgraphs and also computationally natural as it can be shown to belong to $coNP$.

The main tools in this chapter are efficiently constructible graphs of high girth and high chromatic number, a result about testing monotone graph properties, as well as basic ideas from the theory of recursive functions. Of independent interest is a *precise characterization* of the monotone graph properties that can be tested with ϵ being part of the input, which we obtain as one of the main steps of the chapter. Somewhat surprisingly, this characterization relies on the recursiveness of a certain graph functional that seems irrelevant to property-testing.

References: The results of this chapter were submitted for publication as:

- N. Alon and A. Shapira, A separation theorem in property-testing.

Potpourri (Chapter 4)

In this chapter we include some additional results that did not fit the previous chapters of this part of the thesis. Two of the results that we prove are that for any function f there exists a monotone graph property that cannot be tested with one-sided error using fewer than $f(\epsilon)$ queries, and a compactness-type result stating that if a graph is far from satisfying an infinite family of hereditary properties then it must also be far from satisfying one of these properties.

Part II: On The Possibility of Small Query Complexity

The results obtained in the first chapter of this thesis give general positive results concerning various graph properties. However, the bounds they guarantee are given by extremely fast growing functions of the error parameter ϵ. It is thus natural to investigate for which properties can one guarantee that the query complexity will be upper bounded by a function of ϵ that grows "relatively slowly". As is common in computer-science a natural family

of relatively slow growing functions is polynomials in $1/\epsilon$. Regretfully, we cannot give a characterization of the graph properties that can be tested with $poly(1/\epsilon)$ queries. We thus look at specific families of properties and try to give characterization within these restricted families. Alon [1] initiated this line of research, by considering the properties $\mathcal{P}_H$ of being H-free. It was shown in [1] that $\mathcal{P}_H$ has a one-sided error tester with query complexity $poly(1/\epsilon)$ if and only if H is bipartite. Note that the family of properties $\mathcal{P}_H$ is a subclass of the monotone graph properties, namely, the monotone properties that can be expressed in terms of a single forbidden subgraph.

In this chapter we consider other natural families of graph properties and obtain characterizations within them. The proofs of this part of the thesis combine combinatorial, graph theoretic and probabilistic arguments with results from additive number theory.

Testing Induced Subgraphs (Chapter 5)

In the first chapter of this part of the thesis we consider the properties of being induced H-free, that are denoted for short by $\mathcal{P}^*_H$. Note that the family of properties $\mathcal{P}^*_H$ is a subclass of the hereditary graph properties, namely, the hereditary properties that can be expressed in terms of a single forbidden induced subgraph. Let G be a graph on n vertices, H be a graph on h vertices, and suppose that G is ϵ-far from satisfying $\mathcal{P}^*_H$. It was shown in [6] that in this case G contains at least $f(\epsilon, h)n^h$ induced copies of H, where $1/f(\epsilon, h)$ is an extremely fast growing function in $1/\epsilon$, that is independent of n (the fourth function in the Ackerman Hierarchy, which is a tower of towers of exponents). As a consequence, it follows that for every H, the property $\mathcal{P}^*_H$ is testable with one-sided error. For some graphs, however, there are obviously much more efficient property testers than the ones guaranteed by the above general result. For example, for the case of H being an edge, there is obviously a one-sided error property tester for $\mathcal{P}_H = \mathcal{P}^*_H$, whose query complexity is $\Theta(1/\epsilon)$.

A natural question, raised by Alon in [1], is to decide for which graphs H the function $1/f(\epsilon, H)$ can be bounded from above by a polynomial in $1/\epsilon$. An equivalent question is for which graphs H, can one design a one-sided error tester for testing $\mathcal{P}^*_H$, whose query complexity is polynomial in $1/\epsilon$. In this chapter we settle this question almost completely by showing that, quite surprisingly, for any graph other than the paths of lengths 1, 2 and 3, the cycle of length 4, and their complements, no such property tester exists. We further show that a similar result also applies to the case of directed graphs, thus answering a question raised in [11]. We finally show that the same results hold even in the case of two-sided error property testers.

References: The results of this chapter appeared as:

- N. Alon and A. Shapira, A characterization of easily testable induced subgraphs, Proc. of the 15^{th} Annual ACM-SIAM Symposium on Discrete Algorithms **(SODA)**, (2004), 935-944. Also, **Combinatorics, Probability and Computing**, 15 (2006), 791-805.

Testing Directed Subgraphs (Chapter 6)

In the second chapter of this part of the thesis we consider the properties of being H-free, that are denoted for short by $\mathcal{P}_H$, were H is a fixed directed graph (digraph for short). We say that a graph satisfies $\mathcal{P}_H$ if it is H-free, that is, if it does not contain any (not necessarily *induced*) copy of H. Let G be a graph on n vertices and suppose that G is ϵ-far from satisfying $\mathcal{P}_H$. We first show that in this case G contains at least $f(\epsilon, H)n^h$ copies of H. This is proved by establishing a directed version of Szemerédi's regularity lemma, and implies that for every H, property $\mathcal{P}_H$ is testable with one-sided error.

As is common with applications of the undirected regularity lemma, here too the function $1/f(\epsilon, H)$ is an extremely fast tower-type function in ϵ. We therefore further prove a precise characterization of all the digraphs H, for which $f(\epsilon, H)$ has a polynomial dependency on ϵ. This implies a characterization of all the digraphs H, for which $\mathcal{P}_H$ has a one sided error property tester, whose query complexity is polynomial in $1/\epsilon$. We further show that the same characterization also applies to two-sided error property testers as well. A special case of this result settles an open problem raised by Alon in [1]. Interestingly, it turns out that if $\mathcal{P}_H$ has a polynomial query complexity, then there is a two-sided ϵ-tester for $\mathcal{P}_H$ that samples only $O(1/\epsilon)$ vertices, whereas any one-sided tester for $\mathcal{P}_H$ makes at least $(1/\epsilon)^{d/2}$ queries, where d is the average degree of H. We also show that the complexity of deciding if for a given directed graph H, $\mathcal{P}_H$ has a polynomial query complexity, is NP-complete, marking an interesting distinction from the case of undirected graphs, where the corresponding problem can be solved in polynomial time.

For some special cases of directed graphs H, we describe very efficient one-sided error property-testers for testing $\mathcal{P}_H$. As a consequence we conclude that when H is an undirected bipartite graph, we can give a one-sided error property tester with query complexity $O((1/\epsilon)^{h/2})$, improving the previously known upper bound of $O((1/\epsilon)^{h^2})$.

References: The results of this chapter appeared as:

- N. Alon and A. Shapira, Testing subgraphs in directed graphs, Proc. of the 35^{th} Annual Symp. on Theory of Computing **(STOC)** 2003, 700–709. Also, **J. of Comp. System Sciences, Special Issue of STOC'03**, 69 (2004), 354-382.

Part III: Algorithmic Results Related to Property Testing

Additive Approximation for Edge Deletion Problems (Chapter 7)

The topic of this chapter is graph modification problems, namely problems of the type: given a graph G, find the smallest number of modifications that are needed in order to turn G into a graph satisfying property $\mathcal{P}$. The main two types of such problems are the following; in *node modification* problems, one tries to find the smallest set of vertices, whose removal turns G into a graph satisfying $\mathcal{P}$, while in *edge modification* problems, one tries to find the smallest number of edge deletions/additions, which turn G into a graph satisfying $\mathcal{P}$. In this chapter we will focus on edge-modification problems. Given a graph property

$\mathcal{P}$ we denote by $E'_{\mathcal{P}}(G)$ the smallest number of edge modifications needed to turn G into a graph satisfying $\mathcal{P}$. Note, that when trying to turn a graph into one satisfying a monotone property we will only use edge deletions. Therefore, in these cases the problem is sometimes called *edge-deletion* problem.

Background and motivation: Graph modification problems are well studied computational problems. In 1979, Garey and Johnson [72] mentioned 18 types of vertex and edge modification problems. Graph modification problems were extensively studied as these problems have applications in several fields, including Molecular Biology and Numerical Algebra. In these applications a graph is used to model experimental data, where edge modifications correspond to correcting errors in the data: adding an edge means correcting a false negative, while deleting an edge means correcting a false positive. Computing $E_{\mathcal{P}}(G)$ for appropriately defined properties $\mathcal{P}$ have important applications in physical mapping of DNA (see [42], [73] and [79]). Computing $E_{\mathcal{P}}(G)$ for other properties arises when optimizing the running time of performing Gaussian elimination on a sparse symmetric positive-definite matrix (see [108]). Other modification problems arise as subroutines for heuristic algorithms for computing the largest clique in a graph (see [114]). Some edge modification problems also arise naturally in optimization of circuit design [50]. We briefly mention that there are also many results about *vertex* modification problems, notably that of Lewis and Yannakakis [93], who proved that for any nontrivial hereditary property $\mathcal{P}$, it is NP-hard to compute the smallest number of vertex deletions, which turn a graph into one satisfying $\mathcal{P}$.

The main results presented in this chapter, give a nearly complete answer to the hardness of additive approximations of the edge-deletion problem for monotone graph properties.

An Algorithm for Any Monotone Property: In the first part of this chapter we prove that for any fixed $\epsilon > 0$ and any monotone property $\mathcal{P}$, there is a deterministic algorithm, which given a graph $G = (V, E)$ of size n, approximates $E'_{\mathcal{P}}(G)$ in linear time $O(|V| + |E|)$ to within an additive error of ϵn^2. This result is obtained via a novel structural graph theoretic technique. One of the applications of this technique (roughly) yields that every graph G, can be approximated by a small weighted graph W, in such a way that $E'_{\mathcal{P}}(G)$ is approximately the optimal solution of a certain "related" problem that we solve on W. This new technique, which may very well have other algorithmic and graph-theoretic applications, applies a result of Alon, Fischer, Krivelevich and Szegedy [6], which is a strengthening of Szemerédi's Regularity Lemma [112]. We then use an efficient algorithmic version of the regularity lemma, which also implies an efficient algorithmic version of the result of [6], in order to transform the existential structural result into the algorithm stated above. Our techniques also allow us to obtain a similar randomized algorithm for estimating $E'_{\mathcal{P}}(G)$. This algorithm is simpler and somewhat more natural than a randomized algorithm for this problem that was independently obtained by Fischer and Newman [64].

On the Possibility of Better Approximation: Given the above general algorithmic result, a natural question is for which monotone properties one can obtain better additive

approximations of $E'_{\mathcal{P}}$. The second main result essentially resolves this problem by giving a precise characterization of the monotone graph properties for which such approximations exist. This characterization states that if there is a bipartite graph that does not satisfy $\mathcal{P}$, then there is a $\delta > 0$ for which it is possible to approximate $E'_{\mathcal{P}}$ to within an additive error of $n^{2-\delta}$ in polynomial time. On the other hand, if all bipartite graphs satisfy $\mathcal{P}$, then for any $\delta > 0$ it is NP-hard to approximate $E'_{\mathcal{P}}$ to within an additive error of $n^{2-\delta}$.

While the proof of the first (positive) case is relatively simple, the proof of the second (negative) case requires several new ideas and involves tools from Extremal Graph Theory together with spectral techniques. Interestingly, prior to this work it was not even known that computing $E'_{\mathcal{P}}$ *precisely* for these properties is NP-hard. We thus answer (in a strong form) a question of Yannakakis [115], who asked in 1981 if it is possible to find a large and natural family of graph properties for which computing $E'_{\mathcal{P}}$ is NP-hard.

References: The results of this chapter appeared as:

- N. Alon, A. Shapira and B. Sudakov, Additive approximation for edge-deletion problems, Proc. of the 46^{th} Annual IEEE Symp. on Foundations of Computer Science **(FOCS)** 2005, 419-428. Also, **Annals of Mathematics**, to appear.

Part I

General Testability Results

Chapter 1

Hereditary Properties and One-Sided Error Testers

1.1 The Main Results

1.1.1 Every hereditary property is testable

As we have discussed in the introduction of the thesis, previously there were many separate results concerning testable graph properties. Our first goal in this chapter is to prove a general positive testability result that will include the previous results as a special case, and will also imply the testability of new properties. We will then use this result in order to obtain a characterization of the (natural) graph properties that can be tested with one-sided error. The following is the main technical result of this chapter.

Theorem 1.1. *Every hereditary graph property is testable with one-sided error.*

The proof of Theorem 1.1 relies on a novel application of a variant of Szemerédi's regularity lemma, proved by Alon, Fischer, Krivelevich and Szegedy [6]. We believe that our application of this lemma may be useful for attacking other problems. The main idea of this application are described in details in Section 1.3.

As we will see later, the testing algorithms we design for a given hereditary property $\mathcal{P}$, simply sample a set of vertices S and accept if and only if the graph induced by S satisfies $\mathcal{P}$. This immediately implies that these testers have one-sided error. Of course, the main difficulty lies in proving that if the input is ϵ-far from satisfying $\mathcal{P}$ then the graph induced by a large enough S (but only large enough as a function of ϵ) will not satisfy $\mathcal{P}$ with high probability.

We note that besides certain partition properties such as having a large cut and having a large clique, which were proved to be testable with two-sided error in [75], essentially any graph property that was studied in the literature is hereditary. Thus, Theorem 1.1 combined with the graph partition problems of [75] imply the testability of (nearly) any

natural graph property[1]. To demonstrate the generality of Theorem 1.1, we use it to infer that many graph properties, which prior to this work were not known to be testable, are in fact testable with one-sided error. These include the following hereditary properties:

- **Perfect Graphs:** A graph G is perfect if for every *induced* subgraph of G, G', the chromatic number of G' equals the size of the largest clique in G'.
- **Chordal Graphs:** A graph is chordal if it contains no *induced* cycle of length at least 4.
- **Interval Graphs:** A graph G on n vertices is an interval graph if there are closed intervals on the real line $I_1, \ldots, I_n$ such that $(i,j) \in E(G)$ if and only if $I_i \cap I_j \neq \emptyset$.
- **Ramsey Graphs:** A graph G is ramsey if there is a 2-coloring of its edges with no monochromatic triangle.
- **Circular-Arc Graphs:** A graph G on n vertices is a circular-arc graph if there are closed intervals on a cycle $I_1, \ldots, I_n$ such that $(i,j) \in E(G)$ if and only if $I_i \cap I_j \neq \emptyset$.
- **Comparability Graphs:** A graph G is a comparability graph if its edges can be oriented such that if there is a directed edge from i to j and from j to k, then there is one from i to k.
- **Permutation Graphs:** A graph G on n vertices is a permutation graph if there is a permutation σ of $\{1, \ldots, n\}$ such that $(i,j) \in E(G)$ iff (i,j) is an inversion under σ.
- **Asteroidal Triple-Free Graphs:** G is asteroidal triple-free if it contains no independent set of 3 vertices such that each pair is joined by a path that avoids the neighborhood of the third.
- **Split Graphs:** G is a split graph if $V(G)$ can be split into a clique and an independent set.

Another abstract family of hereditary graph properties, which have been extensively studied, are the so called *intersection graph properties*. In this case we fix a certain "type" T of sets and say that a graph G on n vertices has the intersection property defined by T, if there are n sets $S_1, \ldots, S_n$ of type T, such that vertices i and j are connected in G if and only if $S_i \cap S_j \neq \emptyset$. For example, the property of being a d-Box (see [41] and its references) is obtained by letting the "type" of the sets be axis parallel boxes in R^d. See the monograph [97] for more information and examples of intersection graph properties.

It is clear that the above surveyed properties are some of the most well-studied properties in graph-theory as well as in theoretical and applied computer-science. These properties also arise naturally in Chemistry, Biology, Social Sciences, Statistics as well as in many other areas. See [78], [97], [104] and their references, where other hereditary properties and their applications are also discussed.

[1] A natural graph property that is not testable, is the property of being isomorphic to a specific graph H, where H is a "complex" enough graph. See [61] and Chapter 2 for more details.

To further convey the reader of the power of Theorem 1.1 we mention that it immediately implies, for example, that for every ϵ there is $c = c(\epsilon)$, such that if a graph G is ϵ-far from being Chordal then G contains an **induced** cycle of length at most c, and that similar results hold for any other hereditary property. This is non-trivial as it is not clear a priori that there is no graph that is, say, $\frac{1}{100}$-far from being Chordal and yet contains only induced cycles of length at least, say, $\Omega(\log n)$. Put in other way, if G has the property that all its induced subgraphs of size $c = c(\epsilon)$ are chordal, then G is ϵ-close to being Chordal. This gives a strong connection between the *local* properties of a graph and its *global* properties. In fact, we can show that an analogous result holds for any graph property, see Theorem 4.6.

1.1.2 Oblivious testing with one-sided error

By a result of [6] and [77], it is possible to assume that a property tester works by making its queries non-adaptively. In other words, the tester first picks a random subset of vertices S, and then continues without making additional queries. Inspecting previous results on property-testing, motivates the following notion of a slightly more restricted tester, which works while being "oblivious" to the size of the input[2].

Definition 1.2. (Oblivious Tester) *A tester (one-sided or two-sided) is said to be* oblivious *if it works as follows: given ϵ the tester computes an integer $Q = Q(\epsilon)$ and asks an oracle for a subgraph induced by a set of vertices S of size Q, where the oracle chooses S randomly and uniformly from the vertices of the input graph. If Q is larger than the size of the input graph then the oracle returns the entire graph. The tester then accepts or rejects (possibly randomly) according to ϵ and the graph induced by S.*

Note, that by insisting that the oracle chooses the set of vertices S, an oblivious tester indeed operates without knowing the size of the input, because if the tester had to choose S then it would have to know the size of the input graph in order to specify a vertex of the graph. We believe that the above definition captures the essence of property testing in the dense graph model as essentially all the testers that have been analyzed in this model were in fact oblivious, or could trivially be turned into oblivious testers. Even the testers for properties such as having an independent set of size $\frac{1}{2}n$ or a cut with at least $\frac{1}{8}n^2$ edges (see [75]), whose definition involves the size of the graph, have oblivious testers. The reason is simply that these properties can easily be expressed without using the size of the graph. For example, in order to test if a graph has a cut with at least $\frac{1}{8}n^2$ edges one can sample some $Q = Q(\epsilon)$ vertices and accept the input if and only if the graph induced on the sample has a cut of size at least $(\frac{1}{8} - \frac{\epsilon}{2})Q^2$ (of course, one needs to prove that this sampling scheme indeed works, see [75]). Another family of graph properties for which we can confine ourselves to oblivious testers is the family of hereditary properties, which is shown to be testable by an oblivious tester in the present work. We finally note that most "applications"

[2]The tester implied by the results of [77] and [6] may use the size of the input in order to determine both the query complexity and in order to make its decisions

of property-testing (see [59] and [107]) involve testing properties of huge networks such as the Internet, whose size is anyway unknown.

Observe, that there are two restrictions that the above definition imposes on an oblivious tester. The first is that it cannot use the size of the input in order to determine the size Q, of the sample of vertices. In other words, Q is only a function of ϵ and not a function of ϵ and n. The reader should note that a tester for a testable graph property (as defined in the introduction of this thesis) may have a query complexity that is *bounded* by a function of ϵ but one that *depends* on the size of the graph (e.g. $Q(\epsilon, n) = 1/\epsilon + (-1)^n$). Though this seems like an annoying technicality, we prove in Chapter 3 that this subtlety may have non-trivial ramifications. The second, seemingly more severe, restriction on an oblivious tester is that it cannot use the size of the input in order to make its decisions after the subgraph induced on the set S of Q vertices has been obtained. One can easily "cook" graph properties that cannot be tested by an oblivious tester. However, these properties are somewhat non-natural. One example out of many is the following property, which we denote by $\mathcal{P}'$: A graph on an even number of vertices satisfies $\mathcal{P}'$ if and only if it is bipartite, while a graph on an odd number of vertices satisfies $\mathcal{P}'$ if and only if it is triangle-free. A tester for $\mathcal{P}'$ clearly must use the size of the input in order to make its decision regarding the graph induced by the sample.

We now turn to the main result of this chapter, which gives a characterization of the graph properties that can be tested with 1-sided error by an oblivious tester. Intuitively, in order to test a property with 1-sided the tester must "find" some kind of proof that the input does not satisfy the property. Of course the graph itself is such a proof, but as we confine ourselves to testers whose number of queries is independent of the size of the input, the tester must find a *small* proof of this fact. For hereditary properties, such proofs exists, and are in fact (relatively) abundant. This is the main idea behind our algorithm for testing hereditary properties, see Lemma 1.12. A natural question is if other non-hereditary properties have such small proofs. For example, having a clique of size $\frac{1}{2}n$ obviously does not have such small proofs. The reason is that for any fixed graph C there are graphs that contain C as an induced subgraph and have a clique of size $\frac{1}{2}n$, and graphs that contain C as an induced subgraph and are far from having a clique of size $\frac{1}{2}n$. In [77] it was shown that when considering the partition-problems of [75], which contain the clique property as a special case, then non-hereditary partition properties cannot be tested with 1-sided error. For general properties the situation is much more involved. However, considering only oblivious testers enables us to precisely characterize the graph properties, which are testable with one-sided error. To state this characterization we need the following definition:

Definition 1.3. (Semi-Hereditary) *A graph property $\mathcal{P}$ is semi-hereditary if there exists a hereditary graph property $\mathcal{H}$ such that the following holds:*

1. *Any graph satisfying $\mathcal{P}$ also satisfies $\mathcal{H}$.*
2. *For any $\epsilon >$ there is an $M(\epsilon)$, such that any graph of size at least $M(\epsilon)$, which is ϵ-far from satisfying $\mathcal{P}$, contains an induced subgraph, which does not satisfy $\mathcal{H}$* [3].

[3] As $\mathcal{H}$ is hereditary, an equivalent and simpler condition is that G itself does not satisfy $\mathcal{P}$. However, the

Clearly, any hereditary graph property $\mathcal{P}$ is also semi-hereditary because we can take $\mathcal{H}$ in the above definition to be $\mathcal{P}$ itself. In simple words, a semi-hereditary property $\mathcal{P}$ is obtained by taking a hereditary graph property $\mathcal{H}$, and removing from it a (possibly infinite) set of graphs. This means that the first item in Definition 1.3 is satisfied. As there are graphs not satisfying $\mathcal{P}$ that do satisfy $\mathcal{H}$ these graphs do not contain any induced subgraph that does not satisfy $\mathcal{H}$ (because $\mathcal{H}$ is hereditary). The only restriction, which is needed in order to get item 2 in Definition 1.3, is that $\mathcal{P}$ will be such that for any $\epsilon > 0$ there will be only finitely many graphs that are ϵ-far from satisfying it, and yet contain no induced subgraph that does not satisfy $\mathcal{H}$.

We are now ready to state the main result of this chapter.

Theorem 1.4. *A graph property $\mathcal{P}$ has an oblivious one-sided error tester* **if and only if** *$\mathcal{P}$ is semi-hereditary.*

Returning to the graph property $\mathcal{P}'$ discussed above, note that by Theorem 1.1 this property, which is not semi-hereditary, can be tested with one-sided error by a non-oblivious tester. Therefore, it is not the case that a graph property is testable if and only if it is semi-hereditary. However, if we disregard this and other non-natural graph properties then we may assume that in order to test them we can confine ourselves to oblivious testers. Theorem 1.4 can thus be considered as a *precise characterization* of the natural graph properties which are testable with one-sided error. We believe that it may be very interesting to further study property-testing via the framework of oblivious testers, see Section 1.5.

1.1.3 Comparison to previous results

We next survey the previous results on graph property-testing, which were shown to be testable with one-sided error. As all these properties are hereditary, their testability with one-sided error follows as a special case of Theorem 1.1.

- **H-free:** For every fixed graph H let $\mathcal{P}_H$ be the property of not containing a copy of H, and let $\mathcal{P}^*_H$ be the property of not containing an induced copy of H. The property $\mathcal{P}_H$ was (implicitly) shown to be testable in [4], and $\mathcal{P}^*_H$ was shown to be testable in [6].

- **k-colorability:** The k-colorability property was (implicitly) shown to be testable already in [105]. In [75], a simplified explicit tester was studied with a significantly better query complexity. This result was further improved by [7].

- **Induced vertex colorability:** The main technical step in the proof of the main result of [6] was in showing that for every *finite* set of k-colored graphs $\mathcal{K}$, one can test the property of a graph being vertex k-colorable with no induced colored graph from the set $\mathcal{K}$. Note, that any such property is hereditary

condition above will be more convenient for the proof of Theorem 1.4.

- **Induced edge colorability:** Following [6], further induced edge-colorability properties were studied in [60]. In this case we have a *finite* set of *k-edge*-colored graphs $\mathcal{K}$, and the property defined by $\mathcal{K}$ is that of having a k-edge-coloring with no induced colored graph from the set $\mathcal{K}$. Note, that any such property is hereditary, and that by Theorem 1.1 we can even take $\mathcal{K}$ to be an infinite family of edge-colored graphs.

- **Graph partition problems:** One of the main results of [75] is that any graph-partition problem is testable with *two-sided* error. A characterization of the graph-partition properties that are testable with one-sided error was obtained in [77]. This characterization (essentially) follows as a special case of Theorem 1.4, as what it (implicitly) states is that a partition problem is testable with one-sided error if and only if it is hereditary.

- **One-sided vs. two-sided testers:** Alon has shown ([77], Appendix D) that if a hereditary graph property is testable with two-sided error then it is also testable with one-sided error (but not necessarily with the same query complexity). By Theorem 1.1, this transformation becomes obsolete, as Theorem 1.1 directly asserts that any hereditary graph property is testable with one-sided error.

- **Bounded first order graph properties:** Theorem 4.8 extends the main result of [6], where the first order graph-property can contain only a single predicate A_i. See Section 4.5 for more details on this subject.

It is important to note that Theorems 1.1 and 1.4 do not assert the existence of one-sided error testers, which are as efficient as the ad-hoc testers that were designed for every specific property in the above mentioned papers. For example, the query complexity of the tester for k-colorability that follows as a special case of Theorem 1.1, is significantly larger than the query complexity which is guaranteed by the main result of [75] and [7]. These large bounds are obviously a consequence of the generality of Theorems 1.1 and 1.4. Furthermore, by Theorem 4.1, the upper bounds of Theorems 1.1 and 1.4 cannot be generally improved even for monotone graph properties. See the precise statement in Section 4.2.

Organization: Our main tool in the proof of Theorem 1.1 is a novel application of a powerful variant of Szemerédi's Regularity Lemma proved in [6]. In Section 1.2 we introduce the basic notions of regularity and state the regularity lemmas that we use and some of their standard consequences. The proof of Theorem 1.1 is quite involved technically, and thus we give in Section 1.3 an overview of it. In this section we also prove Theorem 4.6. The ideas of this proof, especially the usage of the notion of colored-homomorphism, may be useful for handling other problems involving induced subgraphs. In Section 1.4 we give the full proof of Theorem 1.1 as well as the proof of Theorem 1.4. In Section 1.5, we describe several possible extensions and open problems that this chapter suggests. Throughout the chapter, whenever we relate, for example, to a function $f_{3.1}$, we mean the function f defined in Lemma/Claim/Theorem 3.1.

1.2 Regularity Lemma Background

As we have mentioned, the proof of Theorem 1.1 relies on a variant of Szemeredi's regularity lemma [112]. In this section we discuss the basic notions of regularity related to this lemma, some of the basic applications of regular partitions and state the regularity lemmas that we use in the proof of Theorem 1.1. See [90] for a comprehensive survey on the regularity-lemma. We start with some basic definitions and results.

1.2.1 The basics

For every two nonempty disjoint vertex sets A and B of a graph G, we define $e(A,B)$ to be the number of edges of G between A and B. The *edge density* of the pair is defined as $d(A,B) = e(A,B)/|A||B|$, where $e(A,B)$ denotes the number of edges connecting A and B.

Definition 1.5. (γ-regular pair) *A pair (A,B) is* γ-regular*, if for any two subsets $A' \subseteq A$ and $B' \subseteq B$, satisfying $|A'| \geq \gamma|A|$ and $|B'| \geq \gamma|B|$, the inequality $|d(A',B') - d(A,B)| \leq \gamma$ holds.*

Note that a sufficiently large random bipartite graph, where each edge is chosen independently with probability d, is very likely to be a γ-regular pair with density roughly d, for any $\gamma > 0$. Thus, in some sense, the smaller γ is, the closer a γ-regular pair is to looking like a random bipartite graph. For this reason, the reader who is unfamiliar with the regularity lemma and its applications, should try and compare the statements given in this section to analogous statements about random graphs. One such example is Lemma 1.6 below. Let F be a graph on f vertices and let G be a graph obtained by taking a copy of F, replacing every vertex with a sufficiently large independent set, every edge with a random bipartite graph of large enough edge density and every non-edge with a random bipartite graph of small enough edge density. It is easy to show that with high probability, G contains many induced copies of F. Lemma 1.6 shows that in order to infer that G contains many copies of F, it is enough to replace every edge with a "regular enough" pair. Several versions of this lemma were previously proved in papers using the regularity lemma. See e.g. Lemma 3.2 in [6].

Lemma 1.6. *For every real $0 < \eta < 1$ and integer $f \geq 1$ there exist $\gamma = \gamma_{1.6}(\eta, f)$ and $\delta = \delta_{1.6}(\eta, f)$ with the following property. Suppose that F is a graph on f vertices $v_f, \ldots, v_f$, and that $U_1, \ldots, U_f$ is an f-tuple of disjoint vertex sets of G such that for every $1 \leq i < j \leq f$ the pair (U_i, U_j) is γ-regular. Moreover, suppose that whenever $(v_i, v_j) \in E(F)$ we have $d(U_i, U_j) \geq \eta$, and whenever $(v_i, v_j) \notin E(F)$ we have $d(U_i, U_j) \leq 1 - \eta$. Then, at least $\delta \prod_{i=1}^{f} |U_i|$ of the f-tuples $u_1 \in U_1, \ldots, u_f \in U_f$ span an* **induced** *copy of F, where each u_i plays the role of v_i.*

Remark 1.7. *Observe, that the functions $\gamma_{1.6}(\eta, f)$ and $\delta_{1.6}(\eta, f)$ may and will be assumed to be monotone non-increasing in f. Also, for ease of future definitions (in particular the one given in (1.5)) we set $\gamma_{1.6}(\eta, 0) = \delta_{1.6}(\eta, 0) = 1$ for any $0 < \eta < 1$.*

Note, that in terms of regularity, Lemma 1.6 requires all the pairs (U_i, U_j) to be γ-regular. However, and this will be very important later in the chapter, the requirements in terms of density are not very restrictive. In particular, if $\eta \leq d(U_i, U_j) \leq 1 - \eta$ then we don't care if (i, j) is an edge of F.

A partition $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$ of the vertex set of a graph is called an *equipartition* if $|V_i|$ and $|V_j|$ differ by no more than 1 for all $1 \leq i < j \leq k$ (so in particular each V_i has one of two possible sizes). The Regularity Lemma of Szemerédi can be formulated as follows.

Lemma 1.8. ([112]) *For every m and $\epsilon > 0$ there exists a number $T = T_{1.8}(m, \epsilon)$ with the following property: Any graph G on $n \geq T$ vertices, has an equipartition $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$ of $V(G)$ with $m \leq k \leq T$, for which all pairs (V_i, V_j), but at most $\epsilon\binom{k}{2}$ of them, are ϵ-regular.*

The original formulation of the lemma allows also for an exceptional set with up to ϵn vertices outside of this equipartition, but one can first apply the original formulation with a somewhat smaller parameter instead of ϵ and then evenly distribute the exceptional vertices among the sets of the partition to obtain this formulation. The function $T_{1.8}(m, \epsilon)$ may and is assumed to be monotone nondecreasing in m and monotone non-increasing in ϵ.

Another lemma, which will be very useful in this chapter is Lemma 1.9 below. Some versions of this lemma appear in various papers applying the Regularity Lemma. See e.g. Corollary 3.4 in [6].

Lemma 1.9. *For every l and γ there exists $\delta = \delta_{1.9}(l, \gamma)$ such that for every graph G with $n \geq \delta^{-1}$ vertices there exist disjoint vertex sets $W_1, \ldots, W_l$ satisfying:*

1. *$|W_i| \geq \delta n$.*
2. ***All** $\binom{l}{2}$ pairs are γ-regular.*
3. *Either all pairs are with densities at least $\frac{1}{2}$, or all pairs are with densities less than $\frac{1}{2}$.*

Remark 1.10. *Observe, that the function $\delta_{1.9}(l, \gamma)$ may and will be assumed to be monotone non-increasing in l and monotone non-decreasing in γ. Therefore, for ease of future applications we will assume that for all l and γ we have $\delta_{1.9}(l, \gamma) \leq 1/2$.*

1.2.2 The main technical lemma

In this subsection we state the main technical lemma that we need for the proof of Theorem 1.1. To this end we introduce a convenient way of handling hereditary properties.

Definition 1.11. (Forbidden Induced Subgraphs) *For a hereditary graph property $\mathcal{P}$, define $\mathcal{F} = \mathcal{F}_{\mathcal{P}}$ to be the set of graphs which are minimal with respect to not satisfying property $\mathcal{P}$. In other words, a graph F belongs to $\mathcal{F}$ if it does not satisfy $\mathcal{P}$, but any graph obtained from F by removing a vertex, satisfies $\mathcal{P}$.*

For a (possibly infinite) family of graph $\mathcal{F}$, a graph G is said to be *induced $\mathcal{F}$-free* if it contains no induced copy of any graph $F \in \mathcal{F}$. Note, that for any hereditary graph property $\mathcal{P}$ there is a family of graphs $\mathcal{F} = \mathcal{F}_{\mathcal{P}}$ such that a graph satisfies $\mathcal{P}$ if and only if it is induced $\mathcal{F}$-free. For $\mathcal{F}$ one can simply take the family of forbidden induced subgraphs as in Definition 1.11. For example, when $\mathcal{P}$ is the property of being Chordal (see Subsection 1.1.1) then $\mathcal{F}_{\mathcal{P}}$ is the set of cycles of length at least 4. As another example note that if $\mathcal{P}$ is the property of being bipartite then $\mathcal{F}_{\mathcal{P}}$ is the family of odd cycles. Observe, that $\mathcal{F}$ may contain *infinitely* many graphs. Clearly for any family $\mathcal{F}$, the property of being induced $\mathcal{F}$-free is hereditary, thus, the hereditary graph properties are precisely the graph properties, which are equivalent to being induced $\mathcal{F}$-free for some family $\mathcal{F}$. For ease of presentation, it will be more convenient to derive Theorem 1.1 from the following (essentially equivalent[4]) lemma, whose proof is the main technical step in this chapter.

Lemma 1.12. *For every (possibly* **infinite***) family of graphs $\mathcal{F}$, there are functions $N_{\mathcal{F}}(\epsilon)$, $f_{\mathcal{F}}(\epsilon)$ and $\delta_{\mathcal{F}}(\epsilon)$ such that the following holds for any $\epsilon > 0$: If a graph G on $n \geq N_{\mathcal{F}}(\epsilon)$ vertices is ϵ-far from being induced $\mathcal{F}$-free, then G contains δn^f* **induced** *copies of a graph $F \in \mathcal{F}$ of size f, where $f \leq f_{\mathcal{F}}(\epsilon)$ and $\delta \geq \delta_{\mathcal{F}}(\epsilon)$.*

Let us give some intuition as to the difficulty of proving the above lemma. For simplicity let us consider first the case where we require the family of graphs $\mathcal{F}$ not to appear in a graph G as subgraph and not necessarily as induced subgraphs. In this case the property defined by $\mathcal{F}$ is that of being $\mathcal{F}$-free rather than induced $\mathcal{F}$-free. We stress that some of the details below are not completely accurate as they are only intended to give the main ideas and difficulties in the proof of Lemma 1.12.

A standard application of Lemmas 1.6 and 1.8 shows that for any *finite* set of graphs $\mathcal{F}$, the property of being $\mathcal{F}$-free is testable. We first use Lemma 1.6 by setting f to be the size of the largest graph in $\mathcal{F}$ and letting $\eta = \epsilon$. Lemma 1.6 gives a $\gamma_{1.6}$, which tells us how regular an equipartition should be (that is, how small should γ be) in order to find many copies of a member of $\mathcal{F}$ in it, assuming the input graph is ϵ-far from being $\mathcal{F}$-free. We then apply Lemma 1.8, with $\gamma = \gamma_{1.6}$. The main difficulty with applying this strategy when $\mathcal{F}$ is infinite is that we do not know a priori the size of the member of $\mathcal{F}$ that we will eventually find in the equipartition that Lemma 1.8 returns. After finding $F \in \mathcal{F}$ in an equipartition, we may find out that F is too large for Lemma 1.6 to be applied, because Lemma 1.8 was not used with a small enough γ. One may then try to find a new equipartition based on the size of F. However, that requires using a smaller γ, and thus the new equipartition may be larger (that is, contain more partition classes), and thus contain only larger members of $\mathcal{F}$. Hence, even the new γ is not good enough in order to apply Lemma 1.6. This leads to a circular definition of constants, which seems unbreakable. In the next subsection we introduce a version of the regularity lemma proved in [6] for a different reason. This lemma enables us to break this circular chain of definitions. This lemma can be considered a variant of the standard regularity lemma, where one can use a function that defines γ as a function of the size of the equipartition[5], rather than having to use a fixed γ as in Lemma 1.8.

[4] See Section 1.4 for a discussion about the subtle difference.

[5] This is a simplification of the actual statement, see item (3) in the statement of Lemma 1.14

1.2.3 The functional regularity lemma

Our main tool in the proof of Theorem 1.1 in addition to Lemmas 1.6 and 1.9 is Lemma 1.14 below, proved in [6]. This lemma can be considered a variant of the standard regularity lemma, where one can use a function that defines ϵ as a function of the size of the partition, rather than having to use a fixed ϵ as in Lemma 1.8. We denote such functions by $\mathcal{E}$ throughout the chapter. To state the lemma we need the following definition.

Definition 1.13. (The function $W_{\mathcal{E},m}$) *Let $\mathcal{E}(r) : \mathbb{N} \mapsto (0,1)$ be an arbitrary monotone non-increasing function. Let also m be an arbitrary positive integer. We define the function $W_{\mathcal{E},m} : \mathbb{N} \mapsto (0,1)$ inductively as follows: $W_{\mathcal{E},m}(1) = T_{1.8}(m, \mathcal{E}(0))$. For any integer $i > 1$ put $R = W_{\mathcal{E},m}(i-1)$ and define*

$$W_{\mathcal{E},m}(i) = T_{1.8}(R, \mathcal{E}(R)/R^2). \tag{1.1}$$

Lemma 1.14. ([6]) *For every integer m and monotone non-increasing function $\mathcal{E} : \mathbb{N} \mapsto (0,1)$ define*

$$S = S_{1.14}(m, \mathcal{E}) = W_{\mathcal{E},m}(100/\mathcal{E}(0)^4).$$

For any graph G on $n \geq S$ vertices, there exists an equipartition $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$ of $V(G)$ and an induced subgraph U of G, with an equipartition $\mathcal{B} = \{U_i \mid 1 \leq i \leq k\}$ of the vertices of U, that satisfy:

1. *$m \leq k \leq S$.*
2. *$U_i \subseteq V_i$ for all $i \geq 1$, and $|U_i| \geq n/S$.*
3. *In the equipartition $\mathcal{B}$,* **all** *pairs are $\mathcal{E}(k)$-regular.*
4. *All but at most $\mathcal{E}(0)\binom{k}{2}$ of the pairs $1 \leq i < j \leq k$ are such that $|d(V_i, V_j) - d(U_i, U_j)| < \mathcal{E}(0)$.*

Remark 1.15. *For technical reasons (see the proof in [6]), Lemma 1.14 requires that for any $r > 0$ the function $\mathcal{E}(r)$ will satisfy $\mathcal{E}(r) \leq \min\{\mathcal{E}(0)/4, 1/4r^2\}$. However, we can always assume wlog that $\mathcal{E}$ satisfies this condition because if it does not, then we can apply Lemma 1.14 with $\mathcal{E}'$ which is defined as $\mathcal{E}'(r) = \min\{\mathcal{E}(r), \mathcal{E}(0)/4, 1/4r^2\}$. We will thus disregard this technicality.*

The main power of Lemma 1.14 is that for *any* function $\mathcal{E}$ it allows us to find k sets of vertices $V_1, \ldots, V_k$ of size $\Omega(n)$ such that all pairs (V_i, V_j) are $\mathcal{E}(k)$-regular. Note, that in Lemma 1.8 we first fix the regularity measure γ, and then get via the lemma, k sets of vertices, where k can be very large in terms of γ.

One of the difficulties in the proof of Theorem 1.4, is in showing that all the constants that are used in the course of the proof can be upper bounded by functions depending on ϵ only. The following observation will thus be useful.

Proposition 1.16. *If m is bounded by a function of ϵ only then for any $\mathcal{E} : \mathbb{N} \mapsto (0,1)$, the integer $S = S_{1.14}(m, \mathcal{E})$ can be upper bounded by a function of ϵ only* [6].

It should be noted that the dependency of the function $T_{1.8}(m, \epsilon)$ on ϵ is a tower of exponents of height polynomial in $1/\epsilon$ (see the proof in [90]). Thus, even for moderate functions $\mathcal{E}$ the integer S has a huge dependency on ϵ, which is a tower of towers of exponents of height polynomial in $1/\epsilon$.

One of the main results of [6] is that for every *finite* set of graphs $\mathcal{F}$, the property of not containing any member of $\mathcal{F}$ as an induced subgraph can be tested with one-sided error and with query complexity depending on ϵ only. The proof technique in [6], which applies Lemmas 1.6, 1.9 and 1.14 critically relies on the fact that the family of graphs is finite. The main step in the proof of Theorem 1.1 is in extending the above to *infinite* families of graphs. The techniques we apply in the next section, in particular the notion of colored-homomorphism, may be useful in dealing with other problems involving induced subgraphs.

1.3 Overview of the New Regularity Technique

The proof of Lemma 1.12 is rather technical and long and appears in its entirety in Section 1.4. In this section we try to give an overview of its proof, while keeping out most of the (unnecessary) technical details. We break the overview of the proof of Lemma 1.12 into two parts. In the first subsection we give an overview of the proof of a version of Lemma 1.12, which is suitable for handling monotone properties. We believe that the intuition of this version of Lemma 1.12 is much easier to explain. In the second subsection we give an overview of the proof of Lemma 1.12.

1.3.1 Intuition for monotone properties

Recall that our motivation for proving Lemma 1.12 is that any hereditary property is equivalent to being induced $\mathcal{F}$-free for some set of graphs $\mathcal{F}$. It is easy to see that a similar equivalence holds with respect to monotone graph properties, and being $\mathcal{F}$-free. More precisely, any monotone property is equivalent to the property of being $\mathcal{F}$-free for some (possibly infinite) family of forbidden graphs $\mathcal{F}$. In this subsection we will sketch the proof of the following lemma.

Lemma 1.17. *For every (possibly* **infinite***) family of graphs $\mathcal{F}$, there are functions $N_{\mathcal{F}}(\epsilon)$, $f_{\mathcal{F}}(\epsilon)$ and $\delta_{\mathcal{F}}(\epsilon)$ such that the following holds for any $\epsilon > 0$: If a graph G on $n \geq N_{\mathcal{F}}(\epsilon)$ vertices is ϵ-far from being $\mathcal{F}$-free, then G contains δn^f copies of a graph $F \in \mathcal{F}$ of size f, where $f \leq f_{\mathcal{F}}(\epsilon)$ and $\delta \geq \delta_{\mathcal{F}}(\epsilon)$.*

[6] In our application of Lemma 1.14 the function $\mathcal{E}$ will (implicitly) depend on the error parameter ϵ. For example, we will set $\mathcal{E}(r) = f(r, \epsilon)$ for some function f. However, that will not change the fact that $S_{1.14}(m, \mathcal{E})$ can be upper bounded by a function of ϵ only.

It is not difficult to see that just as Lemma 1.12 immediately implies that any hereditary property is testable (see the proof of Theorem 1.1 for the full details), the above lemma can be used to infer that any monotone graph property is testable. In this subsection we will sketch an overview of the proof of the above lemma.

Throughout the chapter we will make an extensive use of the notion of graph homomorphism, which we turn to formally define.

Definition 1.18. (Homomorphism) *A homomorphism from a graph F to a graph K is a mapping $\varphi : V(F) \mapsto V(K)$, which maps edges to edges, namely $(v,u) \in E(F)$ implies $(\varphi(v), \varphi(u)) \in E(K)$.*

In what follows, $F \mapsto K$ denotes the fact that there is a homomorphism from F to K. We will also say that a graph H is homomorphic to K if $H \mapsto K$. Note, that a graph H is homomorphic to a complete graph of size k if and only if H is k-colorable.

For the proof of Lemma 1.17 we will need a version of Lemma 1.6 that is suitable for finding non-induced copies of a certain fixed graph[7]. Let F be a graph on f vertices and K a graph on k vertices, and suppose $F \mapsto K$. Let G be a graph obtained by taking a copy of K, replacing every vertex with a sufficiently large independent set, and every edge with a random bipartite graph of edge density d. It is easy to show that with high probability, G contains a copy of F (in fact, many). The following lemma shows that in order to infer that G contains a copy of F, it is enough to replace every edge with a "regular enough" pair. Intuitively, the larger f and k are, and the sparser the regular pairs are, the more regular we need each pair to be, because we need the graph to be "closer" to a random graph. This is formulated in the lemma below. Several versions of this lemma were previously proved in papers using the regularity lemma (see [90]).

Lemma 1.19. *For every real $0 < \eta < 1$, and integers $k, f \geq 1$ there exist $\gamma = \gamma_{1.19}(\eta, k, f)$, and $N = N_{1.19}(\eta, k, f)$ with the following property. Let F be any graph on f vertices, and let $U_1, \ldots, U_k$ be k pairwise disjoint sets of vertices in a graph G, where $|U_1| = \ldots = |U_k| \geq N$. Suppose there is a mapping $\varphi : V(F) \mapsto \{1, \ldots, k\}$ such that the following holds: If (i,j) is an edge of F then $(U_{\varphi(i)}, U_{\varphi(j)})$ is γ-regular with density at least η. Then $U_1, \ldots, U_k$ span a copy of F.*

For an equipartition of a graph G, let the *regularity graph* of G, denoted $R = R(G)$, be the following graph: We first use Lemma 1.8 in order to obtain the equipartition satisfying the assertions of the lemma. Let k be the size of the equipartition. Then, R is a graph on k vertices, where vertices i and j are connected if and only if (V_i, V_j) is a dense regular pair (with the appropriate parameters). In some sense, the regularity graph is an approximation of the original graph, up to γn^2 modifications. One of the main (implicit) implications of the regularity lemma is the following: Suppose we consider two graphs to be *similar* if

[7] The reader may (rightfully) wonder why do we need a lemma for finding not-necessarily induced copies if we have a lemma for finding induced ones. The reason is that the requirements of Lemma 1.6 are more difficult to satisfy than the requirements of Lemma 1.19. In particular, in Lemma 1.6 the copies of F can have only one vertex in each set U_i while in Lemma 1.19 they can have an arbitrary number. This is partially why the proof of Lemma 1.17 is significantly simpler compared to the proof of Lemma 1.12

their regularity graphs are identical. It thus follows from Lemma 1.8 that for every $\gamma > 0$, the number of graphs that are pairwise non-similar is bounded by a function of γ only ($2^{\binom{T}{2}}$, where $T = T_{1.8}(1/\gamma, \gamma)$). Namely, up to γn^2 modifications, all the graphs can be approximated using a set of equipartitions of size bounded by a function of γ only. The reader is referred to [54] where this interpretation of the regularity lemma is also (implicitly) used. This leads us to the key definitions of the proof of Theorem 1.17. The reader should think of the graphs R considered below as the set of regularity graphs discussed above, and the parameter r as representing the size of R.

Definition 1.20. (The family $\mathcal{F}_r$) *For any (possibly infinite) family of graphs $\mathcal{F}$, and any integer r let $\mathcal{F}_r$ be the following set of graphs: A graph R belongs to $\mathcal{F}_r$ if it has at most r vertices and there is at least one $F \in \mathcal{F}$ such that $F \mapsto R$.*

Practicing definitions, observe that if $\mathcal{F}$ is the family of odd cycles, then $\mathcal{F}_r$ is precisely the family of non-bipartite graphs of size at most r. In the proof of Lemma 1.17, the set $\mathcal{F}_r$, defined above, will represent a subset of the regularity graphs of size at most r. Namely, those R for which there is at least one $F \in \mathcal{F}$ such that $F \mapsto R$. As r will be bounded by a function of ϵ only, and thus finite, we can take the maximum over all the graphs $R \in \mathcal{F}_r$, of the size of the smallest $F \in \mathcal{F}$ such that $F \mapsto R$. We thus define

Definition 1.21. (The function $\Psi_{\mathcal{F}}$) *For any family of graphs $\mathcal{F}$ and integer r for which $\mathcal{F}_r \neq \emptyset$, define*

$$\Psi_{\mathcal{F}}(r) = \max_{R \in \mathcal{F}_r} \min_{\{F \in \mathcal{F}: F \mapsto R\}} |V(F)|. \tag{1.2}$$

Define $\Psi_{\mathcal{F}}(r) = 0$ if $\mathcal{F}_r = \emptyset$. Therefore, $\Psi_{\mathcal{F}}(r)$ is monotone non-decreasing in r.

Practicing definitions again, note that if $\mathcal{F}$ is the family of odd cycles, then $\Psi_{\mathcal{F}}(r) = r$ when r is odd, and $\Psi_{\mathcal{F}}(r) = r - 1$ when r is even. The "right" way to think of the function $\Psi_{\mathcal{F}}$ is the following: Let R be a graph of size at most r and suppose we are guaranteed that there is a graph $F' \in \mathcal{F}$ such that $F' \mapsto R$ (thus $R \in \mathcal{F}_r$). Then by this information only and *without* having to know the structure of R itself, the definition of $\Psi_{\mathcal{F}}$ implies that there is a graph $F \in \mathcal{F}$ of size at most $\Psi_{\mathcal{F}}(r)$, such that $F \mapsto R$.

The function $\Psi_{\mathcal{F}}$ has a critical role in the proof of Lemma 1.17. The first usage of this function is that as by Lemma 1.8 we can upper bound the size of the regularity graph R (via the function $T_{1.8}$), we can also upper bound the size of the smallest graph $F \in \mathcal{F}$ for which $F \mapsto R$. As we have mentioned in the previous section, the main difficulty that prevents one from proving Lemma 1.17 using Lemma 1.19 is that one does not know a priori the size of the graph that one may expect to find in the equipartition. This leads us to define the following function where $0 < \epsilon < 1$ is an arbitrary real.

$$\mathcal{E}(r) = \begin{cases} \epsilon, & r = 0 \\ \gamma_{1.6}(\epsilon,\ r,\ \Psi_{\mathcal{F}}(r)), & r \geq 1 \end{cases} \tag{1.3}$$

In simple words, given r, which will represent the size of the equipartition and thus also the size of the regularity graph which it defines, $\mathcal{E}(r)$ returns "how regular" this equipartition

should be in order to allow one to find many copies of the *largest* graph one may possibly have to work with. Note, that we obtain the upper bound on the size of this largest possible graph, by invoking $\Psi_{\mathcal{F}}(r)$. As for different families of graphs $\mathcal{F}$, the function $\Psi_{\mathcal{F}}(r)$ may behave differently, $\mathcal{E}(r)$ may also behave differently for different families $\mathcal{F}$, as it is defined in terms of $\Psi_{\mathcal{F}}(r)$. However, and this is one of the key points of the proof, as we are fixing the family of graphs $\mathcal{F}$, the function $\mathcal{E}(r)$ depends only on r.

Given the above definitions we apply Lemma 1.14 with a slight modification of $\mathcal{E}(r)$ in order to obtain an equipartition of G. We then throw away edges that reside inside the sets V_i and between (V_i, V_j), whose edge density differs significantly from that of (U_i, U_j) . We then argue that we thus throw away less than ϵn^2 edges. As G is by assumption ϵ-far from not containing a member of $\mathcal{F}$, the new graph still contains a copy of $F \in \mathcal{F}$. By the definition of the new graph, it thus means that there is a (natural) homomorphism from F to the regularity graph of G. We then arrive at the main step of the proof, where we use the key property of Lemma 1.14, item (3), and the definition of $\mathcal{E}(r)$ to get that the sets U_i are regular enough to let us use Lemma 1.19 on them and to infer that they span many copies of some graph $F \in \mathcal{F}$.

1.3.2 Overview of the proof of Lemma 1.12

The proof of Lemma 1.17, which we have sketched in the previous subsection, relied on the notion of graph homomorphism. For the proof of Lemma 1.12 we will need a new type of homomorphism that is suitable for handling induced subgraph.

Definition 1.22. (Colored-Homomorphism) *Let K be a complete graph whose vertices are colored black or white, and whose edges are colored black, white or grey (neither the vertex coloring nor the edge coloring is assumed to be proper in the standard sense). A* colored-homomorphism *from a graph F to a graph K is a mapping $\varphi : V(F) \mapsto V(K)$, which satisfies the following:*

1. *If $(u, v) \in E(F)$ then either $\varphi(u) = \varphi(v) = t$ and t is colored black, or $\varphi(u) \neq \varphi(v)$ and $(\varphi(u), \varphi(v))$ is colored black or grey.*
2. *If $(u, v) \notin E(F)$ then either $\varphi(u) = \varphi(v) = t$ and t is colored white, or $\varphi(u) \neq \varphi(v)$ and $(\varphi(u), \varphi(v))$ is colored white or grey.*

If there is a colored-homomorphism from a graph F to a colored complete graph K, we write for brevity $F \mapsto_c K$. Some explanation is in place as to the meaning of the colors in the above definition. To this end, it is instructive to compare the definition of a colored-homomorphism to the standard notion of homomorphism, that was defined in the previous subsection (recall that for brevity, we denote by $F \mapsto K$ the fact that there is a homomorphism from F to K.). The fact that $F \mapsto K$, simply means that we can partition the vertex set of F into $k = |V(K)|$ subsets $V_1, \ldots, V_k$, such that each V_i is edgeless and if $(i, j) \notin E(K)$ then none of the vertices of F that belong to V_i is connected to any of the vertices of F that belong to V_j. In particular, note that $F \mapsto K_k$ if and only if F is k-colorable (where K_k is a clique of size k). The standard notion of homomorphism

is sufficient for dealing with not necessarily induced subgraphs as was carried out in the previous subsection. The reason is that having a homomorphism to a graph K is "closed under removal of vertices and edges" in the sense that if $F \mapsto K$ and F' is a subgraph of F then $F' \mapsto K$. When one wants to handle *induced* subgraphs it soon turns out that standard homomorphism is not sufficient as it does not supply enough information about F. The clear reason for that is that a standard homomorphism has no requirement about the non-edges of the graph. Returning to the colored-homomorphism from Definition 1.22, suppose we interpret the colors of K as follows: A white edge of K represents a non-edge, a black edge of K represents an existing edge and a grey edge represents a "don't care." As for the vertex colors, we think of a black vertex as a complete graph, and a white vertex as an edgeless graph. Thus, the fact that $F \mapsto_c K$ where K is a colored complete graph of size k is the following: There is a partition of $V(F)$ into k subsets $V_1, \ldots, V_k$ such that each V_i is either complete or edgeless, where V_i is complete if $i \in V(K)$ is black and edgeless if $i \in E(K)$ is white. Also, if (i,j) is colored white then none of the vertices of F that belong to V_i is connected to any of the vertices of F that belong to V_j. Similarly, if (i,j) is colored black then all the vertices of F that belong to V_i are connected to all the vertices of F that belong to V_j. Finally, if (i,j) is colored grey then there is no restriction on pairs $(v \in V_i, u \in V_j)$ (or in our "formal" notation, we "don't care" if $(v \in V_i, u \in V_j)$ is an edge of F). It is clear that a colored-homomorphism carries a lot more information about the structure of F than a standard homomorphism.

Our definition of colored-homomorphism should also be thought of with Lemma 1.6 in mind. Note, that in this lemma we only require $d(U_i, U_j) \geq \eta$ when $(i,j) \in E(F)$ and $d(U_i, U_j) \leq 1 - \eta$ when $(i,j) \notin E(F)$. In particular, if $\eta \leq d(U_i, U_j) \leq 1 - \eta$ then we "don't care" whether $(i,j) \in E(F)$. In fact, as the details of the proof of Lemma 1.12 reveal, the possibility of having grey edges in the coloring of K in the definition of the colored-homomorphism is unavoidable (at least in our proof). Note, that as far as Lemma 1.6 is concerned, we only need the edge coloring in the colored-homomorphism. The details below supply some explanation for the need of the vertex coloring.

We now turn to discuss the relation between the standard regularity lemma (Lemma 1.8), the stronger regularity lemma (Lemma 1.14) and colored-homomorphism. We stress that some of the explanations we give below are not completely accurate, and are given in order to explain the main ideas of the proof. The formal proof appears in Section 1.4. Given $\epsilon > 0$ and a graph G, Lemma 1.8 returns an equipartition of $V(G)$ of size k. Recall from the previous subsection that the *regularity graph* of G, denoted $R = R(G)$, is the following graph. R is a graph on k vertices, where vertices i and j are connected if and only if (V_i, V_j) is a dense regular pair (with the appropriate parameters). In some sense, the regularity graph is an approximation of the original graph, up to ϵn^2 modifications. This approximation was good enough when considering monotone properties in the previous subsection but it is not good enough when dealing with induced graphs, which is the case we consider here. The reason is that R only approximates the dense pairs of the equipartition, while it carries no restriction or information on the sparse pairs in this equipartition. This is somewhat analogous to the fact that standard homomorphism is not good enough for dealing with induced subgraphs. Just like we defined colored-homomorphism we introduce

colored regularity graphs as follows; Let R be a complete graph on k vertices. Color (i,j) black if (V_i,V_j) is a very dense pair, white if (V_i,V_j) is a very sparse pair, and grey if (V_i,V_j) is neither very dense nor very sparse (we omit the precise definition of "very"). Note, that a colored-regularity graph carries a lot more information about G. Note also how this definition relates to a colored-homomorphism.

Suppose a graph G is ϵ-far from being induced $\mathcal{F}$-free. We would want to apply Lemma 1.8, then construct the colored regularity graph, and then argue that if we make few (less than ϵn^2) modifications in G then the new graph $\tilde{G}$, contains an induced copy of a graph $F \in \mathcal{F}$. Furthermore, as we make very few changes, the colored regularity graph is also a "good" approximation of $\tilde{G}$. We would thus want to use Lemma 1.6, where for the sets $U_1,\ldots,U_f$ we take the clusters $V_1,\ldots,V_k$ of the equipartition in order to get that there are many induced copies of F in G. However, we are faced with the following two problems: (i) As $\mathcal{F}$ may be infinite, we don't know the size of the member of $\mathcal{F}$ that we may expect to find in $\tilde{G}$. As Lemma 1.6 needs to know the size of F in advance, we don't know how small a γ should we choose in order to apply Lemma 1.8 [8]. (ii) Note that Lemma 1.6 allows the copies of F to have only one vertex in each of the sets U_i. However, the copy of the member of $\mathcal{F}$ that we may find in $\tilde{G}$ may have many vertices in each cluster V_i. Note further, that Lemma 1.8 does not guarantee anything about the graphs induced by each V_i.

The main idea of the proof is to overcome the first problem by applying Lemma 1.14 with a suitable function $\mathcal{E}$ that will guarantee that the partition is regular enough even for the largest graph we may expect to find in $\tilde{G}$. For the second problem we apply Lemma 1.9 on each of the clusters V_i in order to find subsets $W_{i,1},\ldots,W_{i,f} \subset V_i$. Note that by Lemma 1.6, if for all j',j'' $d(W_{i,j'},W_{i,j''}) \geq 1/2$ then $W_{i,1},\ldots,W_{i,f}$ span many cliques of size f, while if for all j',j'', $d(W_{i,j'},W_{i,j''}) \leq 1/2$ they span many independent sets of size f (note that by Lemma 1.9 one of these cases holds). This is the main reason for the vertex coloring of R, that is, we color vertex i of R black, if the sets returned by Lemma 1.9 are very dense, and white if they are sparse. We note that overcoming both problems mentioned above *simultaneously* adds another level of complication.

An important ingredient in the proof of Lemma 1.12 will be the following function. The reader should think of the graphs R considered below as the set of colored-regularity graphs discussed above, and the parameter r as representing the size of R.

Definition 1.23. (The family $\mathcal{F}_r$) *For any (possibly infinite) family of graphs $\mathcal{F}$, and any integer r let $\mathcal{F}_r$ be the following set of graphs: A colored complete graph R belongs to $\mathcal{F}_r$ if it has at most r vertices and there is at least one $F \in \mathcal{F}$ such that $F \mapsto_c R$.*

In the proof of Lemma 1.12, the set $\mathcal{F}_r$, defined above, will represent a subset of the colored regularity graphs of size at most r. Namely, those R for which there is at least one $F \in \mathcal{F}$ such that $F \mapsto_c R$. We now define

Definition 1.24. (The function $\Psi_{\mathcal{F}}$) *For any family of graphs $\mathcal{F}$ and integer r for which $\mathcal{F}_r \neq \emptyset$, let*

$$\Psi_{\mathcal{F}}(r) = \max_{R \in \mathcal{F}_r} \min_{\{F \in \mathcal{F}: F \mapsto_c R\}} |V(F)|. \tag{1.4}$$

[8] we had the same difficulty in the previous subsection when we dealt with monotone properties

Define $\Psi_{\mathcal{F}}(r) = 0$ if $\mathcal{F}_r = \emptyset$. Therefore, $\Psi_{\mathcal{F}}(r)$ is monotone non-decreasing in r.

As in the previous subsection, $\Psi_{\mathcal{F}}$ is one of the main tools with which we apply Lemma 1.14. As by Lemma 1.8 we can upper bound the size of the regularity graph R, we can also upper bound the size of the smallest graph $F \in \mathcal{F}$ for which $F \mapsto_c R$.

As we have mentioned in the previously, the main difficulty that prevents one from proving Theorem 1.1 using Lemma 1.6 is that one does not know a priori the size of the graph that one may expect to find in the equipartition. This leads us to the define the following function

$$\mathcal{E}(r) = \gamma_{1.6}(\epsilon/6,\ \Psi_{\mathcal{F}}(r)) \cdot \delta_{1.9}(\Psi_{\mathcal{F}}(r),\ \gamma_{1.6}(\epsilon/6,\ \Psi_{\mathcal{F}}(r))) \tag{1.5}$$

We next try to explain why the above defined $\mathcal{E}(r)$ when applied with Lemma 1.14 is useful in resolving the two difficulties mentioned above. Recall that r stands for the size of the colored-regularity graph returned by Lemma 1.14. If we apply Lemma 1.14 with the above $\mathcal{E}$ then by the first term in the definition of $\mathcal{E}$ we know that the sets U_i (recall the statement of Lemma 1.8) are regular enough to allow one to apply Lemma 1.6 with the largest member of $\mathcal{F}$, which we may need to work with. This is due to invoking $\Psi_{\mathcal{F}}(r)$. This "resolves" the first problem we mentioned earlier. The reason we need the second term in the definition of $\mathcal{E}$ is that we intend to apply Lemma 1.9 on each of the sets V_i in order to obtain certain subsets $W_{1,i}, \ldots, W_{j,i}$ of V_i. This term guarantees that even subsets of V_i will be "regular-enough" for our purposes. This way we "resolve" the second problem mentioned earlier.

1.4 Proofs of Main Results

We start with the proof of Lemma 1.12, which is the main technical step in the proof of Theorem 1.1. We then use Theorem 1.1 in order to prove Theorem 1.4. We assume the reader is familiar with the overview of the proof of Lemma 1.12 given in Section 1.3. For the proof we need the following simple and well-known fact, which states that large enough subsets of a regular pair are themselves somewhat regular.

Claim 1.25. *If (A, B) is a γ-regular pair with density η, and $A' \subseteq A$ and $B' \subseteq B$ satisfy $|A'| \geq \xi|A|$ and $|B'| \geq \xi|B|$ for some $\xi \geq \gamma$, then (A', B') is a $\max\{2\gamma, \gamma/\xi\}$-regular pair.*

Proof: As (A, B) is a γ-regular pair with density η, then by definition of a regular pair, for every pair of subsets of $A' \subseteq A$ with $|A'| \geq \xi|A| \geq \gamma|A|$ and $B' \subseteq B$ with $|B'| \geq \xi|B| \geq \gamma|B|$ we have $|d(A', B') - d(A, B)| \leq \gamma$. Note, that if A' and B' are as above, then for every pair of subsets $A'' \subseteq A'$ and $B'' \subseteq B'$ satisfying $|A''| \geq \frac{\gamma}{\xi}|A'|$ and $|B''| \geq \frac{\gamma}{\xi}|B'|$ also satisfy $|A''| \geq \gamma|A|$ and $|B''| \geq \gamma|B|$. Therefore, by the γ-regularity of (A, B) we have $|d(A'', B'') - d(A, B)| \leq \gamma$. We thus conclude that $|d(A'', B'') - d(A', B')| \leq 2\gamma$. Hence, (A', B') is $\max\{2\gamma, \gamma/\xi\}$-regular. $\square$

Proof of Lemma 1.12: Fix any family of graphs $\mathcal{F}$. Let $\Psi_{\mathcal{F}}(r)$ be the function from

Definition 1.24 and define the following functions of r:

$$\alpha(r) = \delta_{1.9}(\Psi_{\mathcal{F}}(r),\ \gamma_{1.6}(\epsilon/6,\ \Psi_{\mathcal{F}}(r))), \tag{1.6}$$

$$\beta(r) = \alpha(r) \cdot \gamma_{1.6}(\epsilon/6,\ \Psi_{\mathcal{F}}(r)), \tag{1.7}$$

and

$$\mathcal{E}(r) = \begin{cases} \epsilon/6, & r = 0 \\ \min\{\beta(r),\ \epsilon/6\}, & r \geq 1 \end{cases} \tag{1.8}$$

For the rest of the proof set

$$S(\epsilon) = S_{1.14}(6/\epsilon, \mathcal{E}), \tag{1.9}$$

and note that as we define $S(\epsilon)$ in terms of $m = 6/\epsilon$ we get by Proposition 1.16 that $S(\epsilon)$ is indeed bounded by a function of ϵ only. We now set $N_{\mathcal{F}}(\epsilon)$ to be the following function of ϵ

$$N = N_{\mathcal{F}}(\epsilon) = S(\epsilon) \tag{1.10}$$

(as we have just argued, $S(\epsilon)$ and therefore also N can be upper bounded by functions of ϵ only). We postpone the definition of $f_{\mathcal{F}}(\epsilon)$ and $\delta_{\mathcal{F}}(\epsilon)$ till the end of the proof.

In the rest of the proof we consider any graph G on n vertices, with $n \geq N \geq S(\epsilon)$, which is ϵ-far from being induced $\mathcal{F}$-free. Given G, we can use Lemma 1.14 with $m = 6/\epsilon$ and $\mathcal{E}(r)$ as defined in (1.8), in order to obtain an equipartition of $V(G)$ into $6/\epsilon \leq k \leq S(\epsilon)$ clusters $V_1, \ldots, V_k$ (this is possible by item (1) in Lemma 1.14). Throughout the rest of the proof, k will denote the size of the equipartition returned by Lemma 1.14. By item (2) of Lemma 1.14, for every $1 \leq i \leq k$ we have sets $U_i \subseteq V_i$ each of size at least $n/S(\epsilon)$. Also, by item (3) of Lemma 1.14, **every** pair of these sets is at least $\beta(k)$-regular (recall that $\mathcal{E}(k) \leq \beta(k)$). For each $1 \leq i \leq k$, apply Lemma 1.9 on the subgraph induced by G on each U_i with $\ell = \Psi_{\mathcal{F}}(k)$ and $\gamma = \gamma_{1.6}(\epsilon/6,\ \Psi_{\mathcal{F}}(k))$ in order to obtain the appropriate sets $W_{i,1}, \ldots, W_{i,\Psi_{\mathcal{F}}(k)} \subset U_i$, all of size at least $\alpha(k)|U_i|$ (recall the definition of $\alpha(r)$ in (1.6)). It is crucial to note that we apply Lemma 1.9 on each of the sets $U_1, \ldots, U_k$ *after* we apply Lemma 1.14 on G, thus we "know" the value of k. The following observation will be useful for the rest of the proof:

Claim 1.26. *All the pairs $(W_{i,i'}, W_{j,j'})$ are $\gamma_{1.6}(\epsilon/6, \Psi_{\mathcal{F}}(k))$-regular. Also, if $i \neq j$ then we also have $|d(W_{i,i'}, W_{j,j'}) - d(U_i, U_j)| \leq \epsilon/6$.*

Proof: Consider first pairs that belong to the same set U_i. In this case, the fact that any pair $(W_{i,i'}, W_{i,j'})$ is $\gamma_{1.6}(\epsilon/6, \Psi_{\mathcal{F}}(k))$-regular follows immediately from our choice of these sets, as we applied Lemma 1.9 on each set U_i with $\gamma = \gamma_{1.6}(\epsilon/6,\ \Psi_{\mathcal{F}}(k))$. Consider now pairs that belong to different sets U_i, U_j. As was mentioned above, any pair (U_i, U_j) is $\beta(k)$-regular. As each set $W_{i,j}$ satisfies $|W_{i,j}| \geq \alpha(k)|U_i|$, we get from Claim 1.25 and the definition of $\beta(k)$ that any pair $(W_{i,i'}, W_{j,j'})$ is at least $\max\{2\beta(k), \beta(k)/\alpha(k)\} \leq \gamma_{1.6}(\epsilon/6, \Psi_{\mathcal{F}}(k))$-regular (here we use the fact that $\alpha(k) \leq 1/2$, which is guaranteed by Remark 1.10). Finally, as each of the sets $W_{i,j}$ satisfies $|W_{i,j}| \geq \alpha(k)|U_i| \geq \beta(k)|U_i| \geq \mathcal{E}(k)|U_i|$ we get from the fact that each pair (U_i, U_j) is $\mathcal{E}(k)$-regular that $|d(W_{i,i'}, W_{j,j'}) - d(U_i, U_j)| \leq \mathcal{E}(k) \leq \epsilon/6$, thus completing the proof. $\square$

Recall that our goal is to show that G contains many induced copies of some graph $F \in \mathcal{F}$. To this end, we would like to apply Lemma 1.6 on some appropriately chosen subset of the sets $W_{i,j}$ defined above. As by Claim 1.26 all the pairs of sets $W_{i,j}$ are regular (we will latter infer that they are regular enough for our purposes), we just have to find sets, whose densities will correspond to the edge set of some graph $F \in \mathcal{F}$ (recall the statement of Lemma 1.6). To this end, we define a graph $\tilde{G}$ that will help us in choosing the sets $W_{i,j}$. The graph $\tilde{G}$ is obtained from G by adding and removing the following edges, in the following order:

1. For $1 \leq i < j \leq k$ such that $|d(V_i, V_j) - d(U_i, U_j)| > \epsilon/6$, for all $v \in V_i$ and $v' \in V_j$ the pair (v, v') becomes an edge if $d(U_i, U_j) \geq \frac{1}{2}$, and becomes a non-edge if $d(U_i, U_j) < \frac{1}{2}$.

2. For $1 \leq i < j \leq k$ such that $d(U_i, U_j) < \frac{2}{6}\epsilon$, all edges between V_i and V_j are removed. For all $1 \leq i < j \leq k$ such that $d(U_i, U_j) > 1 - \frac{2}{6}\epsilon$, all non-edges between V_i and V_j become edges.

3. If for a fixed i all densities of pairs from $W_{i,1}, \ldots, W_{i,l}$ are less than $\frac{1}{2}$, all edges within the vertices of V_i are removed. Otherwise, all the above densities are at least $\frac{1}{2}$ (by the choice of $W_{i,1}, \ldots, W_{i,l}$ through Lemma 1.9), in which case all non-edges within V_i become edges.

In what follows we denote by $d(A, B)$ and $\tilde{d}(A, B)$ the edge density of the pair (A, B) in G and $\tilde{G}$, respectively. The following claim states several relations between the densities of G and $\tilde{G}$.

Claim 1.27. *For any i and $i' < j'$ we either have $\tilde{d}(W_{i,i'}, W_{i,j'}) = 1$ and $d(W_{i,i'}, W_{i,j'}) \geq \frac{1}{2}$ or $\tilde{d}(W_{i,i'}, W_{i,j'}) = 0$ and $d(W_{i,i'}, W_{i,j'}) \leq \frac{1}{2}$. Also, for any $i < j$ and any i', j' precisely one of the following holds:*

1. *$\tilde{d}(V_i, V_j) = 1$ and $d(W_{i,i'}, W_{j,j'}) \geq \epsilon/6$.*
2. *$\tilde{d}(V_i, V_j) = 0$ and $d(W_{i,i'}, W_{j,j'}) \leq 1 - \epsilon/6$.*
3. *$\epsilon/6 \leq \tilde{d}(V_i, V_j) \leq 1 - \epsilon/6$ and $\epsilon/6 \leq d(W_{i,i'}, W_{j,j'}) \leq 1 - \epsilon/6$.*

Proof: The proof follows easily from the three steps for obtaining $\tilde{G}$ from G. The first assertion of the claim follows directly from the third step of obtaining $\tilde{G}$. As for the second assertion, assume the first step was applied to a pair (V_i, V_j). In this case either $\tilde{d}(V_i, V_j) = 1$ and $d(U_i, U_j) \geq 1/2$ or $\tilde{d}(V_i, V_j) = 0$ and $d(U_i, U_j) \leq 1/2$. By Claim 1.26 we get that in the former case for any i', j' we have $d(W_{i,i'}, W_{j,j'}) \geq 1/2 - \epsilon/6 \geq \epsilon/6$, while in the later $d(W_{i,i'}, W_{j,j'}) \leq 1/2 + \epsilon/6 \leq 1 - \epsilon/6$, as needed. Note, that if the first step was applied to a pair (V_i, V_j) then the second step has no effect, thus either (1) or (2) will hold at the end of the process. Assume the second step was applied to a pair (V_i, V_j). In this case either $\tilde{d}(V_i, V_j) = 1$ and $d(U_i, U_j) \geq 1 - \epsilon/3$ or $\tilde{d}(V_i, V_j) = 0$ and $d(U_i, U_j) \leq \epsilon/3$. Again, by Claim 1.26, we get that in the former case $d(W_{i,i'}, W_{j,j'}) \geq 1 - \epsilon/3 - \epsilon/6 \geq \epsilon/6$ while in the later $d(W_{i,i'}, W_{j,j'}) \leq \epsilon/3 + \epsilon/6 \leq 1 - \epsilon/6$. If none of the two steps was applied to (V_i, V_j), then

we initially had $|d(V_i,V_j) - d(U_i,U_j)| \leq \epsilon/6$ and $\epsilon/3 \leq d(U_i,U_j) \leq 1-\epsilon/3$. Thus, item (3) holds as in this case we have $\epsilon/6 \leq d(V_i,V_j) = \tilde{d}(V_i,V_j) \leq 1-\epsilon/6$ and by Claim 1.26 for any i',j' we have $\epsilon/6 \leq d(W_{i,i'},W_{j,j'}) \leq 1-\epsilon/6$. □

Claim 1.28. *The graphs G and $\tilde{G}$ differ by less than ϵn^2 edges.*

Proof: As the number of pairs $v \in V_i, v' \in V_j$ is n^2/k^2, and by item (4) of Lemma 1.14 the number of pairs $1 \leq i < j \leq k$ for which $|d(V_i,V_j) - d(U_i,U_j)| > \epsilon/6 = \mathcal{E}(0)$ is at most $\mathcal{E}(0)\binom{k}{2} = \frac{1}{6}\epsilon\binom{k}{2}$, in the first step we changed less than $\frac{1}{6}\epsilon\binom{k}{2}\frac{n^2}{k^2} \leq \frac{1}{6}\epsilon n^2$ edges. In the second stage, if $d(U_i,U_j) < \frac{2}{6}\epsilon$ then by the modifications made in the first step, we have $d(V_i,V_j) < \frac{1}{2}\epsilon$. Similarly if $d(U_i,U_j) > 1-\frac{2}{6}\epsilon$ then by the modifications made in the first step, we have $d(V_i,V_j) > 1-\frac{1}{2}\epsilon$. Thus in this step we make at most $\binom{k}{2}\frac{1}{2}\epsilon(n^2/k^2) \leq \frac{1}{2}\epsilon n^2$ modifications. Finally, in the third step we make at most $k\binom{n/k}{2} \leq n^2/k$ modifications. As we apply Lemma 1.14 with $m = 6/\epsilon$, we have $n^2/k \leq \frac{1}{6}\epsilon n^2$. Altogether, we make less than ϵn^2 modifications. □

We now turn to use the notion of colored-homomorphism, which was introduced in Section 1.3. For the rest of the proof, let R be the following colored complete graph on k vertices. We color $i \in V(R)$ white if V_i is edgeless in $\tilde{G}$. Otherwise (i.e. V_i is a complete graph in $\tilde{G}$, by step (3) in obtaining $\tilde{G}$ from G) we color v_i black. If $\tilde{d}(V_i,V_j) = 0$ we color (i,j) white, if $\tilde{d}(V_i,V_j) = 1$ we color (i,j) black, otherwise (i.e. $\epsilon/6 \leq \tilde{d}(V_i,V_j) \leq 1-\epsilon/6$, by Claim 1.27) we color (i,j) grey. Our goal in the following two claims is to identify a graph $F \in \mathcal{F}$, which we will later show to be abundant in G.

Claim 1.29. *$\tilde{G}$ spans an induced copy of a graph $F' \in \mathcal{F}$. Moreover, $F' \mapsto_c R$.*

Proof: As G is by assumption ϵ-far from being induced $\mathcal{F}$-free, and by Claim 1.28 $\tilde{G}$ is obtained from G by making less than ϵn^2 modifications (of adding and removing edges) $\tilde{G}$ spans an induced copy of a graph $F' \in \mathcal{F}$. We claim that there is a colored-homomorphism from F' to R. Indeed, consider a mapping $\varphi : V(F') \mapsto V(R)$ which maps all the vertices of F' that belong to V_i to vertex i of R. We claim that this is a colored-homomorphism from F' to R. Suppose first that (u,v) is an edge of F'. If u and v belong to the same vertex set V_i, then V_i must be complete in $\tilde{G}$. By definition of φ they are both mapped to $i \in V(R)$ and by our coloring of R, vertex i is colored black. If $u \in V_i$ and $v \in V_j$ then it cannot be the case that $\tilde{d}(V_i,V_j) = 0$, hence $(i,j) \in E(R)$ was not colored white. Similarly, if (u,v) is not an edge of F', then if u and v belong to the same vertex set V_i, then V_i must be edgeless. Hence, vertex i is colored white. If $u \in V_i$ and $v \in V_j$ then it cannot be the case that $d'(V_i,V_j) = 1$, hence $(i,j) \in E(R)$ was not colored black. We thus get that φ satisfies the definition of a colored-homomorphism. □

Claim 1.30. *There is a graph $F \in \mathcal{F}$ of size $f \leq \Psi_{\mathcal{F}}(k)$ for which $F \mapsto_c R$.*

Proof: By Claim 1.29, there is a graph $F' \in \mathcal{F}$ for which $F' \mapsto_c R$. Therefore, R belongs to $\mathcal{F}_k$ (recall Definition 1.22 and the fact that R is of size k). It thus follows from the definition of $\Psi_{\mathcal{F}}$ that $\mathcal{F}$ contains a graph of size at most $\Psi_{\mathcal{F}}(k)$ such that $F \mapsto_c R$. □

The reader may want to recall at this stage that in order to apply Lemma 1.6 with respect to a graph on f vertices we need f distinct vertex sets. The following proposition will enable us to apply Lemma 1.6 on an appropriately chosen f sets of vertices in order to infer that G contains many induced copies of F.

Proposition 1.31. *Let F be the graph from Claim 1.30 and denote its vertex set by $\{1,\ldots,f\}$ with $f \leq \Psi_{\mathcal{F}}(k)$. Let $\varphi : V(F) \mapsto V(R)$ be the colored homomorphism from F to R, which is guaranteed to exist by Claim 1.30, and put $t_i = \varphi(i)$ for every $i \in V(F)$. The following holds with respect to the sets $W_{t_1,1},\ldots,W_{t_f,f}$:*

- *If $(i,j) \in E(F)$ then $d(W_{t_i,i}, W_{t_j,j}) \geq \epsilon/6$.*
- *If $(i,j) \notin E(F)$ then $d(W_{t_i,i}, W_{t_j,j}) \leq 1 - \epsilon/6$.*

Proof: First, note that we choose the sets as $W_{t_1,1},\ldots,W_{t_f,f}$ in order to make sure that we do not choose the same $W_{i,i'}$ twice, because we may need to use several sets $W_{i,j}$ from the same set U_i. Also, observe that as $f \leq \Psi_{\mathcal{F}}(k)$ and we obtained through Lemma 1.9 $\ell = \Psi_{\mathcal{F}}(k)$ sets $W_{i,j}$ from each U_i, we can indeed choose the sets in the above manner, even if all the chosen sets $W_{i,j}$ belong to the same U_i.

Assume that $(i,j) \in E(F)$. As φ is a colored homomorphism from F to R we conclude that either $\varphi(i) = \varphi(j) = t$ and $t \in V(R)$ is colored black or $\varphi(i) = t \neq t' = \varphi(j)$ and $(t,t') \in E(R)$ is colored black or grey. By the way we colored R in the paragraph preceding Claim 1.29 this means that either $\varphi(i) = \varphi(j) = t$ and V_t is a complete graph in $\tilde{G}$ or $\varphi(i) = t \neq t' = \varphi(j)$ and $\tilde{d}(V_t, V_{t'}) \geq \epsilon/6$. Finally, by Claim 1.27 this means that in both cases $d(W_{t_i,i}, W_{t_j,j}) \geq \epsilon/6$. The case of $(i,j) \notin E(F)$ is analogous. $\square$

The proof now follows easily from the above proposition. Consider the sets $W_{t_1,1},\ldots,W_{t_f,f}$ as in Proposition 1.31. By Claim 1.26 any pair of these sets is at least $\gamma_{1.6}(\epsilon/6, \Psi_{\mathcal{F}}(k))$-regular in G. Moreover, by Proposition 1.31, these $f \leq \Psi_{\mathcal{F}}(k)$ sets satisfy in G (**not** in $\tilde{G}$) the edge requirements of Lemma 1.6, which are needed in order to infer that they span many induced copies of F (recall that F has at most $\Psi_{\mathcal{F}}(k)$ vertices). Thus, Lemma 1.6 ensures that $W_{t_1,1},\ldots,W_{t_f,f}$ span in G (**not** in $\tilde{G}$) at least

$$\delta_{1.6}(\epsilon/6, \Psi_{\mathcal{F}}(k)) \cdot \prod_{i=1}^{f} |W_{t_i,i}| \tag{1.11}$$

induced copies of F. We next show that we can take F as the graph in the statement of the lemma. To show this, we should only define the functions $f_{\mathcal{F}}(\epsilon)$ and $\delta_{\mathcal{F}}(\epsilon)$ (the function $N_{\mathcal{F}}(\epsilon)$ is defined in (1.10)). As $|U_i| \geq n/S(\epsilon)$ and $|W_{t_i,i}| \geq \alpha(k)|U_i|$, we conclude from (1.11) that G contains at least

$$\delta_{1.6}(\epsilon/6, \Psi_{\mathcal{F}}(k)) \cdot (\alpha(k)/S(\epsilon))^f \cdot n^f \tag{1.12}$$

induced copies of F. Thus, as $f \leq \Psi_{\mathcal{F}}(k)$, $k \leq S(\epsilon)$ and by the monotonicity properties of all the functions considered in the proof, we can replace k with $S(\epsilon)$ and f with $\Psi_{\mathcal{F}}(S(\epsilon))$

and thus define

$$f_{\mathcal{F}}(\epsilon) = \Psi_{\mathcal{F}}(S(\epsilon)). \tag{1.13}$$

Similarly, we can replace k and f in (1.12) in order to define

$$\delta_{\mathcal{F}}(\epsilon) = \frac{\delta_{1.6}(\epsilon/6, \Psi_{\mathcal{F}}(S(\epsilon)))}{(S(\epsilon)/\alpha(S(\epsilon)))^{\Psi_{\mathcal{F}}(S(\epsilon))}}. \tag{1.14}$$

This completes the proof of Lemma 1.12. □

Before proving Theorem 1.1 we briefly discuss the notions of uniform and non-uniform testing, which will be discussed in Chapter 3. We give here only a rough overview of this chapter and the way it relates to the present one. A tester is defined in Chapter 3 as *non-uniform* if it knows ϵ in advance, and therefore should be able to distinguish between graphs that satisfy $\mathcal{P}$ from those that are ϵ-far from satisfying it (only for that specific ϵ). A tester is *uniform* if it can accept ϵ as part of the input. The main result of Chapter 3 is that there are monotone graph properties, which have non-uniform one-sided testers but cannot be tested by a uniform (one-sided or two-sided) testers. It thus follows that we cannot design uniform testers for all the hereditary graph properties.

Note, that in (1.10),(1.13) and (1.14) the only function, which may be non-computable is $\Psi_{\mathcal{F}}$. Thus whenever this function is computable so are the three functions of Lemma 1.12. As the proof of Theorem 1.1 suggests (see below), once these functions are computable, the tester is uniform. Finally, we note that for any reasonable graph property, and in particular those that were discussed in Subsection 1.1.1, $\Psi_{\mathcal{F}}$ is indeed computable (not necessarily very efficiently). Thus, these properties are testable in the usual sense. We thus assume henceforth that $\mathcal{F}$ is such that the functions $N_{\mathcal{F}}(\epsilon)$, $f_{\mathcal{F}}(\epsilon)$ and $\delta_{\mathcal{F}}(\epsilon)$ are computable. Note however, that even if they are not computable, we still get a non-uniform tester for any (decidable) hereditary graph property.

Proof of Theorem 1.1: We show that any hereditary property can be tested with one-sided error even by an oblivious tester. Fix any hereditary graph property $\mathcal{P}$, and let $\mathcal{F}$ be the family of forbidden induced subgraphs of $\mathcal{P}$ as in Definition 1.11. Let $N_{\mathcal{F}}(\epsilon)$, $f_{\mathcal{F}}(\epsilon)$ and $\delta_{\mathcal{F}}(\epsilon)$ be the functions of Lemma 1.12 and assume they are computable. To design our one-sided error tester for $\mathcal{P}$ we just need to note that if a graph on n vertices contains at least δn^f induced copies of a graph F on f vertices, then sampling $2/\delta$ sets of f vertices each, which is a total of $2f/\delta$, finds an induced copy of F with probability at least $2/3$.

Given a graph G the one-sided error tester for $\mathcal{P}$ works as follows; it asks the oracle for a subgraph of G induced by a randomly chosen set of $\max\{N_{\mathcal{F}}(\epsilon),\ 2f_{\mathcal{F}}(\epsilon)/\delta_{\mathcal{F}}(\epsilon)\}$ vertices. It declares G to be a graph satisfying $\mathcal{P}$ if and only if the induced subgraph on S satisfies $\mathcal{P}$. Clearly, if G satisfies $\mathcal{P}$, then as $\mathcal{P}$ is hereditary the algorithm accepts G with probability 1. If G is ϵ-far from satisfying $\mathcal{P}$ and G has less that $N_{\mathcal{F}}(\epsilon)$ vertices, the algorithm answers correctly with probability 1, as in this case S spans G. If G has more than $N_{\mathcal{F}}(\epsilon)$ vertices, then by Lemma 1.12 there is a member of $\mathcal{F}$ of size $f = f_{\mathcal{F}}(\epsilon)$ such that G spans $\delta_{\mathcal{F}}(\epsilon)n^f$ induced copies of F. By the observation from the preceding paragraph, S spans an induced

copy of F with probability at least $2/3$. As $F \in \mathcal{F}$ and $\mathcal{P}$ is hereditary, we get that with probability at least $2/3$, the graph spanned by S does not satisfy $\mathcal{P}$. Hence, the tester rejects G with probability at least $2/3$. Also, its query complexity is always a function of ϵ only. $\square$

Given the above result we now prove the characterization of the graph properties that can be tested with one-sided error by oblivious testers.

Proof of Theorem 1.4: Let $\mathcal{P}$ be a semi-hereditary property and let $\mathcal{H}$ be the hereditary graph property as in Definition 1.3. We next show that $\mathcal{P}$ has an oblivious one-sided error tester. As $\mathcal{H}$ is hereditary we get from Theorem 1.1 and the fact that its proof actually gives an oblivious tester for $\mathcal{H}$ that there is a function $Q_{\mathcal{H}}(\epsilon)$ such that $\mathcal{H}$ can be tested by an oblivious one-sided error tester with query complexity $Q_{\mathcal{H}}(\epsilon)$. The oblivious tester T we design for testing $\mathcal{P}$ works as follows: its query complexity is $Q(\epsilon) = \max\{M(\epsilon/2), Q_{\mathcal{H}}(\epsilon/2)\}$. After getting from the oracle the randomly chosen induced subgraph, which we denote by G', the tester T proceeds as follows: If G' is of size strictly smaller than $Q(\epsilon)$, the algorithm accepts if and only if G' satisfies $\mathcal{P}$. If G' is of size at least $Q(\epsilon)$ the algorithm accepts if and only if G' satisfies $\mathcal{H}$.

We turn to show that T is indeed an oblivious one-sided error tester for $\mathcal{P}$. We first observe that T satisfies the definition of an oblivious tester. We also note that if the input graph is of size less than $Q(\epsilon)$ then we accept the input if and only if it satisfies $\mathcal{P}$ because by the definition of an oblivious tester this means that the input graph was of size less than $Q(\epsilon)$ and therefore the oracle returned the entire input graph. Let us now consider an input of size at least $Q(\epsilon)$ and recall that $Q(\epsilon) \geq M(\epsilon/2)$. If this input satisfies $\mathcal{P}$ then by the first item of Definition 1.3 it also satisfies $\mathcal{H}$, and as in this case we accept if and only if G' satisfies $\mathcal{H}$ this means that T accepts the input. Hence, T has one-sided error. Suppose now that the input is ϵ-far from satisfying $\mathcal{P}$. This means that after adding/deleting $\frac{1}{2}\epsilon n^2$ edges, the input is still $\frac{\epsilon}{2}$-far from satisfying $\mathcal{P}$. By item 2 of Definition 1.3 and as in this case the input must be of size at least $M(\epsilon/2)$, this means that after adding/deleting $\frac{1}{2}\epsilon n^2$ edges, the input still contains an induced subgraph not satisfying $\mathcal{H}$. In other words, this means that the input is at least $\frac{\epsilon}{2}$-far from satisfying $\mathcal{H}$. As $Q(\epsilon) \geq Q_{\mathcal{H}}(\epsilon/2)$ we infer that with probability at least $2/3$ the graph G' spans an induced subgraph not satisfying $\mathcal{H}$ and therefore G' does not satisfy $\mathcal{H}$ (as it is hereditary). As in this case T accepts if and only if G' satisfies $\mathcal{H}$, this means that T will reject an input that is ϵ-far from satisfying $\mathcal{P}$ with probability at least $2/3$.

Assume now that property $\mathcal{P}$ has a one-sided error oblivious tester T. Our goal is to show the existence of a hereditary property $\mathcal{H}$ as in Definition 1.3. Let $\mathcal{F}$ be the following family of graphs: a graph F on $|V(F)|$ vertices belongs to $\mathcal{F}$ if (i) For some $\epsilon > 0$ the query complexity of T satisfies $Q(\epsilon) = |V(F)|$ (recall that the query complexity of T is a function of ϵ only). (ii) If for this ϵ the sample of vertices spans a graph isomorphic to F, then T rejects the input with positive probability. We claim that we can take $\mathcal{H}$ in Definition 1.3 to be the property of being induced $\mathcal{F}$-free.

To establish the first item of Definition 1.3 it is enough to show that there is no graph G satisfying $\mathcal{P}$, which spans an induced subgraph isomorphic to a graph $F \in \mathcal{F}$. Suppose

such a G exists, and consider the execution of T on G with an ϵ for which $Q(\epsilon) = |V(F)|$. By definition of $\mathcal{F}$ we get that T asks for a random subgraph of G of size $|V(F)|$, and that if T gets a graph isomorphic to F it rejects G with positive probability. As we assume that G spans an induced copy of a graph isomorphic to F, this means that T has a non-zero probability of rejecting G, contradicting our assumption that T is one-sided.

To establish the second item of Definition 1.3, we claim that we can take $M(\epsilon) = Q(\epsilon)$. Indeed, consider a graph G on at least $Q(\epsilon)$ vertices that is ϵ-far from satisfying $\mathcal{P}$. As T is a tester for $\mathcal{P}$ it should reject G with non-zero probability. By definition of an oblivious tester and as G has at least $Q(\epsilon)$ vertices, this means that G must contain an induced subgraph F, of size precisely $Q(\epsilon)$, with the property that if T gets F from the oracle then it rejects G. By definition of $\mathcal{F}$ this means that $F \in \mathcal{F}$. Hence, we can take F itself to be the graph not satisfying $\mathcal{H}$. $\square$

1.5 Concluding Remarks and Open Problems

- Our main result in this chapter can be considered a characterization of the natural graph properties that are testable with one-sided error. Thus, a natural and interesting open problem related to this chapter is to complete the characterization of the graph properties that are testable with one-sided error by arbitrary testers, and not just oblivious ones.

- Theorem 1.1 asserts that any hereditary property is testable with one-sided error. However, the upper bounds on the query complexity, which this theorem guarantees are huge. Even for rather simple properties, these bounds are towers of towers of exponents of height polynomial in $1/\epsilon$. Some specific properties, such as k-colorability, have far more efficient testers, whose query complexity is polynomial in $1/\epsilon$ (see [7]). For others, like being H-free (that is, containing no copy of H as a (not necessarily induced) subgraph), it is known that whenever H is not bipartite, there is no tester (one-sided or two-sided) whose query complexity is polynomial in $1/\epsilon$ (see [1] and Chapter 6). Recall that a hereditary property $\mathcal{P}$ is equivalent to being $\mathcal{F}_\mathcal{P}$-free for a possibly infinite family of graphs $\mathcal{F}_\mathcal{P}$. The hardness of testing hereditary properties for which $\mathcal{F}_\mathcal{P}$ is finite is (relatively) well understood, as it follows from the main result of Chapter 5 that if $\mathcal{F}_\mathcal{P}$ has a graph on at least 5 vertices, then there is no tester (one-sided or two-sided) for $\mathcal{P}$, whose query complexity is polynomial in $1/\epsilon$. When $\mathcal{F}_\mathcal{P}$ is infinite the situation is much more complicated, and there are no general results which guarantee or rule out the possibility of designing testers with query complexity polynomial in $1/\epsilon$. In particular, a natural intriguing and probably challenging problem is the following:

  Which hereditary graph properties can be tested with $poly(1/\epsilon)$ queries?

  As a special case of this problem, it seems interesting to study the query complexity needed to test the natural graph properties that were discussed in Subsection 1.1.1.

- Theorem 1.4 gives a precise characterization of the graph properties that have oblivious one-sided testers. It may thus be simpler, but still very interesting, to resolve the following problem:

  Which graph properties have (possibly two-sided) oblivious testers?

  Note, that the definition of an oblivious tester implicitly assumes that the query complexity of such a tester is a function of ϵ only.

- Fischer and Newman [64] have recently shown that every testable graph property is also estimable, namely, for any such property one can estimate how far is a given graph from satisfying the property (in this chapter this quantity is denoted by ϵ) while making a constant number of queries. Combining Theorem 1.1 and the result of [64] we get that any hereditary property is estimable. See Chapter 7 for more results on this topic.

Chapter 2

Szemerédi Partitions and Two-Sided Error Testers

2.1 The Main Result

2.1.1 Background on Szemerédi's regularity lemma

Our main result in this chapter gives a purely combinatorial characterization of the testable graph properties. As we have previously mentioned, the first properties that were shown to be testable in [75] were certain graph partition properties. As it turns out, our characterization relies on certain "enhanced" partition properties, whose existence is guaranteed by the celebrated regularity lemma of Szemerédi [112]. We start by introducing some standard definitions related to the regularity lemma. For a comprehensive survey about the regularity lemma the reader is referred to [90]. For the convenience of the reader we repeat some definitions that were given in the previous chapter.

For every two nonempty disjoint vertex sets A and B of a graph G, we define $e(A,B)$ to be the number of edges of G between A and B. The *edge density* of the pair is defined by $d(A,B) = e(A,B)/(|A||B|)$.

Definition 2.1. (γ-regular pair) *A pair (A,B) is* γ-regular*, if for any two subsets $A' \subseteq A$ and $B' \subseteq B$, satisfying $|A'| \geq \gamma|A|$ and $|B'| \geq \gamma|B|$, the inequality $|d(A',B') - d(A,B)| \leq \gamma$ holds.*

Throughout the chapter it will be useful to observe that in the above definition it is enough to require that $|d(A',B') - d(A,B)| \leq \gamma$ for sets $A' \subseteq A$ and $B' \subseteq B$, of sizes $|A'| = \gamma|A|$ and $|B'| = \gamma|B|$. A partition $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$ of the vertex set of a graph is called an *equipartition* if $|V_i|$ and $|V_j|$ differ by no more than 1 for all $1 \leq i < j \leq k$ (so in particular every V_i has one of two possible sizes). The *order* of an equipartition denotes the number of partition classes (k above).

Definition 2.2. (γ-regular equipartition) *An equipartition $\mathcal{B} = \{V_i \mid 1 \leq i \leq k\}$ of the vertex set of a graph is called* γ-regular *if all but at most $\gamma\binom{k}{2}$ of the pairs (V_i, V_j) are γ-regular.*

In what follows an equipartition is said to *refine* another if every set of the former is contained in one of the sets of the latter. Szemerédi's regularity lemma can be formulated as follows[1].

Lemma 2.3 ([112]). *For every m and $\gamma > 0$ there exists $T = T_{1.8}(m, \gamma)$ with the following property: If G is a graph with $n \geq T$ vertices, and $\mathcal{A}$ is any equipartition of the vertex set of G of order at most m, then there exists a refinement $\mathcal{B}$ of $\mathcal{A}$ of order k, where $m \leq k \leq T$ and $\mathcal{B}$ is γ-regular. In particular, for every m and $\gamma > 0$ there exists $T = T_{1.8}(m, \gamma)$ such that any graph with $n \geq T$ vertices, has a γ-regular equipartition of order k, where $m \leq k \leq T$.*

The regularity lemma guarantees that every graph has a γ-regular equipartition of (relatively) small order. As it turns out in many applications of the regularity lemma, one is usually interested in the densities of the bipartite graphs connecting the sets V_i of the regular partitions. In fact, one important consequence of the regularity lemma is that in many cases knowing the densities connecting the sets V_i (approximately) tells us all we need to know about a graph. Roughly speaking, if a graph G has a regular partition of order k and we define a weighted graph $R(G)$, of size k, where the weight of edge (i, j) is $d(V_i, V_j)$, then by considering an appropriate property of $R(G)$ one can infer many properties of G. As the order of the equipartition is guaranteed to be bounded by a function of γ, this means that for many applications, every graph has an approximate description of *constant-complexity* (we will return to this aspect in a moment). As it turns out, this interpretation of the regularity lemma is the key to our characterization. We believe that our characterization of the testable graph properties is an interesting application of this aspect of the regularity lemma.

Given the above discussion it seems natural to define a graph property, which states that a graph has a given γ-regular partition, that is, an equipartition which is γ-regular and such that the densities between the sets V_i belong to some predefined set of densities.

Definition 2.4 (Regularity-Instance). *A* regularity-instance *is given by an error-parameter $0 < \gamma \leq 1$, an integer k, a set of $\binom{k}{2}$ densities $0 \leq \eta_{ij} \leq 1$ indexed by $1 \leq i < j \leq k$, and a set $\overline{R}$ of pairs (i, j) of size at most $\gamma\binom{k}{2}$. A graph is said to* satisfy *the regularity-instance if it has an equipartition $\{V_i \mid 1 \leq i \leq k\}$ such that for all $(i, j) \notin \overline{R}$ the pair (V_i, V_j) is γ-regular and satisfies $d(V_i, V_j) = \eta_{i,j}$. The* complexity *of the regularity-instance is* $\max(k, 1/\gamma)$.

Note, that in the above definition the set $\overline{R}$ corresponds to the set of pairs (i, j) for which (V_i, V_j) is not necessarily a γ-regular pair (possibly, there are at most $\gamma\binom{k}{2}$ such pairs). Also, note that the definition of a regularity-instance does not impose any restriction on the graphs spanned by any single set V_i. By Theorem 1.8, for any $0 < \gamma \leq 1$ any graph satisfies *some* regularity instance with an error parameter γ and with an order bounded by a function γ. The first step needed in order to obtain our characterization of the testable properties, is that the property of satisfying any given regularity-instance is testable. This is also the main technical result of this chapter.

[1]Note that the formulation here is a little different from the one given in the previous chapter. This formulation is mainly useful for the proof of Corollary 2.29.

Theorem 2.5. *For any regularity-instance R, the property of satisfying R is testable.*

2.1.2 The characterization

Many of the recent results on testing graph properties in the dense graph model relied on Lemma 1.8. Our main result in this chapter shows that this is not a coincidence. Each of the papers which applied the regularity lemma to test a graph property used different aspects of what can be inferred from certain properties of a regular partition of a graph. These results however, use the properties of the regular partition in an *implicit* way. For example, the main observation needed in order to infer that triangle-freeness is testable, is that if the regular partition has three sets V_i, V_j, V_k, which are connected by regular and dense bipartite graphs, then the graph contains many triangles. However, to *test* triangle freeness we do not need to know the regular partition, we just need to find a triangle in the graph. As Theorem 2.5 allows us to test for having a certain regular partition, it seems natural to try and test properties by *explicitly* checking for properties of the regular partition of the input. Returning to the previous discussion on viewing the regularity lemma as a constant complexity description of a graph, being able to explicitly test for having a given regular partition should allow us to test more complex properties as we can obtain all the information of the regular partition and not just *consequences* of having some regular partition. The next definition tries to capture the graph properties $\mathcal{P}$ that can be tested via testing a certain set of regularity instances.

Definition 2.6 (Regular-Reducible)**.** *A graph property $\mathcal{P}$ is* regular-reducible *if for any $\delta > 0$ there exists $r = r(\delta)$ such that for any n there is a family $\mathcal{R}$ of at most r regularity-instances each of complexity at most r, such that the following holds for every n-vertex graph G:*

1. *If G satisfies $\mathcal{P}$ then for some $R \in \mathcal{R}$, G is δ-close to satisfying R.*
2. *If G is ϵ-far from satisfying $\mathcal{P}$, then for any $R \in \mathcal{R}$, G is $(\epsilon - \delta)$-far from satisfying R.*

The reader may observe that in the above definition the value of δ may be arbitrarily close to 0. If we think of $\delta = 0$ then we get that a graph satisfies $\mathcal{P}$ if and only if it satisfies one of the regularity instances of $\mathcal{R}$. With this interpretation in mind, in order to test $\mathcal{P}$ one can test the property of satisfying any one of the instances of $\mathcal{R}$. Therefore, in some sense we "reduce" the testing of property $\mathcal{P}$ to the testing of regularity-instances. As the main result of this chapter states, the testable graph properties are precisely those for which testing them can be carried out by testing for some property of their regular partitions.

Theorem 2.7 (Main Result)**.** *A graph property is testable* **if and only if** *it is regular-reducible.*

If we have to summarize the moral of our characterization in one simple sentence, then it says that a graph property $\mathcal{P}$ is testable if and only if $\mathcal{P}$ is such that knowing a regular partition of a graph G is sufficient for telling whether G is ϵ-far or ϵ-close to satisfying $\mathcal{P}$.

In other words, there is a short "proof" that G is either ϵ-close or ϵ-far from satisfying $\mathcal{P}$. Thus, in a more "computational complexity" jargon, we could say that a graph property is testable if and only if it has the following "interactive proof": A prover gives a verifier the description of a regularity-instance R, which the input G is (supposedly) close to satisfying. The verifier, using Theorem 2.5, then verifies if G is indeed close to satisfying R. The way to turn this interactive proof into a testing algorithm is to apply the constant-complexity properties of the regularity lemma that we have previously discussed; as the order of the regular partition is bounded by a function of ϵ, there are only *finitely* many regularity-instances that the prover may potentially send to the verifier. Therefore, the verifier does not need to get an alleged regular-instances, it can simply try them all! Theorem 2.7 thus states that in some sense testing regularity-instances is the "hardest" property to test, because by Theorem 2.7 any testing algorithm can be turned into a testing algorithm for regularity-instances. However, we stress that this is true only on the *qualitative* level, because using Theorem 2.7 in order to turn a tester into a tester, which tests for regularity-instances, may significantly increase its query complexity. The main reason is that the proofs of Theorems 2.5 and 2.7 apply Lemma 1.8 and thus only give weak upper bounds. We also note that the terminology of regular-reducible is not far from being a standard reduction because in order to prove one of the directions of Theorem 2.7 we indeed test a property $\mathcal{P}$, which is regular-reducible to a set $\mathcal{R}$, by testing the regularity-instances of $\mathcal{R}$. Theorem 2.7 also gives further convincing evidence to the "combinatorial" nature of property testing in the dense graph model as was recently advocated by Goldreich [81].

As is evident from Definition 2.6, the characterization given in Theorem 2.7 is not a "quick recipe" for inferring whether a given property is testable. Still, we can use Theorem 2.7 in order to obtain unified proofs for several previous results. As we have alluded to before, these results can be inferred by showing that it is possible (or impossible) to reduce the testing of the property to testing if a graph satisfies certain regularity-instances. We believe that these proofs give some (non-explicit) structural explanation as to what makes a graph property testable. See Section 2.6 for more details. It is thus natural to ask if one can come up with more "handy" characterizations. We doubt that such a characterization exists, mainly because it should (obviously) be equivalent to Theorem 2.7. One supporting evidence is a recent related study of graph homomorphism [37] that led to a different characterization, which is also somewhat complicated to apply.

2.1.3 Organization and overview

The first main technical step of the proof of Theorem 2.5 is taken in Section 2.2. In this section we prove that if the densities of pairs of subsets of vertices of a bipartite graph are close to the density of the bipartite graph itself, then the bipartite graph can be turned into a regular-pair using relatively few edge modifications. Rephrasing this gives that we can increase the regularity measure of a bipartite pair by making relatively few edge modifications. The second main step is taken in Section 2.4. In this section, we show that sampling a constant number of vertices guarantees that the sample and the graph will have (roughly) the *same* set of regular partitions. We believe that this result may be of

independent interest. By applying the results of Sections 2.2 and 2.4 we prove Theorem 2.5 in Section 2.5. In this section we also prove one of the directions of Theorem 2.7, asserting that if a graph property is regular-reducible then it is testable. Along with Theorem 2.5, a second tool that we need in order to prove this direction is the main result of [64]. We apply this result in order to infer that for any regularity-instance R, one can not only test the property of satisfying R, but can also estimate how far is a given graph from satisfying R. This *estimation* of the distance to satisfying regularity-instances is key to *testing* a property via a regularity-reduction. The proof of the second direction of Theorem 2.7 appears in Section 2.3. To prove this direction we first show that knowing that a graph G satisfies a regularity instance enables us to estimate the number of copies of certain graphs in G. We then apply the main result of [77] about canonical testers along with the main result of Section 2.2 in order to "pick" those regularity-instances that can constitute the family $\mathcal{R}$ in Definition 2.6. In Section 2.6 we use Theorem 2.7 in order to reprove some previously known results in property-testing. The main interest of these proofs is that they apply Theorem 2.7 in order to prove in a unified manner results that had distinct proofs. Section 2.7 contains some concluding remarks.

2.2 Enhancing Regularity with Few Edge Modifications

The definition of a γ-regular pair of density η requires a pair of sets of vertices to satisfy several density requirements. Our main goal in this section is to show that if a pair of vertex sets are close (in an appropriate sense) to satisfying these requirements, then it is indeed close to being a γ-regular pair of density η. For example, consider the property of being a 0.1-regular pair with edge density 0.5. Intuitively, it seems that if the edge density of a bipartite graph G on vertex sets A and B of size m each is close to 0.5, and the density of any pair $A' \subseteq A$ and $B' \subseteq B$ of sizes $0.1m$ is close to 0.5 ± 0.1, then G should be close to satisfying the property. However, note that it may be the case that there are pairs (A', B'), whose density is smaller than 0.4, and other pairs, whose density is larger than 0.6. Thus, only removing or only adding edges (even randomly) will most likely not turn G into a 0.1-regular pair of density 0.5. In order to show that G is indeed close to satisfying the property, we take a "convex combination" of G with a random graph, whose density is 1/2. The intuition is that the random graph will not change the density of G much, but, because a random graph is highly regular, it will increase the regularity of G. The main result of this section is formalized in the following lemma, which is an important ingredient in the proofs of both directions of Theorem 2.7.

In this lemma, as well as throughout the rest of the chapter, when we write $x = a \pm b$ we mean $a - b \leq x \leq a + b$.

Lemma 2.8. *The following holds for any $0 < \delta \leq \gamma \leq 1$: Suppose that (A, B) is a $(\gamma + \delta)$-regular pair with density $\eta \pm \delta$, where $|A| = |B| = m \geq m_{2.8}(\eta, \delta)$. Then, it is possible to make at most $50\frac{\delta}{\gamma^2}m^2$ edge modifications and turn (A, B) into a γ-regular pair with density precisely η.*

The proof of Lemma 2.8 has two main steps, which are captured in Lemmas 2.9 and 2.10

below. The first step is given in the following lemma, which enables us to make relatively few edge modifications and thus make sure that the density of a pair is exactly what it should be, while at the same time not decreasing its regularity by much.

Lemma 2.9. *Suppose that (A, B) is a $(\gamma+\delta)$-regular pair satisfying $d(A, B) = \eta \pm \delta$, where $|A| = |B| = m \geq m_{2.9}(\eta, \delta)$. Then, it is possible to make at most $2\delta m^2$ modifications, and thus turn (A, B) into a $(\gamma + 2\delta)$-regular pair with density precisely η.*

The second and main step, which implements the main idea presented at the beginning of this section, takes a bipartite graph, whose density is precisely η, and returns a bipartite graph, whose density is still η but with a better regularity measure.

Lemma 2.10. *The following holds for any $0 < \delta \leq \gamma \leq 1$. Let A and B be two vertex sets of size $m \geq m_{2.10}(\delta, \gamma)$, satisfying $d(A, B) = \eta$. Suppose further that for any pair of subsets $A' \subseteq A$ and $B' \subseteq B$ of size γm we have $d(A', B') = \eta \pm (\gamma + \delta)$. Then, it is possible to make at most $\frac{3\delta}{\gamma}m^2$ edge modifications and thus turn (A, B) into a γ-regular pair with density precisely η.*

We now turn to prove the above three lemmas. Following them is a corollary of Lemma 2.8, which will be used in the proof of Theorem 2.7. For the proofs of this section we need the following standard Chernoff-type large deviation inequality.

Lemma 2.11. *Suppose $X_1, \ldots, X_n$ are n independent Boolean random variables, where $Prob[X_i = 1] = p_i$. Let $E = \sum_{i=1}^{n} p_i$. Then, $Prob[|\sum_{i=1}^{n} X_i - E| \geq \delta n] \leq 2e^{-2\delta^2 n}$.*

Proof (of Lemma 2.9): Suppose that $d(A, B) = \eta + p$, where $|p| \leq \delta$, and assume for now that $p \geq 0$. Suppose first that $p \leq \delta(\gamma + 2\delta)^2$. In this case we just remove any $pm^2 (\leq \delta m^2)$ edges and thus make sure that $d(A, B) = \eta$. Furthermore, as for any pair (A', B') of size $(\gamma + 2\delta)m$ we initially had $d(A', B') = \eta + p \pm (\gamma + \delta)$, it is easy to see that because we remove $pm^2 \leq \delta(\gamma + 2\delta)^2 m^2$ edges, we now have $\eta - \gamma - 2\delta \leq d(A', B') \leq \eta + \gamma + \delta$, which satisfies $d(A', B') = \eta \pm (\gamma + 2\delta)$. Thus, in this case we turned (A, B) into a $(\gamma + 2\delta)$-regular pair of density η.

Suppose now that $p \geq \delta(\gamma + 2\delta)^2$. Our way for turning (A, B) into a $(\gamma + 2\delta)$-regular pair with density η will consist of two stages. In the first we will randomly remove some of the edges connecting A and B. We will then deterministically make some additional modifications. To get that after these two stages (A, B) has the required properties we show that with probability $3/4$ the pair (A, B) is $(\gamma + 2\delta)$-regular and with the same probability $d(A, B) = \eta$. By the union bound we will get that with probability at least $1/2$ the pair (A, B) has the required two properties.

In the first (random) step, we remove each of the edges connecting A and B randomly and independently with probability $\frac{p}{\eta+p}$. Then, the expected number of edges removed is $\frac{p}{\eta+p}(\eta + p)|A||B| = p|A||B| \leq \delta|A||B|$, and the expected value of $d(A, B)$ is η. As we assumed that $p \geq \delta(\gamma + 2\delta)^2$ we have $d(A, B) \geq \delta(\gamma + 2\delta)^2$. Therefore, the number of edges we may randomly remove is at least $\delta(\gamma + 2\delta)^2 m^2$. Therefore, by Lemma 2.11, for large enough $m \geq m_{2.9}(\delta, \gamma)$, the probability that $d(A, B)$ deviates from η by more than $m^{-0.5}$ is

at most 3/4. In particular, the number of edge modifications made is at most $\frac{3}{2}\delta m^2$ with probability at least 3/4. Now (this is the second, deterministic step) we can add or remove at most $m^{1.5}$ edges arbitrarily and thus make sure that $d(A,B) = \eta$. The total number of edge modifications is also at most $\frac{3}{2}\delta m^2 + m^{1.5} \leq 2\delta m^2$, for large enough $m \geq m_{2.9}(\delta,\gamma)$. Note that we have thus established that with probability at least 3/4 after the above two stages $d(A,B) = \eta$.

As (A,B) was assumed to be $(\gamma+\delta)$-regular, we initially had $d(A',B') = \eta + p \pm (\gamma+\delta)$ for any pair of subsets $A' \subseteq A$ and $B' \subseteq B$ of size $(\gamma+2\delta)$. As in the first step we removed each edge with probability $\frac{p}{\eta+p}$, the expected value of $d(A'B')$ after the first step is between

$$(\eta+p+\gamma+\delta)(1-\frac{p}{\eta+p}) \leq \eta+\gamma+\delta$$

and

$$(\eta+p-\gamma-\delta)(1-\frac{p}{\eta+p}) \geq \eta-\gamma-\delta.$$

Recall that we have already established that with probability at least 3/4 we have $d(A,B) = \eta$ and that for any pair (A',B') of size $(\gamma+2\delta)m$ the expected value of $d(A',B')$ is $\eta \pm (\gamma+\delta)$. Hence, to show that after the two steps (A,B) is a $(\gamma+2\delta)$-regular pair with probability at least 1/2, it suffices to show that with probability at least 3/4, the densities of all pairs (A',B') do not deviate from their expectation by more than δ.

Suppose first that $d(A',B')$ was originally at most $\frac{1}{2}\delta$. This means that when we randomly remove edges from (A,B) we can change $d(A',B')$ by at most $\frac{1}{2}\delta$. Thus in this case $d(A',B')$ can deviate from its expectation by at most $\frac{1}{2}\delta$. Also, when adding or removing $m^{1.5}$ edges to (A,B) in the second step we can change $d(A',B')$ by at most $m^{-0.5}/(\gamma+2\delta)^2 \leq \frac{1}{2}\delta$ for large enough $m \geq m_{2.9}(\delta,\gamma)$. Thus, for such pairs we are guaranteed that $d(A',B') = \eta \pm (\gamma+2\delta)$.

Suppose now that $d(A',B')$ was at least $\frac{1}{2}\delta$. Thus the number of edges, which were considered for removal between A' and B' in the first step was at least $\frac{1}{2}\delta(\gamma+2\delta)^2 m^2$. Hence, by Lemma 2.11 the probability that $d(A',B')$ deviates from its expectation by more than $\frac{1}{2}\delta$ is at most $2e^{-2(\frac{1}{2}\delta)^2 \frac{1}{2}\delta(\gamma+2\delta)^2 m^2}$. Thus, as there are at most 2^{2m} pairs of such sets (A',B'), we conclude by the union-bound that for large enough $m \geq m_{2.9}(\delta,\gamma)$, with probability at least 3/4 all sets (A',B') of size $(\gamma+2\delta)m$ satisfy $d(A',B') = \eta \pm (\gamma+\frac{3}{2}\delta)$. As in the previous paragraph, adding or removing $m^{1.5}$ edges in the second step can change $d(A',B')$ by at most $\frac{1}{2}\delta$, so in this case we also have $d(A',B') = \eta \pm (\gamma+2\delta)$.

Finally, in the case that p above is negative we can use essentially the same argument. The only modification is that we add edges instead of remove them. □

Proof (of Lemma 2.10): For any vertex $a \in A$ and $b \in B$ we do the following: we flip a coin with bias $\frac{2\delta}{(\delta+\gamma)}$. If the coin comes up heads we make no modification between the vertices a and b. If the coin comes up tails then we disregard the adjacency relation between a and b and do the following: we flip another coin with bias η. If the coin comes up heads then we connect a and b, and otherwise we leave them disconnected. In what follows we call the coins flipped in the first step the *first* coins, and those flipped in the second step

the *second* coins.

Claim: With probability at least 3/4, we make at most $\frac{3\delta}{\gamma}m^2$ edge modifications.

Proof. Note that the number of edge modifications is at most the number of first coins that came up heads. The distribution of these m^2 coins is given by the Binomial distribution $B(m^2, \frac{2\delta}{(\delta+\gamma)})$, whose expectation is $\frac{2\delta}{(\delta+\gamma)}m^2$, and by Lemma 2.11 the probability of deviating by more than $\frac{1}{2}\delta m^2$ from this expectation is at most $2e^{-2(\delta/2)^2 m^2}$. For large enough $m \geq m_{2.10}(\delta, \gamma)$ we get that with probability at least 3/4 we make at most $\frac{2\delta}{(\delta+\gamma)}m^2 + \frac{1}{2}\delta m^2 \leq \frac{2.5\delta}{\gamma}m^2$ modifications. □

The following observation will be useful for the next two claims: Fix a pair of connected vertices $a \in A$ and $b \in B$. For them to become disconnected both coins must come up tails, thus the probability of them staying connected is $(1 - \frac{2\delta}{(\delta+\gamma)} + \frac{2\eta\delta}{(\delta+\gamma)})$. Now, fix a pair of disconnected vertices $a \in A$ and $b \in B$. For them to become connected the first coin must come up tails and the second must come up heads, so the probability of them becoming connected is $\frac{2\eta\delta}{(\delta+\gamma)}$.

Claim: With probability at least 3/4, we have $d(A, B) = \eta \pm m^{-0.5}$.

Proof. Recall that by assumption the number of connected vertices was ηm^2. Thus, by the above observation the expected number of connected vertices is

$$\eta m^2(1 - \frac{2\delta}{(\delta+\gamma)} + \frac{2\eta\delta}{(\delta+\gamma)}) + (1-\eta)m^2\frac{2\eta\delta}{(\delta+\gamma)} = \eta m^2.$$

By Lemma 2.11 we get that for large enough $m \geq m_{2.10}(\delta, \gamma)$ the probability of deviating from this expectation by more than $m^{-0.5}$ is at most 1/4. □

Claim: With probability at least 3/4, all sets $A' \subseteq A$ and $B' \subseteq B$ of size γm satisfy $d(A', B') = \eta \pm (\gamma - \frac{1}{2}\delta)$.

Proof. Fix any pair of such sets. Let e denote the number of edges originally spanned by these sets. As in the previous claim we get that the expected number of edges spanned by (A', B') is

$$e(1 - \frac{2\delta}{(\delta+\gamma)} + \frac{2\eta\delta}{(\delta+\gamma)}) + (|A'||B'| - e)\frac{2\eta\delta}{(\delta+\gamma)} = e(1 - \frac{2\delta}{(\delta+\gamma)}) + |A'||B'|\frac{2\eta\delta}{(\delta+\gamma)}.$$

Recall that by assumption $e = |A'||B'|(\eta \pm (\gamma + \delta))$. Thus, the expected number of edges spanned by (A', B') is at most

$$|A'||B'|(\eta+\gamma+\delta)(1 - \frac{2\delta}{(\delta+\gamma)}) + |A'||B'|\frac{2\eta\delta}{(\delta+\gamma)} = |A'||B'|(\eta + \gamma + \delta - \frac{2\delta\gamma}{\delta+\gamma} - \frac{2\delta^2}{\delta+\gamma}) =$$

$$|A'||B'|(\eta + \gamma - \delta),$$

Similarly, we infer that the expected number of edges spanned by (A', B') is at least

$$|A'||B'|(\eta-\gamma-\delta)(1-\frac{2\delta}{(\delta+\gamma)})+|A'||B'|\frac{2\eta\delta}{(\delta+\gamma)}=|A'||B'|(\eta-\gamma-\delta+\frac{2\delta\gamma}{\delta+\gamma}+\frac{2\delta^2}{\delta+\gamma})=$$

$$|A'||B'|(\eta-\gamma+\delta)\ .$$

By Lemma 2.11 the probability that the number of edges between A' and B' will deviate from its expectation by more than $\frac{1}{2}\delta|A'||B'|$ is at most $2e^{-2(\delta/2)^2|A'||B'|}=2e^{-2(\delta/2)^2(\gamma m)^2}$. As the number of pairs (A', B') is at most 2^{2m} we get by the union bound, provided that $m \geq m_{2.10}(\delta,\gamma)$ is large enough, that with probability at least 3/4 all the pairs (A', B') of size γm satisfy this property. Thus for all pairs (A', B') of size γm we have $d(A', B') = \eta \pm (\gamma - \frac{1}{2}\delta)$. □

Combining the above three claims we get that with constant probability we make at most $\frac{2.5\delta}{\gamma}m^2$ modifications and thus make sure that $d(A, B) = \eta \pm m^{-0.5}$ and furthermore that for any pair of sets (A', B') of size γm we have $d(A', B') = \eta \pm (\gamma - \frac{1}{2}\delta)$. Now we can add or remove at most $m^{1.5}$ edges to make sure that $d(A, B) = \eta$. For any pair of sets (A', B') of size γm this will change $d(A', B')$ by at most $m^{-0.5}/\gamma^2 \leq \frac{1}{2}\delta$ for large enough m. This means that we will have $d(A', B') = \eta \pm \gamma$, implying that (A, B) is γ-regular with density η, completing the proof of the lemma. □

Proof (of Lemma 2.8): By Lemma 2.9 we can make at most $2\delta m^2$ edge modifications and thus turn (A, B) into a $(\gamma+2\delta)$-regular pair with density η. Thus, every pair of subsets $A'' \subseteq A$ and $B'' \subseteq B$ of size γm has density at most

$$(\eta+\gamma+2\delta)(\gamma+2\delta)^2m^2/\gamma^2m^2 \leq (\eta+\gamma+2\delta)(1+8\delta/\gamma) \leq \eta+\gamma+14\delta/\gamma.$$

Similarly, the density of such a pair is at least $\eta - \gamma - 14\delta/\gamma$. We thus conclude that (A, B) has density precisely η, and every pair of subsets (A'', B'') of size γm has density $\eta\pm(\gamma+14\delta/\gamma)$. Now we can use Lemma 2.10 to make at most $3\frac{14\delta/\gamma}{\gamma}m^2 = 42\frac{\delta}{\gamma^2}m^2$ additional edge modifications and thus turn (A', B') into γ-regular pair with density precisely η. The total number of modifications is $42\frac{\delta}{\gamma^2}m^2 + 2\delta m^2 \leq 50\frac{\delta}{\gamma^2}m^2$ as needed. □

We finish this section with the following application of Lemma 2.8 that will be useful later in the chapter.

Corollary 2.12. *Let R be a regularity-instance of order k, error-parameter γ, $\binom{k}{2}$ edge densities $\eta_{i,j}$ and set of non-regular pairs $\overline{R}$. If a graph G has an equipartition $\mathcal{V} = \{V_1, \ldots, V_k\}$ of order k such that*

1. *$d(V_i, V_j) = \eta_{i,j} \pm \frac{\gamma^2\epsilon}{50}$ for all $i < j$.*
2. *Whenever $(i, j) \notin \overline{R}$, the pair (V_i, V_j) is $(\gamma + \frac{\gamma^2\epsilon}{50})$-regular.*

Then G is ϵ-close to satisfying R.

Proof: For any $(i,j) \notin \overline{R}$ we can use Lemma 2.8 and make at most $50\frac{\gamma^2\epsilon/50}{\gamma^2}(n/k)^2 \leq \epsilon n^2/k^2$ edge modifications to turn (V_i, V_j) into a γ-regular pair with density $\eta_{i,j}$. As there are at most $\binom{k}{2}$ pairs this is a total of at most ϵn^2 modifications. We have thus turned G into a graph satisfying R by making at most ϵn^2 edge modifications, as needed. □

2.3 Any Testable Property is Regular-Reducible

In this section we prove the first direction of Theorem 2.7.

Lemma 2.13. *If a graph property is testable then it is regular-reducible.*

Our starting point in the proof of Lemma 2.13 is the following result of [77] (extending a result of [6]) about canonical testers:

Lemma 2.14 ([6, 77])**.** *If a graph property $\mathcal{P}$ can be tested on n-vertex graphs with $q = q(\epsilon, n)$ edge queries, then it can also be tested by a tester, which makes its queries by uniformly and randomly choosing a set of $2q$ vertices, querying all the pairs and then accepting or rejecting (deterministically) according to the graph induced by the sample, the value of ϵ and the value of n. In particular, it is a non-adaptive tester making $\binom{2q}{2}$ queries.*

Restating the above, by (at most) squaring the query complexity, we can assume without loss of generality that a property-tester works by sampling a set of vertices of size $q(\epsilon, n)$ and accepting or rejecting according to some graph property of the sample. As noted in [77], the graph property that the algorithm may search for in the sample may be different from the property, which is tested. In fact, the property the algorithm checks for in the sample may depend on ϵ and on the size of the input graph. Our main usage of Lemma 2.14 is that it allows to pick the graphs of size q that cause a tester for $\mathcal{P}$ to accept. The first technical step that we take towards proving Lemma 2.13 is proving some technical results about induced copies of graphs spanned by graphs satisfying a given regularity-instance. These results enable us to deduce from the fact that a graph satisfies some regularity-instance the probability that a given tester accepts the graph. We then use these results along with Lemma 2.14 and some additional arguments in order to prove that any testable property is regular reducible. The details follow.

Definition 2.15. *Let H be a graph on h vertices, let W be a weighted complete graph on h vertices, where the weight of an edge (i,j) is $\eta_{i,j}$. For a permutation $\sigma : [h] \to [h]$ define*

$$IC(H, W, \sigma) = \prod_{(i,j)\in E(H)} \eta_{\sigma(i),\sigma(j)} \prod_{(i,j)\notin E(H)} (1 - \eta_{\sigma(i),\sigma(j)})$$

Suppose $V_1, \ldots, V_k$ are k vertex sets, each of size m, and suppose the bipartite graph spanned by V_i and V_j is a bipartite random graph with edge density $\eta_{i,j}$. Let H be a graph of size k, and let $\sigma : [k] \to [k]$ be some permutation. What is the expected number of k-tuples of vertices $v_1 \in V_1, \ldots, v_k \in V_k$, which span an induced copy of H with each v_i playing the role of $\sigma(i)$? It is easy to see that the answer is $IC(H, W, \sigma)m^k$, where W

is the weighted complete graph with weights $\eta_{i,j}$. The following claim shows that this is approximately the case when instead of random bipartite graphs we take regular enough bipartite graphs. The proof is a standard application of the definition of a regular pair and is thus omitted from this extended abstract. See Lemma 4.2 in [61] for a version of the proof.

Claim 2.16. *For any δ and h, there exists a $\gamma = \gamma_{2.16}(\delta, h)$ such that the following holds: Suppose $V_1, \ldots, V_h$ are h sets of vertices of size m each, and that all the pairs (V_i, V_j) are γ-regular. Define W to be the weighted complete graph on h vertices, whose weights are $\eta_{i,j} = d(V_i, V_j)$. Then, for any graph H on h vertices and for any $\sigma : [k] \to [k]$, the number of h-tuples $v_1 \in V_1, \ldots, v_h \in V_h$, which span an induced copy of H with each v_i playing the role of the vertex $\sigma(i)$ is*

$$(IC(H, W, \sigma) \pm \delta)m^h$$

We would now want to consider the total number of induced copies of some graph.

Definition 2.17. *Let H be a graph on h vertices, let W be a weighted complete graph on h vertices, where the weight of edge (i, j) is $\eta_{i,j}$. Let $Aut(H)$ denote the number of automorphisms of H. Define*

$$IC(H, W) = \frac{1}{Aut(H)} \sum_{\sigma} IC(H, W, \sigma).$$

Continuing the discussion before Claim 2.16, it is easy to see that in this case the expected number of induced copies of H having one vertex in each of the sets V_i is $IC(H, W)$. Again, we can show that the same is approximately true when we replace random bipartite graphs with regular enough bipartite graphs.

Claim 2.18. *For any δ and k, there exists a $\gamma = \gamma_{2.18}(\delta, k)$ such that the following holds: Suppose that $V_1, \ldots, V_k$ are sets of vertices of size m each, and that all the pairs (V_i, V_j) are γ-regular. Define K to be the weighted complete graph on k vertices, whose weights are $\eta_{i,j} = d(V_i, V_j)$. Then, for any graph H of size k, the number of induced copies of H, which have precisely one vertex in each of the sets $V_1, \ldots, V_k$ is*

$$(IC(H, W) \pm \delta)m^k$$

Proof. Set $\gamma_{2.18}(\delta, k) = \gamma_{2.16}(\delta/k!, k)$. Suppose $V_1, \ldots, V_k$ are as in the statement of the claim and let H by any graph on k vertices. By Claim 2.16 for every permutation $\sigma : [k] \to [k]$, the number of induced copies of H which have precisely one vertex v_i in each set V_i such that v_i plays the role of vertex $\sigma(i)$ is $IC(H, W, \sigma) \pm \delta m^k/k!$. If we sum over all permutations $\sigma : [k] \to [k]$ we get $\sum_\sigma (IC(H, W, \sigma) \pm \delta/k!)m^k$. This summation, however, counts copies of H several times. More precisely, each copy is thus counted $Aut(H)$ times.

Thus, dividing by $Aut(H)$ gives that the number of such induced copies is

$$\begin{aligned}\frac{1}{Aut(H)}\left(\sum_{\sigma}(IC(H,W,\sigma)\pm\delta/k!\right)m^k) &= \left(\frac{1}{Aut(H)}\sum_{\sigma}IC(H,W,\sigma)\pm\delta\right)m^k \\ &= (IC(H,W)\pm\delta)m^k \quad \square\end{aligned}$$

We would now want to consider the number of induced copies of a graph H, when the number of sets V_i is larger than the size of H.

Definition 2.19. *Let H be a graph on h vertices, let R be a weighted complete graph of size at least h where the weight of an edge (i,j) is $\eta_{i,j}$, and let $\mathcal{W}$ denote all the subsets of $V(W)$ of size h. Define*

$$IC(H,R)=\sum_{W\in\mathcal{W}}IC(H,W).$$

The following lemma shows that knowing that a graph satisfies some regularity-instance R, enables us to estimate the number of induced copies spanned by any graph, which satisfies R.

Lemma 2.20. *For any δ and q, there are $k=k_{2.20}(\delta,q)$ and $\gamma=\gamma_{2.20}(\delta,q)$ with the following properties: For any regularity-instance R of order at least k and with error parameter at most γ, and for every graph H of size $h\leq q$, the number of induced copies of H in any n-vertex graph satisfying R is*

$$(IC(H,R)\pm\delta)\binom{n}{h}$$

Proof. Put $k=k_{2.20}(\delta,q)=\frac{\delta}{10q^2}$ and $\gamma=\gamma_{2.20}(\delta,q)=\min\{\frac{\delta}{3q^2},\gamma_{2.18}(\frac{1}{3}\delta,q)\}$. Let R be any regularity instance as in the statement, let G be any graph satisfying R, and let H be any graph of size $h\leq q$. Let $V_1,\ldots,V_\ell$ be an equipartition of G satisfying R. For the proof of the lemma it will be simpler to consider an equivalent statement of the lemma, stating that if one samples an h-tuple of vertices from G, then the probability that it spans an induced copy of H is $IC(H,R)\pm\delta$.

First, note that by our choice of k we get from a simple birthday-paradox argument, that the probability that the h-tuple of vertices has more than one vertex in any one of the sets V_i is at most $\frac{1}{3}\delta$. Second, observe that as the equipartition of R is γ-regular and $\gamma\leq\delta$, we get that the probability that the h-tuple of vertices contains a pair $v_i\in V_i$ and $v_j\in V_j$ such that (V_i,V_j) is not γ-regular is at most $\binom{h}{2}\gamma\leq\binom{q}{2}\gamma\leq\frac{1}{3}\delta$. Thus, it is enough to show that conditioning on the events: (i) the h vertices $v_1,\ldots,v_h$ belong to distinct sets V_i, (ii) if $v_i\in V_i$, $v_j\in V_j$ and (V_i,V_j) is γ-regular, then the probability that they span an induced copy of H is $IC(H,R)\pm\frac{1}{3}\delta$. Assuming events (i) and (ii) hold let us compute the probability that the h-tuple of vertices spans an induced copy of H, while conditioning on the h sets from $V_1,\ldots,V_\ell$ which contain the h vertices. For every possible set W of h sets V_i

we get from the choice of γ via Claim 2.18 that the probability that they span an induced copy of H is $IC(H,W) \pm \frac{1}{3}\delta$. This means that the conditional probability that the h-tuple of vertices span an induced copy of H is $IC(H,R) \pm \frac{1}{3}\delta$, as needed. $\square$

Proof (of Lemma 2.13): Suppose $\mathcal{P}$ is testable by a tester $\mathcal{T}$, and assume without loss of generality that $\mathcal{T}$ is canonical. This assumption is possible by Lemma 2.14. Let $q(\epsilon)$ be the upper bound guarantee for the query complexity of $\mathcal{T}$. Fix any n and δ and assume that $\delta < 1/12$ (otherwise, replace δ with $1/13$). Let $q = q(\delta, n) \leq q(\delta)$ be the query complexity, which is sufficient for $\mathcal{T}$ to distinguish between n-vertex graphs satisfying $\mathcal{P}$ and those that are δ-far from satisfying it, with success probability at least $2/3$. As $\mathcal{T}$ is canonical, if it samples a set of vertices and gets a graph of size q, it either rejects or accepts deterministically. Hence, we can define a set $\mathcal{A}$, of all the graphs Q of size q, such that if the sample of vertices spans a graph isomorphic to Q, then $\mathcal{T}$ accepts the input. We finally put $k = k_{2.20}(\delta/2^{\binom{q}{2}}, q)$, $\gamma = \gamma_{2.20}(\delta/2^{\binom{q}{2}}, q)$ and $T = T_{1.8}(k, \gamma)$. For any $k \leq t \leq T$ consider all the (finitely many) regularity-instances of order t, where for the edge densities $\eta_{i,j}$ we choose a real from the set $\{0, \frac{\delta\gamma^2}{50q^2}, 2\frac{\delta\gamma^2}{50q^2}, 3\frac{\delta\gamma^2}{50q^2}, \ldots, 1\}$. Let $\mathcal{I}$ be the union of all these regularity-instances. Note, that all the above constants, as well as the size of $\mathcal{I}$ and the complexity of the regularity-instances in $\mathcal{I}$, are determined as a function of δ only (and the property $\mathcal{P}$).

We claim that we can take $\mathcal{R}$ in Definition 2.6 to be

$$\mathcal{R} = \{R \in \mathcal{I} : \sum_{H \in \mathcal{A}} IC(R,H) \geq 1/2\} .$$

To see this, first note that the expression $\sum_{H\in\mathcal{A}} IC(R,H)$ is an estimation of the fraction of induced copies of graphs from $\mathcal{A}$ in a graph satisfying R. Combining the facts that the graphs in $\mathcal{A}$ all have size q and the use of Lemma 2.20 with $\delta/2^{\binom{q}{2}}$ we infer that the expression $\sum_{H\in\mathcal{A}} IC(R,H)$ is an estimate of the number of induced copies of graphs from $\mathcal{A}$ in a graph satisfying R, up to an additive error of at most $\delta\binom{n}{q}$.

Suppose a graph G satisfies $\mathcal{P}$. This means that $\mathcal{T}$ accepts G with probability at least $2/3$. In other words, this means that at least $\frac{2}{3}\binom{n}{q}$ of the subsets of q vertices of G span a graph isomorphic to one of the members of $\mathcal{A}$. By Lemma 1.8 G has some γ-regular partition of size at least k and at most T. As the densities in the regularity-instances in $\mathcal{A}$ differ by $\frac{\delta\gamma^2}{50q^2}$ we get that the densities of the regular partition of G differ by at most $\frac{\delta\gamma^2}{50q^2}$ from the densities of one of the regularity-instances $R \in \mathcal{I}$. Corollary 2.12 implies that G is δ/q^2-close to satisfying one of the regularity-instances of $\mathcal{I}$. Note that adding and/or removing an edge can decrease the number of induced copies of members of $\mathcal{A}$ in G by at most $\binom{n-2}{q-2}$. Thus adding and/or removing $\delta n^2/q^2$ edges can decrease the number of induced copies of members of $\mathcal{A}$ in G by at most $\delta\frac{n^2}{q^2}\binom{n-2}{q-2} \leq \delta\binom{n}{q}$. Thus, after these at most $\delta n^2/q^2$ edge modifications we get a graph that satisfies one of the regularity-instances $R \in \mathcal{I}$ where at least $(\frac{2}{3}-\delta)\binom{n}{q} > (\frac{1}{2}+\delta)\binom{n}{q}$ of the subsets of q vertices of the new graph span a member of $\mathcal{A}$ (here we use the assumption that $\delta < 1/12$). As explained in the previous paragraph, by our choice of k and γ via Lemma 2.20, this means that $\sum_{H\in\mathcal{A}} IC(R,H) \geq 1/2$. By

the definition of $\mathcal{R}$ this means that $R \in \mathcal{R}$, so G is indeed δ-close to satisfying one of the regularity-instances of $\mathcal{R}$.

Suppose now that a graph G is ϵ-far from satisfying $\mathcal{P}$. If $\delta \geq \epsilon$ then there is nothing to prove, so assume that $\delta < \epsilon$. If G is $(\epsilon - \delta)$-close to satisfying a regularity-instance $R \in \mathcal{R}$, then by the definition of $\mathcal{R}$ and our choice of k and γ via Lemma 2.20 it is $(\epsilon - \delta)$-close to a graph G', such that at least $(\frac{1}{2} - \delta)\binom{n}{q} > (\frac{1}{3} + \delta)\binom{n}{q}$ of the subsets of q vertices of G' span an induced copy of a graph from $\mathcal{A}$. In other words, this means that $\mathcal{T}$ accepts G' with probability at least $\frac{1}{3} + \delta$. This means that G' cannot be δ-far from satisfying $\mathcal{P}$ as we assume that q is enough for $\mathcal{T}$ to reject with probability at least $2/3$ graphs that are δ-far from satisfying $\mathcal{P}$. However, as G is ϵ-far from satisfying $\mathcal{P}$ any graph that is $(\epsilon - \delta)$-close to G must be δ-far from satisfying $\mathcal{P}$, a contradiction. □

2.4 Sampling Regular Partitions

The main result of this section (roughly) asserts that for every fixed γ, if we sample a constant number of vertices from a graph G, then with high probability the graph induced by the sample and the graph G will have the same set of γ-regular partitions. To formally state this result we introduce the following definition:

Definition 2.21 (δ-similar regular-partition)**.** *An equipartition $\mathcal{U} = \{U_i \mid 1 \leq i \leq k\}$ is* δ-similar *to a γ-regular equipartition $\mathcal{V} = \{V_i \mid 1 \leq i \leq k\}$, of the same order k (where $0 < \gamma \leq 1$), if:*

1. *$d(U_i, U_j) = d(V_i, V_j) \pm \delta$ for all $i < j$.*
2. *Whenever (V_i, V_j) is γ-regular, (U_i, U_j) is $(\gamma + \delta)$-regular.*

Observe that in the above definition, the two equipartitions $\mathcal{V}$ and $\mathcal{U}$ may be equipartitions of different graphs. In what follows, if $G = (V, E)$ is a graph and $Q \subseteq V(G)$, then $G[Q]$ denotes the subgraph induced by G on Q. Our main result in this section is the following:

Lemma 2.22. *For every k, δ there exists $q = q_{2.22}(k, \delta)$ such that a sample Q, of q vertices from a graph G, satisfies the following with probability at least $2/3$: If G has a γ-regular equipartition $\mathcal{V}$ of order at most k, then $G[Q]$ has an equipartition $\mathcal{U}$, which is δ-similar to $\mathcal{V}$. Also, If $G[Q]$ has a γ-regular equipartition $\mathcal{U}$ of order at most k, then G has an equipartition $\mathcal{V}$, which is δ-similar to $\mathcal{U}$.*

The proof of Lemma 2.22 has two main stages. For the first one we need a weaker result, which says that a sample of vertices has a regular partition, but with a *weaker* regularity measure.

Lemma 2.23 ([60])**.** *For every k and γ there exists $q = q_{2.23}(k, \gamma)$ such that if a graph G has a γ-regular equipartition $\mathcal{V} = \{V_1, \ldots, V_k\}$ of order k, then with probability at least $2/3$, a sample of q vertices will have an equipartition $\mathcal{U} = \{U_1, \ldots, U_k\}$ satisfying:*

1. *$d(U_i, U_j) = d(V_i, V_j) \pm \delta$ for all $i < j$.*

2. *Whenever* (V_i, V_j) *is* γ*-regular* (U_i, U_j) *is* $50\gamma^{1/5}$*-regular.*

For our purposes however, we cannot allow a weaker regularity as in the above lemma. Our main tool in the proof of Lemma 2.22 is Lemma 2.25 below, which establishes that if two graphs share *one* γ-regular equipartition, then they share *all* the γ'-regular-partitions where γ' is slightly larger than γ. This will allow us to strengthen Lemma 2.23 and thus obtain Lemma 2.22. For the statement of this lemma we need the following definition:

Definition 2.24 ((δ,γ)-similar regular-partitions)**.** *Two equipartitions* $\mathcal{V} = \{V_i \mid 1 \le i \le k\}$ *and* $\mathcal{U} = \{U_i \mid 1 \le i \le k\}$ *of the same order* k*, are said to be* (δ,γ)-similar *if:*

1. $d(U_i, U_j) = d(V_i, V_j) \pm \delta$ *for all* $i < j$*.*

2. *For all but at most* $\gamma\binom{k}{2}$ *of the pairs* $i < j$*, both* (V_i, V_j) *and* (U_i, U_j) *are* γ*-regular.*

Lemma 2.25. *For every* k *and* δ *there exists* $\zeta = \zeta_{2.25}(k,\delta)$ *with the following property: suppose that two graphs* $G = (V, E)$ *and* $\overline{G} = (\overline{V}, \overline{E})$ *have* (ζ,ζ)*-similar regular-equipartitions* $\mathcal{V} = \{V_1, \ldots, V_\ell\}$ *and* $\overline{\mathcal{V}} = \{\overline{V}_1, \ldots, \overline{V}_\ell\}$ *with* $\ell \ge 1/\zeta$*. Then, if* $\overline{G}$ *has a* γ*-regular equipartition* $\overline{\mathcal{A}} = \{\overline{A}_1, \ldots, \overline{A}_k\}$ *then* G *has an equipartition* $\mathcal{A} = \{A_1, \ldots, A_k\}$*, which is* δ*-similar to* $\overline{\mathcal{A}}$*.*

We turn to prove Lemma 2.25. We then use it to prove Lemma 2.22.

Proof (of Lemma 2.25): Let $\overline{A}_1, \ldots, \overline{A}_k$ be any equipartition of $\overline{G}$. Recall that ℓ denotes the order of the equipartition $\overline{\mathcal{V}}$, which is also the order of $\mathcal{V}$. For every $1 \le p \le \ell$ and $1 \le q \le k$ set $\overline{AV}_{p,q} = \overline{V}_p \cap \overline{A}_q$ and $\alpha_{p,q} = |\overline{AV}_{p,q}|/|\overline{V}_p|$. For every $1 \le p \le \ell$ and $1 \le q \le k$ let $AV_{p,q}$ be **any** subset of V_p of size $\alpha_{p,q}|V_p|$. Finally for every $1 \le q \le k$ define $A_q = \bigcup_{p=1}^{\ell} AV_{p,q}$. Instead of stating what $\zeta_{2.25}(k,\delta)$ should be, we state along the way different upper bound on $\zeta_{2.25}(k,\delta)$ that will depend only on k and δ. One can then take the minimum of all these values as $\zeta_{2.25}(k,\delta)$

Claim 1: If $(\overline{A}_q, \overline{A}_{q'})$ is γ-regular then $(A_q, A_{q'})$ is $(\gamma+\delta)$-regular.

Proof. To simplify the notation we assume that $(\overline{A}_1, \overline{A}_2)$ is γ-regular and prove that (A_1, A_2) is $(\gamma+2\delta)$-regular. Set $\eta = d(\overline{A}_1, \overline{A}_2)$. As Claim 2 below asserts $d(A_1, A_2) = \eta \pm \delta$. Thus we need to show that $d(A'_1, A'_2) = \eta \pm (\gamma+\delta)$ for every $A'_1 \subseteq A_1$ and $A'_2 \subseteq A_2$ of sizes $(\gamma+\delta)|A_1|$ and $(\gamma+\delta)|A_2|$, respectively. For simplicity we show that $d(A'_1, A'_2) \le \eta+\gamma+\delta$, as showing that $d(A'_1, A'_2) \ge \eta-\gamma-\delta$ is similar. Recall that each set A_q is the union of ℓ sets $AV_{1,q}, \ldots, AV_{\ell,q}$. For every $1 \le i, j \le \ell$ put $AV'_{i,1} = AV_{i,1} \cap A'_1$ and $AV'_{j,2} = AV_{j,2} \cap A'_2$. We can rephrase our goal in terms of the number of edges as follows

$$\sum_{1 \le i,j \le \ell} e(AV'_{i,1}, AV'_{j,2}) \le (\eta+\gamma+\delta)|A'_1||A'_2| = (\eta+\gamma+\delta)(\gamma+\delta)^2|A_1||A_2|. \tag{2.1}$$

Let n denote the number of vertices of G. To prove (2.1) we turn to bound the contribution to the LHS (= Left Hand Side) of (2.1) of three types of pairs of (i, j):

- **Pairs (i,j) for which $i=j$:** Observe that the maximum possible number of edges connecting all pairs $(AV_{i,1}, AV_{j,2})$ for which $i=j$ is given by $\sum_i \alpha_{i,1}\alpha_{i,2}|A_1||A_2|$. Furthermore, for any $1 \leq i \leq \ell$ we have $0 \leq \alpha_{i,1}, \alpha_{i,2} \leq k/\ell$ (this is because $|V_1| = \ldots = |V_\ell| = n/\ell$ and $|A_1| = \ldots = |A_k| = n/k$). By Claim 2.26 we get that $\sum_i \alpha_{i,1}\alpha_{i,2}|A_1||A_2| \leq \frac{k}{\ell}|A_1||A_2|$ and if we choose a ζ satisfying $\ell \geq 1/\zeta \geq 6k/\delta^3 \geq 6k/\delta(\gamma+\delta)^2$ we can infer that the contribution of the pairs (i,i) to the LHS of (2.1) is at most $\frac{1}{6}\delta(\gamma+\delta)^2|A_1||A_2|$ (note that $\ell \geq 1/\zeta$ is guaranteed by the statement of the lemma).

- **Pairs (i,j) for which either $|AV'_{i,1}| < \zeta|V_i|$ or $|AV'_{j,2}| < \zeta|V_j|$:** Consider the $1 \leq i \leq \ell$ in (2.1) for which $|AV'_{i,1}| < \zeta|V_i| = \zeta n/\ell$. The total number of vertices of G that belong to such sets is clearly at most ζn, therefore the total number of such vertices in A_1 is at most $k\zeta|A_1|$. Similarly, the total number of vertices of A_2 which belong to sets $|AV'_{j,2}|$ for which $|AV'_{j,2}| < \zeta|V_j|$ is at most $k\zeta|A_2|$. Therefore the contribution of pairs (i,j) to the LHS of (2.1) for which either $|AV'_{i,1}| < \zeta|V_i|$ or $|AV'_{j,2}| < \zeta|V_j|$ is at most $2k\zeta|A_1||A_2|$. If we choose ζ so that it satisfies $\zeta \leq \frac{\delta^3}{12k} \leq \frac{\delta(\gamma+\delta)^2}{12k}$, such pairs (i,j) can contribute to the LHS of (2.1) a total of at most $\frac{1}{6}\delta(\gamma+\delta)^2|A_1||A_2|$.

  For a later step of the proof it will be important to note that by the above reasoning, the number of vertices of A'_1 that belong to sets $AV'_{i,1}$ of size smaller than $\zeta|V_i|$ is at most $\delta|A_1|$. Similarly the number of vertices of A'_2 that belong to sets $AV'_{j,2}$ of size smaller than $\zeta|V_j|$ is at most $\delta|A_2|$.

- **Pairs (i,j) for which (V_i,V_j) is not ζ-regular:** Recall, that $\mathcal{V}$ is a ζ-regular equipartition therefore at most ζn^2 edges of G connect pairs of clusters (V_i,V_j) that are not ζ-regular. As $|A_1| = |A_2| = n/k$ this means that the number of edges connecting A_1 and A_2 that belong to pairs (V_i,V_j) that are not ζ-regular is at most $k^2\zeta(n/k)^2 = k^2\zeta|A_1||A_2|$. If we choose ζ so that $\zeta \leq \frac{1}{6}\delta^3/k^2 \leq \frac{1}{6}\delta(\gamma+\delta)^2/k^2$, such pairs can contribute at most $\frac{1}{6}\delta(\gamma+\delta)^2|A_1||A_2|$ to the sum in (2.1).

We have thus accounted for all pairs (i,j) in (2.1) for which either $i=j$, (V_i,V_j) is not ζ-regular, $|AV'_{i,1}| < \zeta|V_i|$ or $|AV'_{j,2}| < \zeta|V_j|$. Specifically, we have shown that they can contribute at most $\frac{1}{2}\delta(\gamma+\delta)^2|A_1||A_2| = \frac{1}{2}\delta|A'_1||A'_2|$ to the LHS of (2.1). Therefore, we can now reduce proving (2.1) to showing that

$$\sum_{i\in I, j\in J, i\neq j} e(AV'_{i,1}, AV'_{j,2}) = \sum_{i\in I, j\in J, i\neq j} d(AV'_{i,1}, AV'_{j,2})|AV'_{i,1}||AV'_{j,2}| \leq (\eta+\gamma+\frac{1}{2}\delta)|A'_1||A'_2|, \tag{2.2}$$

while assuming that all $i \in I$ and $j \in J$ in the above sum satisfy $|AV'_{i,1}| \geq \zeta|V_i|$ and $|AV'_{j,2}| \geq \zeta|V_j|$. Note, that the lemma assumes that if (V_i,V_j) is ζ-regular then so is $(\overline{V}_i, \overline{V}_j)$. Therefore we can assume that for any $i \in I$, $j \in J$, $i \neq j$

$$d(AV'_{i,1}, AV'_{j,2}) = d(V_i,V_j) \pm \zeta. \tag{2.3}$$

and

$$d(\overline{AV'}_{i,1}, \overline{AV'}_{j,2}) = d(\overline{V}_i, \overline{V}_j) \pm \zeta. \tag{2.4}$$

The reason is that if $i < j$ is such that (V_i, V_j) and $(\overline{V}_i, \overline{V}_j)$ are ζ-regular and furthermore $|AV'_{i,1}| \geq \zeta|V_i|$ and $|AV'_{j,2}| \geq \zeta|V_j|$ then the above follows from the definition of a ζ-regular pair. If one of these conditions does not hold then we will possibly recount some of the edges which we have already accounted for before. If we choose ζ so that $\zeta \leq \frac{1}{6}\delta$ we can use (2.3) to reduce (2.2) to showing

$$\sum_{i\in I, j\in J, i\neq j} d(V_i, V_j)|AV'_{i,1}||AV'_{j,2}| \leq (\eta + \gamma + \frac{2}{3}\delta)|A'_1||A'_2| \tag{2.5}$$

As we assume that $\mathcal{V}$ and $\overline{\mathcal{V}}$ are (ζ, ζ)-similar we have $d(V_i, V_j) = d(\overline{V}_i, \overline{V}_j) \pm \zeta$ for every $i < j$. If we choose ζ so that $\zeta \leq \frac{1}{6}\delta$, we can reduce (2.5) to showing that

$$\sum_{i\in I, j\in J, i\neq j} d(\overline{V}_i, \overline{V}_j)|AV'_{i,1}||AV'_{j,2}| \leq (\eta + \gamma + \frac{1}{3}\delta)|A'_1||A'_2| \tag{2.6}$$

By (2.4) we can reduce (2.6) to showing that

$$\sum_{i\in I, j\in J, i\neq j} d(\overline{AV'_{i,1}}, \overline{AV'_{j,2}})|AV'_{i,1}||AV'_{j,2}| \leq (\eta + \gamma)|A'_1||A'_2|. \tag{2.7}$$

Let $A''_1 = \bigcup_{i\in I} AV'_{i,1}$ and $A''_2 = \bigcup_{j\in J} AV'_{j,2}$. Clearly $|A''_1| \leq |A'_1|$ and $|A''_2| \leq |A'_2|$, thus we can prove (2.7) by deriving the following stronger assertion:

$$\sum_{i\in I, j\in J, i\neq j} d(\overline{AV'_{i,1}}, \overline{AV'_{j,2}})|AV'_{i,1}||AV'_{j,2}| \leq (\eta + \gamma)|A''_1||A''_2|. \tag{2.8}$$

Note, that as we have already mentioned, by our choice of ζ at most $\delta|A_1|$ vertices of A'_1 belong to sets $AV'_{i,1}$ for which $AV'_{i,1} < \zeta|V_1|$. Therefore, we have $|A''_1| \geq |A'_1| - \delta|A_1| \geq \gamma|A_1|$. Similarly, $|A''_2| \geq \gamma|A_2|$. Put $\beta_{i,1} = |AV'_{i,1}|/|A''_1|$ and $\beta_{j,2} = |AV'_{j,2}|/|A''_2|$. For every $i \in I$ let $\overline{AV}'_{i,1}$ be any subset of $\overline{AV}_{i,1}$ of size $\beta_{i,1}|\overline{AV}_{i,1}|$. Similarly, for every $j \in J$ let $\overline{AV}'_{j,2}$ be any subset of $\overline{AV}_{j,2}$ of size $\beta_{j,2}|\overline{AV}_{j,2}|$. Put $\overline{A}''_1 = \bigcup_{i\in I} \overline{AV}_{i,1}$ and $\overline{A}''_2 = \bigcup_{j\in J} \overline{AV}_{j,2}$ and note that just as $|A''_1| \geq \gamma|A_1|$ and $|A''_2| \geq \gamma|A_2|$ we also have $|\overline{A}''_1| \geq \gamma|\overline{A}_1|$ and $|\overline{A}''_2| \geq \gamma|\overline{A}_2|$. Dividing by $|A''_1||A''_2|$ we can restate (2.8) as

$$\sum_{i\in I, j\in J, i\neq j} d(\overline{AV'_{i,1}}, \overline{AV'_{j,2}})\beta_{i,1}\beta_{j,2} \leq \eta + \gamma.$$

Finally, note that the above holds because

$$\sum_{i\in I, j\in J, i\neq j} d(\overline{AV_{i,1}}, \overline{AV_{j,2}})\beta_{i,1}\beta_{j,2} \leq \sum_{1\leq i,j\leq \ell} d(\overline{AV_{i,1}}, \overline{AV_{j,2}})\beta_{i,1}\beta_{j,2} = d(\overline{A''}_1, \overline{A''}_2) \leq \eta + \gamma$$

due to the fact that $(\overline{A}_1, \overline{A}_2)$ is by assumption γ-regular, $d(\overline{A}_1, \overline{A}_2) = \eta$, $|\overline{A}''_1| \geq \gamma|\overline{A}_1|$ and $|\overline{A}''_2| \geq \gamma|\overline{A}_2|$. This completes the proof of the claim. □

Claim 2: For all $q < q'$ we have $d(A_q, A_{q'}) = d(\overline{A}_q, \overline{A}_{q'}) \pm \delta$

Proof. The proof is identical to the above proof. □

The proof of the lemma follows from the above two claims. □

Claim 2.26. *Let $a_1, \ldots, a_\ell$ and $b_1, \ldots, b_\ell$ satisfy $\sum_{1 \leq i \leq \ell} a_i = \sum_{1 \leq i \leq \ell} b_i = 1$ and $0 \leq a_i, b_i \leq k/\ell$, where $k \leq \ell$. Then $\sum_{1 \leq i \leq \ell} a_i b_i \leq k/\ell$.*

Proof. Observe that $\sum_{1 \leq i \leq \ell} a_i b_i \leq \max_{1 \leq i \leq \ell}\{a_i\} \sum_{1 \leq i \leq \ell} b_i \leq k/\ell$. □

Proof (of Lemma 2.22): Set $\zeta = (\zeta_{2.25}(k, \delta)/50)^5$ and $\zeta' = 50\zeta^{1/5}$ and note that $\zeta, \zeta' \leq \zeta_{2.25}(k, \delta)$. Let $\mathcal{V} = \{V_1, \ldots, V_\ell\}$ be a ζ-regular partition of G of order at least $1/\zeta$. Such an equipartition of order at most $T_{1.8}(1/\zeta, \zeta)$ exists by Lemma 1.8. By Lemma 2.23 we get that if we sample a set Q of at least $q_{2.23}(\ell, \zeta)$ vertices from G then with probability at least $2/3$ the graph induced on Q, which we denote by $G[Q]$ will have an equipartition $\mathcal{U} = \{U_1, \ldots, U_\ell\}$, such that $d(V_i, V_j) = d(U_i, U_j) \pm \zeta'$ and such that if (V_i, V_j) is ζ-regular then (U_i, U_j) is ζ'-regular. This means that with probability at least $2/3$, the graph $G[Q]$ is such that G and $G[Q]$ have equipartitions, which are $(\zeta_{2.25}(k, \delta), \zeta_{2.25}(k, \delta))$-similar. Indeed, as these equipartition we can take $\mathcal{V}$ and $\mathcal{U}$, because as $\zeta' \leq \zeta_{2.25}(k, \delta)$ then $d(V_i, V_j) = d(U_i, U_j) \pm \zeta_{2.25}(k, \delta)$. Also, as $\zeta \leq \zeta' \leq \zeta_{2.25}(k, \delta))$, then for all but at most $\zeta_{2.25}(k, \delta)\binom{k}{2}$ of the pairs $i < j$, both (V_i, V_j) and (U_i, U_j) are $\zeta_{2.25}(k, \delta)$-regular. Thus, Lemma 2.25 implies that for any γ-regular partition in G (respectively $G[Q]$) $G[Q]$ (respectively G) has an equipartition that is δ-similar to it. We can thus take $q_{2.22}(k, \delta) = q_{2.23}(\ell, \zeta)$ in the statement of the lemma because ℓ and ζ depend on k and δ. □

2.5 Testing Regular Partitions and Proof of the Main Result

In this section we apply the results of Sections 2.2 and 2.4 to prove Theorem 2.7. We start by proving the main technical result of this chapter by showing that the property of satisfying a regularity-instance is testable with a constant number of queries.

Proof (of Theorem 2.5): Suppose the regularity-instance R has error parameter γ, $\binom{k}{2}$ edge densities $\eta_{i,j}$, and a set of non-regular pairs $\overline{R}$. Given $G = (V, E)$ and ϵ, the algorithm for testing the property of satisfying R, samples a set of vertices Q, of size q, where q will be chosen later, and accepts G if and only if the graph spanned by Q is $\frac{\gamma^4 \epsilon}{200k^2}$-close to satisfying R. In what follows we denote by $G[Q]$ the graph spanned by Q.

Claim 1: If G satisfies R, and $q \geq q_1(\epsilon, k, \gamma)$, then $G[Q]$ is $\frac{\gamma^4 \epsilon}{200k^2}$-close to satisfying R with probability at least $2/3$.

Proof. If $G = (V, E)$ satisfies R, then V has an equipartition into $V_1, \ldots, V_k$ such that for all $(i, j) \notin \overline{R}$ the pair (V_i, V_j) is γ-regular. If we take $q_1(\epsilon, k, \gamma) = q_{2.22}(k, \frac{\gamma^6\epsilon}{10000k^2})$, then by Lemma 2.22, with probability at least 2/3 the graph $G[Q]$ will have an equipartition into k sets $A_1, \ldots, A_k$, such that $d(A_i, A_j) = \eta_{i,j} \pm \frac{\gamma^6\epsilon}{10000k^2}$ for all $i < j$, and if (V_i, V_j) is γ-regular then (A_i, A_j) is $(\gamma + \frac{\gamma^6\epsilon}{10000k^2})$-regular. By Corollary 2.12, this means that $G[Q]$ is $\frac{\gamma^4\epsilon}{200k^2}$-close to satisfying R. □

Claim 2: If G is ϵ-far from satisfying R, and $q \geq q_2(\epsilon, k, \gamma)$, then $G[Q]$ is $\frac{\gamma^4\epsilon}{200k^2}$-far from satisfying R with probability at least 2/3.

Proof. We take $q_2(\epsilon, k, \delta) = q_{2.22}(k, \frac{\gamma^4\epsilon}{200k^2})$. By Lemma 2.22 we get that with probability at least 2/3 the graph $G[Q]$ is such that if it has a γ'-regular equipartition of order k, then G has an equipartition which is $\frac{\gamma^4\epsilon}{200k^2}$-similar to it. We claim that if this event occurs then $G[Q]$ is $\frac{\gamma^4\epsilon}{200k^2}$-far from satisfying R, which is what we want to show. Suppose $G[Q]$ satisfies the above property and assume on the contrary that it is $\frac{\gamma^4\epsilon}{200k^2}$-close to satisfying R. Consider the $\frac{\gamma^4\epsilon}{200k^2}q^2$ edge modifications that make $G[Q]$ satisfy R and consider an equipartition $\mathcal{U} = \{U_1, \ldots, U_k\}$ of $G[Q]$, which satisfies R after performing these modifications. As we made at most $\frac{\gamma^4\epsilon}{200k^2}q^2$ edge modifications, we initially had $d(U_i, U_j) = \eta_{i,j} \pm \frac{\gamma^4\epsilon}{200}$. Consider now any $(i, j) \notin \overline{R}$. After these modifications (U_i, U_j) must be γ-regular with density $\eta_{i,j}$. Therefore, after these modifications every pair $U_i' \subseteq U_i, U_j' \subseteq U_j$ satisfying $|U_i'| \geq \gamma|U_i|$ and $|U_j'| \geq \gamma|U_j|$ satisfies $d(U_i', U_j') = \eta_{i,j} \pm \gamma$. Hence, before the modifications every such pair satisfied $d(U_i', U_j') = \eta_{i,j} \pm (\gamma + \frac{\gamma^2\epsilon}{200})$. Note that this means that every such pair was originally $(\gamma + \frac{\gamma^2\epsilon}{100})$-regular. By our assumption on $G[Q]$ this means that G has an equipartition in $V_1, \ldots, V_k$ such that $d(V_i, V_j) = \eta_{i,j} \pm \frac{\gamma^2\epsilon}{50}$ holds for all $i < j$, and for all $(i, j) \notin \overline{R}$ the pair (V_i, V_j) is $(\gamma + \frac{\gamma^2\epsilon}{50})$-regular. By Corollary 2.12, this means that G is ϵ-close to satisfying R, contradicting our assumption. □

Combining the above two claims we infer that if $q = \max\{q_1(\epsilon, k, \gamma), q_2(\epsilon, k, \gamma)\}$ then with probability at least 2/3 the algorithm distinguishes between the required two cases. Furthermore, the number of queries performed by the algorithm depends only on ϵ, k and γ, and is thus bounded from above by a function of ϵ and r. This completes the proof of the theorem. □

Having established the testability of any given regularity-instance we can prove Theorem 2.7. The last tool we need for the proof is the main result of [64] about estimating graph properties.

Theorem 2.27 ([64])**.** *Suppose that a graph property $\mathcal{P}$ is testable. Then for every $0 \leq \epsilon_1 < \epsilon_2 \leq 1$ there is a randomized algorithm for distinguishing between graphs that are ϵ_1-close to satisfying $\mathcal{P}$ and graphs that are ϵ_2-far from satisfying it. Furthermore, the query complexity of the algorithm can be bounded from above by a function of ϵ_1 and ϵ_2, which is independent of the size of the input.*

Proof (of Theorem 2.7): The first direction is given in Lemma 2.13. For the other direction, suppose that a graph property $\mathcal{P}$ is regular-reducible as per Definition 2.6. Let us fix n and ϵ. Put $r = r(\frac{1}{4}\epsilon)$ and let $\mathcal{R}$ be the corresponding set of regularity instances for $\delta = \frac{1}{4}\epsilon$ as in Definition 2.6. Recall that Definition 2.6 guarantees that the number and the complexity of the regularity-instances of $\mathcal{R}$ are bounded by a function of $\delta = \frac{1}{4}\epsilon$. By Theorem 2.5 for any regularity-instance $R \in \mathcal{R}$, the property of satisfying R is testable. Thus, by Theorem 2.27 for any such R, we can distinguish graphs that are $\frac{1}{4}\epsilon$-close to satisfying R from those that are $\frac{3}{4}\epsilon$-far from satisfying it, while making a number of queries, which is bounded by a function of ϵ. In particular, by repeating the algorithm of Theorem 2.27 an appropriate number of times (that depends only on $r = r(\frac{1}{4}\epsilon)$), and taking the majority vote, we get an algorithm for distinguishing between the above two cases, whose query complexity is a function of ϵ and r, which succeeds with probability at least $1 - \frac{1}{3r}$. As r itself is bounded by a function of ϵ, the number of queries of this algorithm can be bounded by a function of ϵ only.

We are now ready to describe our tester for $\mathcal{P}$: Given a graph G of size n and $\epsilon > 0$, the algorithm uses for every $R \in \mathcal{R}$ the version of Theorem 2.27 described in the previous paragraph, which succeeds with probability at least $1-\frac{1}{3r}$ in distinguishing between the case that G is $\frac{1}{4}\epsilon$-close to satisfying R and the case that it is $\frac{3}{4}\epsilon$-far from satisfying it. If it finds that G is $\frac{1}{4}\epsilon$-close to satisfying some $R \in \mathcal{R}$, then the algorithm accepts, and otherwise it rejects. Observe that as there are at most r regularity-instances in $\mathcal{R}$, we get by the union-bound that with probability at least $2/3$ the subroutine for estimating how far is G from satisfying some $R \in \mathcal{R}$ never errs. We now prove that the above algorithm is indeed a tester for $\mathcal{P}$. Suppose first that G satisfies $\mathcal{P}$. As we set $\delta = \frac{1}{4}\epsilon$ and $\mathcal{P}$ is regular-reducible to $\mathcal{R}$, the graph G must be $\frac{1}{4}\epsilon$-close to satisfying some regularity-instance $R' \in \mathcal{R}$. Suppose now that G is ϵ-far from satisfying $\mathcal{P}$. Again, as we assume that $\mathcal{P}$ is regular-reducible to $\mathcal{R}$, we conclude that G must be $\frac{3}{4}\epsilon$-far from satisfying all of the regularity-instances $R \in \mathcal{R}$. As with probability at least $2/3$ the algorithm correctly decides for any $R \in \mathcal{R}$ if G is $\frac{1}{4}\epsilon$-close to satisfying R or $\frac{3}{4}\epsilon$-far from satisfying it, we get that if G satisfies $\mathcal{P}$ then with probability at least $2/3$ the algorithm will find that G is $\frac{1}{4}\epsilon$-close to satisfying some $R \in \mathcal{R}$, while if G is ϵ-far from satisfying $\mathcal{P}$ then with probability at least $2/3$ the algorithm will find that G is $\frac{3}{4}\epsilon$-far from all $R \in \mathcal{R}$. By the definition of the algorithm, we get that with probability at least $2/3$ it distinguishes between graphs satisfying $\mathcal{P}$ from those that are ϵ-far from satisfying it. This means that the algorithm is indeed a tester for $\mathcal{P}$. □

2.6 Applications of the Main Result

In this section we show that Theorem 2.7 can be used in order to derive some positive and negative results on testing graph properties. We would like to stress that all these proofs implicitly apply the main intuition behind our characterization, which was explained after the statement of Theorem 2.7, that a graph property is testable if and only if knowing the regular partition of the graph is sufficient for inferring if a graph is far from satisfying the property. Our first application of Theorem 2.7 concerns testing for H-freeness; A graph is

said to be H-free if it contains no (not necessarily induced) copy of H. It was implicitly proved in [4] that for any H, the property of being H-free is testable. The main idea of the proof in [4] is that if G is ϵ-far from being H-free then a large enough sample of vertices will contain a copy of H with high probability. Here we derive this result from Theorem 2.7 by giving an alternative proof, which checks if the input satisfies some regularity-instance. For simplicity, we only consider testing triangle-freeness. We briefly mention that an argument similar to the one we use to test triangle-freeness can be used to test any monotone graph property. However, to carry out the proof one needs one additional non-trivial argument, which was proved in [14], so we refrain from including the proof.

Corollary 2.28. *Triangle-freeness is testable.*

Proof: By Theorem 2.7 it is enough to show that triangle-freeness is regular-reducible. Fix any $\delta > 0$ and set $\gamma' = \gamma_{2.18}(\delta, 3)$. Define $\gamma = \min\{\gamma', \delta\}$. We define $\mathcal{R}$ to be all the regularity-instances R satisfying the following: (i) They have regularity parameter γ (ii) They have order at least $1/\gamma$ and at most $T_{1.8}(1/\gamma, \gamma)$ (iii) Their densities $\eta_{i,j}$ are taken from $\{0, \gamma, 2\gamma, \ldots, 1\}$. (iv) They do not contain three clusters V_i, V_j, V_k such that $\eta_{i,j}, \eta_{j,k}, \eta_{i,k}$ are all positive.

To show that this is a valid reduction, assume first that G is ϵ-far from being triangle-free. Assume G is $(\epsilon - \delta)$-close to satisfying a regularity instance $R \in \mathcal{R}$. We can thus make $(\epsilon - \delta)n^2$ edge modifications and get a graph satisfying R. We also remove all edges inside the sets V_i. As by item (ii) each set has size at most $\gamma n \leq \delta n$ we remove less than δn^2 edges. The total number of edges removed is thus less than ϵn^2. By property (iv) of the regularity instances of $\mathcal{R}$ this means that the new graph is triangle-free, which is impossible because we made less than ϵn^2 edge modifications and G was assumed to ϵ-far from being triangle-free. Assume now that G is triangle-free. By Lemma 1.8 G has a γ-regular equipartition $V_1, \ldots, V_k$ of order $1/\gamma \leq k \leq T_{1.8}(1/\gamma, \gamma)$. Note that by our choice of γ' via Claim 2.18, and because $\gamma \leq \gamma'$, there are no i, j, k such that (V_i, V_j), (V_j, V_k), (V_i, V_k) are γ-regular and $d(V_i, V_j), d(V_j, V_k), d(V_i, V_k) \geq \delta$ because such sets span at least one triangle (in fact, many). As by item (iii) the densities of the instances in $\mathcal{R}$ are taken from $\{0, \gamma, 2\gamma, \ldots, 1\}$ we can make at most $\gamma n^2 \leq \delta n^2$ changes and "round down" the densities between the sets into a multiple of γ, while maintaining the regularity of the regular-pairs (we can use Lemma 2.8 here). This means that the new graph satisfies a regularity-instance $R \in \mathcal{R}$, which means that G was δ-close to satisfying R. $\square$

Our second application of Theorem 2.7 is concerned with testing k-colorability. This property was first implicitly proved to be testable in [105]. Much better upper bounds were obtained in [75], and further improved by [7]. As in the case of H-freeness, the main ideas of the proofs in [105, 75, 7] is that if G is ϵ-far from being k-colorable then a large enough sample of vertices will not be k-colorable with high probability. Here we derive this result by applying Theorem 2.7. Though we derive here only the testability of k-colorability, simple variants of the argument can be used to show that all the partition-problems studied in [75] are testable[2].

[2] An alert reader may note that our proof of Theorem 2.7 applies the result of [64], which relies on the

Corollary 2.29. *k-colorability is testable.*

Proof: By Theorem 2.7 it is enough to show that k-colorability is regular-reducible. Fix any $\delta > 0$ and define $\mathcal{R}$ to be all the regularity-instances R satisfying the following: (i) They have regularity measure δ (ii) They have order at least $1/\delta$ and at most $T_{1.8}(2/\delta, \delta)$ (iii) Their densities $\eta_{i,j}$ are taken from $\{0, \delta, 2\delta, \ldots, 1\}$. (iv) The following graph $T = T(R)$ is k-colorable: if R has order t then T has t vertices, and $(i,j) \in E(T)$ iff $\eta_{i,j} > 0$.

To show that this is a valid reduction, assume first that G is is ϵ-far from being k-colorable. Assume G is $(\epsilon - \delta)$-close to satisfying a regularity instance $R \in \mathcal{R}$. We can thus make $(\epsilon - \delta)n^2$ edge modifications and get a graph satisfying R. We also remove all edges inside the sets V_i. As by item (ii) each set has size at most δn we remove less than δn^2 edges. The total number of edges removed is thus less than ϵn^2. By property (iv) of the regularity instances of $\mathcal{R}$ this means that the new graph is k-colorable, which is impossible because we made less than ϵn^2 edge modifications and G was assumed to be ϵ-far from being k-colorable. Assume now that G is k-colorable and let $V_1, \ldots, V_k$ be the partition of $V(G)$, which is determined by a legal k-coloring of G. Break every set V_i into sets $U_{i,1}, \ldots, U_{i,2/\delta k}$ of size $\frac{1}{2}\delta n$. Put all the leftovers from each set in another set L of size $\frac{1}{2}\delta n$. By Lemma 1.8, starting from this equipartition we can get a δ-regular equipartition of G of order at most $T_{1.8}(2/\delta, \delta)$. Note that disregarding the refinement of L the new equipartition must satisfy item (v) in the definition of $\mathcal{R}$. As by item (iii) the densities of the instances in $\mathcal{R}$ are taken from $\{0, \delta, 2\delta, \ldots, 1\}$ we can make at most δn^2 edge modifications and thus "round down" the densities between the sets into a multiple of δ, while maintaining the regularity of the regular-pairs (we can use Lemma 2.8 here). This means that the new graph satisfies a regularity-instance $R \in \mathcal{R}$, which means that G was δ-close to satisfying R. □

The examples that were discussed above apply Theorem 2.7 to obtain positive results. Our third application of Theorem 2.7 derives a negative result. The main focus of [61] is testing for isomorphism to a given fixed graph. It shows that the query complexity of testing for isomorphism grows with a certain parameter, which measures the "complexity" of the graph. Without going into too much detail we just mention that under this measure random graphs are complex. Here we prove that testing for being isomorphic to a graph generated by $G(n, 0.5)$ requires a super-constant number of queries.

Corollary 2.30. *Let I be a graph generated by $G(n, 0.5)$. Then, with probability $1 - o(1)$ the property of being isomorphic to I is not testable.*

Proof: By Theorem 2.7 it is enough to show that with probability $1 - o(1)$ the property of being isomorphic to I is not regular-reducible. Note, that now there is only one value of n to consider in Definition 2.6 because the property we consider is a property of n-vertex graphs. Consider a graph generated by $G(n, 0.5)$. Clearly, by Lemma 2.11 the bipartite graph on any pair of sets of vertices of size $\sqrt{n}$ has density ≈ 0.5. We claim that if I satisfies this property then it is not regular-reducible. Suppose it is regular-reducible and consider

result of [75]. Thus, in the strict sense it is wrong to say that we infer the result of [75] from ours. However, it is not difficult to see that the result of [75] also follows from our (self-contained) proof of Lemma 2.22.

a small δ, say $\delta = 0.01$. Let $\mathcal{R}$ be the set of regularity-instances, which corresponds to this value of δ. Let G be a graph isomorphic to I. By Definition 2.6 it must be the case that G is δ-close to satisfying some $R \in \mathcal{R}$. By the properties of I this means that most densities of R must be close to 0.5. Let k denote the order of R and let $\eta_{i,j}$ denote its densities. Consider a random k-partite graph on sets of vertices $V_1, \ldots, V_k$ each of size n/k, where the bipartite graph connecting V_i and V_j is a random bipartite graph with edge density $\eta_{i,j}$. Clearly this graph is δ-close to satisfying R. On the other hand, it is not difficult to see that as most of the densities $\eta_{i,j}$ should be close to 0.5, then with high probability such a graph must be α-far from being isomorphic to I, for some fixed $\alpha > 0$, say $\alpha = 0.03$. This means that we have a graph that is 0.03-far from satisfying the property and is yet 0.01-close to satisfying one of the regularity-instances of $\mathcal{R}$. As we chose $\delta = 0.01$, this violates the second condition of Definition 2.6. □

2.7 Concluding Remarks and Open Problems

The main result of this chapter gives a combinatorial characterization of the graph properties, which can be tested with a constant number of edge queries in the dense graph model, possibly with a two-sided error. Together with the (near) characterization of [16] of the graph properties that can be tested with one-sided error, and the result of [64] showing that any testable property is also estimable, we get a more or less complete answer to many of the *qualitative* questions on testing graph properties in the dense model. While property testing in the dense model is relatively well understood, there are no general positive or negative results on testing graph properties in the bounded-degree model [76] or the general density model [101]. In these models the query complexity of the tester usually depends on the size of the input. It seems interesting and challenging to obtain general results in these models. One interesting problem is which of the partition problems which were studied in [75] can be tested using a sublinear number of queries. It will also be very interesting to give general positive and negative results concerning the testing of boolean functions.

Chapter 3

Uniform vs Non-uniform Property Testing

3.1 The Main Result

In this chapter we study the following question: are there graph properties that cannot be tested when the testing algorithm receives the error parameter ϵ as part of the input, and can be tested if ϵ is known in advance. Let us start by recalling the definition of a testable graph property that we have used thus far in the thesis.

Definition 3.1. (Testable) *A graph property $\mathcal{P}$ is* testable*, if there is a tester for $\mathcal{P}$ whose query complexity $q(\epsilon, n)$ can be bounded by a function $Q(\epsilon)$, which is independent of the size of the input.*

We stress that the definition of a tester for a testable property allows the query complexity to depend on n. It just requires that it will be possible to *bound* $q(\epsilon, n)$ by some function $Q(\epsilon)$. Therefore, for example $q(\epsilon, n) = 1/\epsilon + (-1)^n$ is a legitimate query complexity as is can be upper bounded by $Q(\epsilon) = 1/\epsilon + 1$. As we will see later, in some cases the distinction between query complexity depending only on ϵ and query complexity bounded by a function of ϵ may have interesting and non-trivial subtleties.

One of the fundamental problems of complexity theory is in understanding the relations between various models of computation. In particular, one would like to know if two models are equivalent or if there are problems, which can be solved in one model but not in the other. Regretfully, in many cases, though it seems obvious that two models of computation are not equivalent, the current techniques are far from enabling one to formally prove that. In this chapter we introduce two natural and realistic models of property-testing [1]. Surprisingly, in our case, though it seems at first that these models are equivalent, we manage to formally prove that they are in fact distinct. En route, we also formally prove that in some cases a

[1] These models are natural and realistic in the sense that they capture all the previous results on testing graph properties.

tester can make a non-trivial usage of both the error parameter ϵ and the size of the input graph n.

The main goal of testing properties in the dense graph model is to design a tester, whose query complexity can be upper bounded by a function, which is independent of the size of the input graph, and thus establish that a certain property is testable. In defining a tester above we have allowed the tester to use the size of the input in order to make its decisions. We now remind the reader of the definition of an oblivious tester, which was first stated in Chapter 1:

Definition 3.2. (Oblivious Tester) *A tester (one-sided or two-sided) is said to be* oblivious *if it works as follows: given ϵ the tester computes an integer $Q = Q(\epsilon)$ and asks an oracle for a subgraph induced by a set of vertices S of size Q, where the oracle chooses S randomly and uniformly from the vertices of the input graph. If Q is larger than the size of the input graph then the oracle returns the entire graph. The tester then accepts or rejects (possibly randomly) according to ϵ and the graph induced by S.*

At this point we remind the reader the discussion in Chapter 1 on why the definition of oblivious-tester is natural in the context of testing algorithm with constant number of queries. Informally, the notion of an oblivious tester means that the size of the input is not an important resource when studying property testing of "natural" graph properties in the dense graph model, such as hereditary properties, whose definition is independent of the input size. We stress that as opposed to the dense graph model, property-testing in the bounded degree model [76] and the general density model [101], usually requires query complexity, which depends on the size of the input graph. Therefore, the notion of oblivious testing is not adequate for those models.

The main resource that we seek to study in this chapter, is the value of the error parameter ϵ. In defining a tester before, we did not mention whether the error parameter ϵ is given as part of the input, or whether the tester is designed to distinguish between graphs that satisfy $\mathcal{P}$ from those that are ϵ-far from satisfying it, when ϵ is a known fixed constant. The current literature about property testing is not clear about this issue as in some papers ϵ is assumed to be a part of the input while in others it is not. We thus introduce the following two definitions:

Definition 3.3. (Uniformly testable) *A graph property $\mathcal{P}$ is* uniformly testable *if it can be tested by an oblivious tester as in Definition 3.2. Note that such a tester accepts ϵ as part of the input.*

Definition 3.4. (Non-uniformly testable) *A graph property $\mathcal{P}$ is* non-uniformly testable *if for every ϵ there is a tester T_ϵ for distinguishing between graphs satisfying $\mathcal{P}$ from those that are ϵ-far from satisfying it, which works as a standard tester making at most Q queries.*

Note, that in Definition 3.4 a tester T_ϵ does not receive ϵ as part of the input. Note also that we can think of a tester T_ϵ as a uniform tester, where the tester "knows" the quantity $Q(\epsilon)$ in advance and does not have to compute it. For this reason it is clear that if $\mathcal{P}$ is uniformly testable then it is also non-uniformly testable: For every ϵ we can define T_ϵ to

perform like the uniform tester for $\mathcal{P}$, while setting $Q = Q(\epsilon)$. As in the definition of a tester above, we generally allow a property to be uniformly (resp. non-uniformly) tested with two-sided error. If a property is uniformly (resp. non-uniformly) testable in a way that graphs satisfying the property are always accepted then the property is said to be uniformly (resp. non-uniformly) testable with one-sided error.

We believe that the distinction between uniform and non-uniform testing was not previously introduced in the literature because all the testable graph (and non graph) properties that were previously studied were in fact uniformly testable. As we have mentioned above, any property that is uniformly testable is also non-uniformly testable. It may thus seem, at least at first glance, that uniformly and non-uniformly property-testing are identical notions. However, the problem with trying to simulate a non-uniform tester(s) using a uniform one is that computing the query-complexity $Q(\epsilon)$, may be non-recursive. Our main result in this chapter is that when considering oblivious testers these two notions are in fact distinct. Moreover, these notions can be shown to be distinct while confining ourselves to graph properties, which are natural with respect to both their combinatorial structure and their computational difficulty.

Theorem 3.5. *There is a graph property $\mathcal{P}$ with the following properties:*

1. *$\mathcal{P}$ can be non-uniformly tested with one-sided error.*
2. *$\mathcal{P}$ cannot be uniformly tested, even with two-sided error.*

Moreover, satisfying $\mathcal{P}$ belongs to coNP and can be expressed in terms of forbidden subgraphs.

For a family of graphs $\mathcal{F}$ we define the property of being $\mathcal{F}$-free as the property of not containing a copy of any graph $F \in \mathcal{F}$ as a (not necessarily induced) subgraph. The property $\mathcal{P}$, which we construct in order to prove Theorem 3.5, is simply the property of being $\mathcal{F}$-free for some carefully defined family of graphs $\mathcal{F}$.

The reader should note that the difference between being uniformly testable and non-uniformly testable, is not as sharp as, say, the difference between P and $P/Poly$. The reason is that in $P/Poly$ the non-uniformity is with respect to the *inputs*, while in our case the non-uniformity is over the *error parameter*. In particular, a non-uniform tester T_ϵ should be able to handle *any* input graph. We note that it is possible to prove Theorem 3.5 by defining an undecidable graph property that can be non-uniformly tested with one-sided error, but obviously cannot be uniformly tested (because by setting $\epsilon = 1/n^2$ we precisely solve the undecidable problem). Theorem 3.5 however, gives a *natural* separation of these two models of property-testing by defining a decidable and combinatorially natural graph property, which satisfies the assertions of Theorem 3.5. We note that the main focus of property testing is in solving problems using the smallest possible amount of information about the input. Hence, undecidable properties are particulary unnatural in the context of property testing as such properties are not solvable/testable even if one has complete knowledge of the input.

3.1.1 Separations in Other Models of Property Testing

Is is natural to ask if it is possible to prove versions of Theorem 3.5 for other models of property testing. In particular, it is natural to ask if such a separation can be proved for the stronger model of property testing, where the tester can use the size of the graph in order to determine its query complexity (which should still be bounded by a function of ϵ) and make its decisions. As it turns out such a separation is not possible.

Proposition 3.6. (Rough Statement) *Suppose we allow a tester to use the size of the input graph in order to determine its query complexity. Then any property that can be tested with number of queries bounded by a function of ϵ, when ϵ is known in advance, can also be tested when ϵ is given as part of the input.*

Theorem 3.5 asserts that there are properties that can be tested by a tester if it knows ϵ in advance, which cannot be tested if ϵ is part of the input. In other words, it asserts that there are non-trivial computations the tester may perform with the error parameter ϵ. Therefore, Proposition 3.6 can be interpreted as asserting that knowing the size of the input can help a tester in a non-trivial way. More precisely, it shows that in some cases it is possible for the tester to compute the query complexity with the aid of the size of the input, while by Theorem 3.5 it is impossible to do so without this information. We note that Proposition 3.6 has some additional interesting implications. Its proof gives a non-trivial example, where the query complexity of a tester can be bounded by a function of ϵ only (as Definition 3.1 requires), while at the same time the query complexity depends on the size of the graph. It also gives a non-trivial example, where though the query complexity of a tester can be bounded by a function of ϵ only, the running time of the tester depends (exponentially!) on the size of the input. See the appendix of this chapter for the full details.

3.1.2 Monotone graph properties

In this subsection we briefly discuss the main result of [14], which was discussed in Section 1.3.1. Throughout the chapter we will make an extensive use of the notion of graph homomorphism, which we redefine for the convenience of the reader..

Definition 3.7. (Homomorphism) *A* homomorphism *from a graph F to a graph K, is a mapping $\varphi : V(F) \mapsto V(K)$ that maps edges to edges, namely $(v,u) \in E(F)$ implies $(\varphi(v), \varphi(u)) \in E(K)$.*

In the rest of the chapter, $F \mapsto K$ will denote that there is a homomorphism from F to K, and $F \not\mapsto K$ will denote that no such homomorphism exists. Just to practice the definition, note that if $F \mapsto K$ then $\chi(F) \leq \chi(K)$. In particular, this means that a graph G has a homomorphism into a clique of size k if and only if G is k-colorable. A key ingredient in the main result of [14] as well as in this chapter, is a certain graph theoretic functional, defined below.

Definition 3.8. (The function $\Psi_{\mathcal{F}}$) *For any (possibly infinite) family of graphs $\mathcal{F}$, and any integer k let $\mathcal{F}_k$ be the following set of graphs: A graph R belongs to $\mathcal{F}_k$ if it has at most k vertices and there is at least one $F \in \mathcal{F}$ such that $F \mapsto R$. For any such family $\mathcal{F}$ and integer k, for which $\mathcal{F}_k \neq \emptyset$, let*

$$\Psi_{\mathcal{F}}(k) = \max_{R \in \mathcal{F}_k} \min_{\{F \in \mathcal{F}: F \mapsto R\}} |V(F)| \,. \tag{3.1}$$

Furthermore, in case $\mathcal{F}_k = \emptyset$, define $\Psi_{\mathcal{F}}(k) = 0$.

Practicing definitions again, note that if $\mathcal{F}$ is the family of odd cycles, then $\mathcal{F}_k$ is precisely the family of non-bipartite graphs of size at most k. Also, in this case $\Psi_{\mathcal{F}}(k) = k$ when k is odd, and $\Psi_{\mathcal{F}}(k) = k - 1$ when k is even. The "right" way to think of $\Psi_{\mathcal{F}}$ is the following: Let R be a graph of size at most k and suppose we are guaranteed that there is a graph $F' \in \mathcal{F}$ such that $F' \mapsto R$ (thus $R \in \mathcal{F}_k$). Then by this information only and *without* having to know the structure of R itself, the definition of $\Psi_{\mathcal{F}}$ implies that there is a graph $F \in \mathcal{F}$ of size at most $\Psi_{\mathcal{F}}(k)$, such that $F \mapsto R$.

As it turns out, $\Psi_{\mathcal{F}}(k)$, which seems to have little, if any, to do with property testing, is in fact crucial to testing monotone graph properties. Call a function *recursive* if there is an algorithm for computing it in finite time (see [102]). The first effect of $\Psi_{\mathcal{F}}(k)$ on testing monotone graph properties is part of the main result of [14], which can be formulated as follows.

Theorem 3.9. ([14]) *For every (possibly infinite) family of graphs $\mathcal{F}$, the property of being $\mathcal{F}$-free is non-uniformly testable with one-sided error* [2]*. Moreover, if $\Psi_{\mathcal{F}}$ is recursive then being $\mathcal{F}$-free is also uniformly testable with one-sided error.*

Remark 3.10. *For the sake of completeness of this thesis, we remark that the fact that $\mathcal{F}$-freeness can be non-uniformly tested with one-sided error follows as a special case of Theorem 1.1 and the remark following its proof. As for the fact that in case $\Psi_{\mathcal{F}}$ is recursive, then $\mathcal{F}$-freeness is also uniformly testable with one-sided error, one can see that this follows from the proof of Theorem 7.4 and the discussion in Section 7.5, as in this case precisely the same functional $\Psi_{\mathcal{F}}$ is used.*

Remark 3.11. *The reader should note that Theorem 3.9 immediately applies also to any monotone property $\mathcal{P}$. The reason is that given $\mathcal{P}$ we can define $\mathcal{F} = \mathcal{F}_{\mathcal{P}}$ to be the set of graphs, which are minimal with respect to not satisfying the property $\mathcal{P}$. For example, if $\mathcal{P}$ is the property of being bipartite then $\mathcal{F}_{\mathcal{P}}$ is the (infinite) family of odd cycles. It is clear that satisfying $\mathcal{P}$ is equivalent to being $\mathcal{F}$-free. For convenience and ease of notation, in this chapter we describe monotone properties via their family of forbidden subgraphs.*

3.1.3 Main ideas and overview of the proof

The proof of Theorem 3.5 consists of two steps. In the first step we prove the somewhat surprising fact, that $\Psi_{\mathcal{F}}(k)$ being recursive is not only sufficient for inferring that being

[2] Again, we only consider decidable graph properties. Hence, in this case we assume that being $\mathcal{F}$-free is decidable.

$\mathcal{F}$-free is uniformly testable (this is guaranteed by Theorem 3.9), but this condition is also necessary. This is formulated in the following Theorem.

Theorem 3.12. *Suppose $\mathcal{F}$ is a family of graphs for which the function $\Psi_{\mathcal{F}}$ is not recursive. Then, the property of being $\mathcal{F}$-free cannot be uniformly tested with one-sided error.*

Note, that in Definition 3.3 the tester is defined as one that may have arbitrarily large query complexity, as long as it can be bounded by a function of ϵ only. Hence, in the case that $\Psi_{\mathcal{F}}$ is not recursive, Theorem 3.12 rules out the possibility of designing a uniform tester with arbitrarily large query complexity, as long as it can be bounded by a function of ϵ only.

The main idea behind the proof of Theorem 3.12 is that by "inspecting" the behavior of a property tester for the property of being $\mathcal{F}$-free one can compute the function $\Psi_{\mathcal{F}}$. The main combinatorial tool in the proof of Theorem 3.12 is a Theorem of Erdős [52] in extremal graph theory, which can be considered as a hypergraph version of the Zarankiewicz problem [91]. As an immediate corollary of Theorems 3.9 and 3.12 we obtain the following result, which *precisely* characterizes the families of graphs $\mathcal{F}$, for which the property of being $\mathcal{F}$-free can be tested uniformly (recall that by Theorem 3.9, for *any* family $\mathcal{F}$, being $\mathcal{F}$-free is non-uniformly testable). By Remark 3.11, this also gives a *precise* characterization of the monotone graph properties that are uniformly testable.

Corollary 3.13. *For every family of graphs $\mathcal{F}$, the property of being $\mathcal{F}$-free is uniformly testable with one-sided error if and only if the function $\Psi_{\mathcal{F}}$ is recursive.*

An immediate consequence of Theorems 3.9 and 3.12 is that in order to separate uniform testing with one-sided error from non-uniform testing with one-sided error, and thus (almost) prove Theorem 3.5, it is enough to construct a family of graphs $\mathcal{F}$ with the following two properties: (i) There is an algorithm for deciding whether a graph F belongs to $\mathcal{F}$ (recall that we confine ourselves to decidable graph properties). (ii) The function $\Psi_{\mathcal{F}}$ is non-recursive. The main combinatorial ingredient in the construction of $\mathcal{F}$ is the fundamental theorem of Erdős [51] in extremal graph theory, which guarantees the existence of graphs with arbitrarily large girth and chromatic number. As we want to prove Theorem 3.5 with a graph property, which is not only decidable, but even belongs to $coNP$, we need explicit constructions of such graphs. To this end, we use explicit constructions of expanders, which are given in [96]. For the construction we also apply some ideas from the theory of recursive functions. Finally, in order to obtain that being $\mathcal{F}$-free cannot be tested even with two-sided error, we use a result of Alon ([77] Appendix D) about testing hereditary graph properties.

Organization: The proof of Theorem 3.12 appears in Section 3.2, and the proof of Theorem 3.5 appears in Section 3.3. Section 3.4 contains concluding remarks and some open problems. The proof of Proposition 3.6 appears in the appendix of this chapter.

3.2 Computing $\Psi_{\mathcal{F}}$ via Testing $\mathcal{F}$-freeness

In this section we describe the proof of Theorem 3.12. Recall that $F \mapsto K$ denotes the fact that there is a homomorphism from F to K (see Definition 3.7). In what follows, an s-blowup of a graph K is the graph obtained from K by replacing every vertex $v_i \in V(K)$ with an independent set I_i, of size s, and replacing every edge $(v_i, v_j) \in E(K)$ with a complete bipartite graph whose partition classes are I_i and I_j. It is easy to see that a blowup of K is far from being K-free (K-free is the property of not containing a copy of K). It is also easy to see that if $F \mapsto K$, then a blowup of K is far from being F-free (see [1] Lemma 3.3). However, in this case the farness of the blowup from being F-free is a function of the size of F. As it turns out, for the proof of Theorem 3.12, we need a stronger assertion where the farness is only a function of $k = |V(K)|$. This stronger assertion is guaranteed by Lemma 3.15 below, whose proof relies on the following theorem of Erdős [52], which is a hypergraph extension of Zarankiewicz's problem [91].

Theorem 3.14. ([52]) *For every integer f there is an integer $N = N(f)$ with the following property: Every k-uniform hypergraph on $n > N$ vertices that contains at least $n^{k-f^{1-k}}$ edges, contains a copy of K_f^k, which is the complete k-partite k-uniform hypergraph, where each partition class is of size f.*

Lemma 3.15. *Let F be a graph on f vertices with at least one edge, let K be a graph on k vertices, and suppose $F \mapsto K$ (thus, $k \geq 2$). Then, for every sufficiently large $n \geq n(f)$, an n/k-blowup of K, is $\frac{1}{2k^2}$-far from being F-free.*

Proof: Denote by B the n-vertex n/k-blowup of K. Our goal is to show that after removing any set of $n^2/2k^2$ edges from B, the resulting graph still contains a copy of F. Name the vertices of K by $1, \ldots, k$ and the independent sets that replaced them by $I_1, \ldots, I_k$. Note, that for every choice of $v_1 \in V_1, \ldots, v_k \in V_k$, these k vertices span a copy of K with $v_i \in V_i$ playing the role of vertex $i \in V(K)$. We thus get that there are precisely $(n/k)^k$ such copies of K in B. (B may very well contain more copies of K but it is simpler to disregard them). Denote the set of these $(n/k)^k$ copies of K by $\mathcal{K}$, and note that each edge in B belongs to *precisely* $(n/k)^{k-2}$ copies of K that belong to $\mathcal{K}$. We conclude that removing any set of $n^2/2k^2$ edges, destroys at most $\frac{1}{2}(n/k)^k$ of the copies of K that belong to $\mathcal{K}$. Thus, after removing any set of $n^2/2k^2$ edges, the new graph, denoted B', contains at least $\frac{1}{2}(n/k)^k$ copies of K that belong to $\mathcal{K}$.

We now define a k-uniform k-partite hypergraph H, based on B'. We think of the k partition classes of H as the k vertex sets of B denoted above by $V_1, \ldots, V_k$. For every k vertices $v_1 \in V_1, \ldots, v_k \in V_k$ that span a copy of K that belongs to $\mathcal{K}$ in B', we put an edge in H containing $v_1, \ldots, v_k$. As B' contains at least $\frac{1}{2}(n/k)^k$ copies of K from $\mathcal{K}$, the hypergraph H contains at least $\frac{1}{2}(n/k)^k$ edges. By Theorem 3.14 for large enough n (i.e. large enough so that $n \geq N(f)$ and so that $\frac{1}{2}(n/k)^k \geq n^{k-f^{1-k}}$), the k-uniform hypergraph H contains a copy of K_f^k. For $1 \leq i \leq k$, let S_i denote the f vertices in V_i, which span this copy of K_f^k. By the definition of H, as well as the definition of the copies of K that belong to $\mathcal{K}$, we may conclude the following: In B', for every $1 \leq i < j \leq k$ for which $(i, j) \in E(K)$,

every vertex $v \in S_i$ is connected to every vertex $u \in S_j$. As $F \mapsto K$ it is now obvious that $S_1, \ldots, S_k$ span a copy of F in B', which is what we wanted to show. $\square$

For the proof of Theorem 4.1 we also need the following simple observation.

Claim 3.16. *Let $\mathcal{F}$ be a family of graphs and let T be a one-sided error uniform tester for the property of being $\mathcal{F}$-free, whose query complexity is $Q(\epsilon)$. If for some $\epsilon_0 > 0$, after T samples a set of vertices S of size $Q(\epsilon_0)$, the graph induced by S is $\mathcal{F}$-free, then T must accept the input.*

Proof: Suppose T does not accept the graph induced by T, and let G' denote the graph induced on the $Q(\epsilon)$ vertices. Suppose now that we were to execute T with the same error parameter ϵ_0 where the input graph is now G'. In that case the algorithm would just get back from the oracle the same graph G' and would reject the input G'. This however contradicts the assumption the T has one-sided error. $\square$

To simplify the proof of Theorem 3.12 we claim that we may assume that $\mathcal{F}$ contains no edgeless graph. Indeed, if $\mathcal{F}$ contains such a graph on t vertices, then by definition $\Psi_{\mathcal{F}}(k) \leq t$ for any k (this is because an edgeless graph has a homomorphism to a single vertex). Thus, it is easy to see (e.g. by applying the algorithm described in the proof below) that in this case $\Psi_{\mathcal{F}}(k)$ is recursive, and there is thus nothing to prove.

Proof of Theorem 3.12: We prove that if the property of being $\mathcal{F}$-free is uniformly testable with one-sided error and with arbitrary query complexity $Q(\epsilon)$, then $\Psi_{\mathcal{F}}(k)$ is recursive. Given a family of graphs $\mathcal{F}$ with no edgeless graph, consider the following algorithm for (nearly) computing $\Psi_{\mathcal{F}}(k)$, which simply implements its definition. The algorithm goes over all graphs R, of size at most k. For every such graph R, it searches for the smallest (in terms of number of vertices) $F \in \mathcal{F}$ for which $F \mapsto R$, if one such F exists [3]. The algorithm then takes the maximum over all graphs R for which it found at least one $F \in \mathcal{F}$ such that $F \mapsto R$. If for all graphs R of size at most k, there is no $F \in \mathcal{F}$ for which $F \mapsto R$, the algorithm returns 0.

The only problem with implementing the above algorithm is that given R, we have no way of knowing when to stop looking for a graph F for which $F \mapsto R$, if none exists. It is thus clear that in order to make sure that the algorithm always terminates with the correct value of $\Psi_{\mathcal{F}}(k)$, it is enough for the algorithm to be able to compute an *upper bound* of the size of such a graph F. In other words, it is enough to be able to compute an integer M such that if there is no $F \in \mathcal{F}$ of size at most M for which $F \mapsto R$, then no such $F \in \mathcal{F}$ exists.

We claim that for any $k \geq 2$ we can take $M = Q(1/2k^2)$ as such an upper bound, where $Q(\epsilon)$ is the upper bound for the query complexity of the uniform tester for the property of being $\mathcal{F}$-free [4]. Note, that M is thus computable, as we can simulate the alleged uniform

[3] As we assume that it is decidable to tell whether a graph belongs to $\mathcal{F}$, we can go over the graphs in $\mathcal{F}$ in order of increasing number of vertices, and for every graph try all possible homomorphisms from F to R

[4] $\Psi_{\mathcal{F}}(1) = 0$ because $\mathcal{F}$ does not contain independent sets.

tester with input $\epsilon = 1/2k^2$ and "see" what is the upper bound $Q = Q(\epsilon)$ that it is going to use [5]. We thus only have to show that it cannot be the case that for some R on k vertices, the smallest $F \in \mathcal{F}$ for which $F \mapsto R$ is larger than $Q(1/2k^2)$. Assume that one such R exists, and consider an n/k blowup of R, denoted by B. As by assumption $F \mapsto R$, we get by Lemma 3.15, that for every sufficiently large n, this blowup is $\frac{1}{2k^2}$-far from being F-free (here we also use the fact that F contains at least one edge). As $F \in \mathcal{F}$, the graph B is also $\frac{1}{2k^2}$-far from being $\mathcal{F}$-free. On the other hand, note that for any graph F' that is spanned by B (i.e. F' is a subgraph of B, which is not necessarily induced), there is a natural homomorphism φ, from F' to R, which maps all the vertices of F' that belong to the independent set that replaced vertex v, to v. As by assumption F is the smallest $F \in \mathcal{F}$ for which $F \mapsto R$, and F has more than $Q(1/2k^2)$ vertices, we conclude that there is no $F' \in \mathcal{F}$ on at most $Q(1/2k^2)$ vertices which is spanned by B. Now, by Claim 3.16, a one-sided error tester must find a member of $\mathcal{F}$ in order to declare that B is not $\mathcal{F}$-free. However, as the smallest member of $\mathcal{F}$ spanned by B has more than $Q(1/2k^2)$ vertices, this cannot be done with query complexity $Q(1/2k^2)$. □

Observe, that when computing the integer M in the above proof we do not know the size of the smallest graph F such that $F \mapsto R$. Hence, had we used a version of Lemma 3.15, where the farness from being F free is also a function of the size of F, we could not have computed the integer M, and thus could not have inferred that $\Psi_{\mathcal{F}}$ is recursive. Note also that the proof works no matter how large the query complexity $Q(\epsilon)$ is (this only affects the running time of the algorithm for computing $\Psi_{\mathcal{F}}(k)$), as long as it is a function of ϵ only.

3.3 Separating Uniform Testing from Non-Uniform Testing

In this section we prove Theorem 3.5 by constructing a family of graphs $\mathcal{F}$ for which it is possible to test the property of being $\mathcal{F}$-free non-uniformly, however it is impossible to test this property uniformly. The key combinatorial part of the construction of $\mathcal{F}$ is Lemma 3.18 below. For the proof of this lemma, we need an algorithm that can efficiently produce graphs with arbitrary large chromatic number and girth (the girth of a graph G denotes the size of the smallest cycle spanned by G). One of the best-known results of Erdős ([51], see also [20]), widely considered to be the most striking early application of the Probabilistic Method, asserts that such graphs exist. For our purposes however, we need explicit construction of such graphs. It is well known that the d-regular expanders, which can be efficiently constructed using the method of [96], have this property. This is formulated in the following theorem.

Theorem 3.17. ([96]) *For every pair of positive integers k and g, there is a graph F satisfying $\chi(F) > k$ and $g(F) > g$. Moreover, such a graph can be constructed in time polynomial in $|V(F)|$.*

[5] Recall, that a uniform tester operates by first computing an upper bound for its query complexity $Q = Q(\epsilon)$. Thus, we can "run" the tester on, say, an edgeless graph.

The reader can find some additional details about the above theorem in the appendix of this chapter. Applying the above theorem we prove the following.

Lemma 3.18. *There is an infinite family of graphs $F_1, F_2, \ldots$ with the following properties:*

1. *All the graphs $F_1, F_2, \ldots$ are connected and have no vertex of degree 1.*
2. *For any $1 \leq i < j$ we have $F_i \not\mapsto F_j$ and $F_j \not\mapsto F_i$.*
3. *There is an algorithm, which given i, prints F_i.*

Proof: We define the graphs $F_1, F_2, \ldots$ inductively as follows; F_1 is defined to be K_3 (i.e., a triangle). For every $i \geq 2$ we pick F_i to be the graph returned by Theorem 3.17, which satisfies $\chi(F_i) > \chi(F_{i-1})$ and $g(F_i) > |V(F_{i-1})|$. By Theorem 3.17 such an F_i exists. To get item (1) of the lemma we can now remove repeatedly from F_i any vertex of degree 1 because removing such a vertex does not change either the girth or the chromatic number of a graph. Also, we can assume without loss of generality that each graph F_i is connected, because if it is not, then we can take as F_i an appropriate connected component of F_i, which has girth and chromatic number at least as large as those of F_i. We thus get item (1) of the lemma. As we can use Theorem 3.17 in order to generate these graphs one after the other, we also get item (3).

We turn to prove item (2). First, note that if $\varphi : V(F) \mapsto V(K)$ is a homomorphism then any legal c-coloring of the vertices of K induces a legal c-coloring of the vertices of F; We simply color $v \in V(F)$ with the color of $\varphi(v)$. Therefore, if $\chi(F) > \chi(K)$ then we have $F \not\mapsto K$. Consider any pair F_i and F_j with $i < j$. As $\chi(F_j) > \chi(F_i)$ we immediately have that $F_j \not\mapsto F_i$. As $g(F_j) > |V(F_i)|$, every subgraph of F_j of size at most $|V(F_i)|$ does not span any cycle. In particular, any such subgraph is 2-colorable. Hence, as $\chi(F_i) > 2$, we also have $F_i \not\mapsto F_j$, completing the proof. □

In order to define the family of graphs $\mathcal{F}$, which we need in order to prove Theorem 3.5, we need the following definition.

Definition 3.19. (The language L_{BH}) *Fix any binary encoding of Turing-Machines. Define L_{BH} (short for* Bounded Halting*) to be the set of all pairs $i\#j$, such that the binary representation of i is a legal encoding of a Turing-Machine, which halts on an empty string within at most j steps.*

Clearly, L_{BH} is a decidable language; we first check if the binary representation of i is a legal encoding of a Turing-Machine. If it is not we reject. Otherwise, we simulate this machine for j steps on an empty string and check if during these j steps the machine halts.

In what follows P_j denotes a path of length j, and $F + P_j$ denotes the graph obtained by connecting P_j to an arbitrary vertex of F. We are now ready to define $\mathcal{F}$.

Definition 3.20. (The family $\mathcal{F}$) *Let $F_1, F_2, \ldots$ be the graphs from Lemma 3.18. Define*

$$\mathcal{F} = \bigcup_{i\#j \in L_{BH}} (F_i + P_j)$$

We now turn to prove that the family $\mathcal{F}$ has the required properties needed in order to satisfy the two assertions of Theorem 3.5. As in this chapter we confine ourselves to decidable properties, we first show that being $\mathcal{F}$-free is a decidable property. In fact, we also need this in order to apply Theorem 3.9. As shown in the next lemma, we can even show that being $\mathcal{F}$-free belongs to $coNP$.

Lemma 3.21. *Being $\mathcal{F}$-free, where $\mathcal{F}$ is the family of graphs from Definition 3.20, is in $coNP$.*

Proof: We prove the equivalent statement that the property of having a subgraph isomorphic to one of the graphs of $\mathcal{F}$ is in NP. Given a graph G of size n, the non-deterministic algorithm guesses a (not necessarily induced) subgraph of G, which we denote by T', a number $1 \leq t \leq n$ and an injective mapping $\sigma : [1, \ldots, t] \mapsto [1, \ldots, n]$. We next describe how the algorithm checks, using t and σ, whether $T' \in \mathcal{F}$.

The algorithm first verifies that T' has the structure of the graphs in $\mathcal{F}$. To this end, it first counts the number of vertices of degree 1 in T'. If this number is not precisely 1, or if T' is not connected the algorithm rejects the input (because by item (1) of Lemma 3.18 all the graphs in $\mathcal{F}$ are connected and have precisely one vertex of degree 1). Otherwise, let j be the length of the walk starting from the single vertex of degree 1, until the first vertex of degree at least 3 (including this last vertex), and let T be the graph obtained from T' by removing the j vertices of this path. The algorithm also rejects if T is not of size t. The algorithm now turns to check if T is isomorphic to one of the graphs F_i of Lemma 3.18, and if this is the case, whether $i\#j \in L_{BH}$.

The algorithm uses Theorem 3.17 in order to produce the graphs $F_1, F_2, \ldots$ as they were defined in Lemma 3.18. Note, that by our definition of these graphs each F_i must be strictly larger than F_{i-1}. If Theorem 3.17 produces a graph of size larger than t without first producing one of size t the algorithm rejects. Assume now that Theorem 3.17 produces a graph, say F_i, of size precisely t. The algorithm now checks whether for every edge $(i, j) \in E(F_i)$ the vertices $(\sigma(i), \sigma(j))$ also form an edge in G (recall that σ is an injective mapping from $[t]$ to n). If all these (at most $\binom{t}{2} \leq \binom{n}{2}$) tests succeed the algorithm moves to the last step, otherwise it rejects. Note, that at this step we know that T is isomorphic to some graph F_i from Lemma 3.18. To complete the verification that $T' \in \mathcal{F}$ the algorithm runs the algorithm (which is polynomial in i and j, which are bounded by n) for checking if $i\#j$ belongs to L_{BH} and accepts if and only if this algorithm accepts.

The above algorithm clearly rejects any G that is $\mathcal{F}$-free, and for any G that is not $\mathcal{F}$-free there is a choice of T', t, and σ, for which it accepts G. Finally, as for any i we have $|V(F_i)| > |V(F_{i-1})|$, we invoke Theorem 3.17 at most n times. As by Theorem 3.17 the time needed to produce each of the graphs F_i is polynomial in $|V(F_i)|$, we almost infer that the total running time of this algorithm is polynomial in n. The only annoying technicality is that it might be the case that we try to invoke Theorem 3.17 on inputs k and g for which the size of the graph it produces is super-polynomial in the size of the input graph G. To overcome this difficulty we can simply simulate the algorithm of Theorem 3.17 and reject if it runs longer than the time needed to produce a graph of size at most n, which is polynomial in n. $\square$

We are now ready to prove the main result of the section:

Lemma 3.22. *The function $\Psi_{\mathcal{F}}$, where $\mathcal{F}$ is the family of graphs from Definition 3.20, is non-recursive.*

Proof: We show that if $\Psi_{\mathcal{F}}$ is recursive, then given a legal encoding of a Turing-Machine M, we can compute an integer N with the following property: If M halts on the empty string, then it does so after at most N steps. We will thus get that we can decide whether M halts on the empty string, because we can simulate M on the empty string for N steps and accept if and only if M halts within these N steps. This will obviously be a contradiction, as deciding if a Turing-Machine halts on an empty string is well-known to be undecidable (see [102]).

Given an integer i, which (correctly) encodes some Turing-Machine M, the algorithm first computes the graph F_i. To this end we rely on item (3) of Lemma 3.18. Let k denote the number of vertices of F_i. We claim that we can set $N = \Psi_{\mathcal{F}}(k)$. First, observe that N is thus computable as $\Psi_{\mathcal{F}}$ is by assumption recursive. If M does not halt on the empty string, then we do not care about the value of N as no matter for how many steps we simulate M, it will never halt, and we will return the correct answer. Assume thus that M halts after T steps. We only have to show that $T \leq N$.

First, observe that for any graph F and integer j we trivially have $F \mapsto F + P_j$ and $F + P_j \mapsto F$. As by item (2) of Lemma 3.18, we know that for any $i < i'$ we have $F_i \not\mapsto F_{i'}$ and $F_{i'} \not\mapsto F_i$ we conclude that for any $i < i'$ and for any j, j' we also have $F_i + P_j \not\mapsto F_{i'} + P_{j'}$ and $F_{i'} + P_{j'} \not\mapsto F_i + P_j$. It thus follows that the only $F \in \mathcal{F}$ such that $F \mapsto F_i$, are the graphs of type $F_i + P_j$ for some integer j. However, as we only put in $\mathcal{F}$ the graphs $F_i + P_j$ for which $i\#j \in L_{BH}$ we infer that the only $F \in \mathcal{F}$ such that $F \mapsto F_i$, are the graphs of type $F_i + P_j$ for $j \geq T$. In particular, the smallest $F \in \mathcal{F}$ such that $F \mapsto F_i$ has size at least T. As $\Psi_{\mathcal{F}}(k)$ takes the maximum over all the graphs of size at most k, and F_i is one of these graphs, we get that $N = \Psi_{\mathcal{F}}(k) \geq T$. Hence, N is indeed an upper bound on the running time of M in the case that it halts on an empty string. □

The last tool we need is the following result of Alon ([77], Appendix D). In [77], the notion of uniformly testing a property was not used, but the statement as it appears below is equivalent to what is proved in [77].

Theorem 3.23. (c.f. [77]) *A hereditary graph property is uniformly testable with two-sided error if and only if it is uniformly testable with one-sided error.*

Proof of Theorem 3.5: We claim that for the property $\mathcal{P}$ of the theorem we can take the property of being $\mathcal{F}$-free with $\mathcal{F}$ being the family given in Definition 3.20. First, being $\mathcal{F}$-free is by definition expressed in terms of forbidden subgraphs and by Lemma 3.21 this property is in $coNP$. In particular, this property is decidable, therefore by Theorem 3.9 it can be tested non-uniformly with one-sided error. Now, by Lemma 3.22 the corresponding function $\Psi_{\mathcal{F}}$ is not recursive. Hence, by Theorem 3.12 this property cannot be tested uniformly with one-sided error. Finally, as this property is hereditary, Theorem 3.23 implies that this property cannot be tested uniformly, even with two-sided error. □

3.4 Concluding Remarks and Open Problems

The main result of the paper, Theorem 3.5, establishes that if we confine ourselves to slightly weakened testers, which are required to compute their complexity as a function of ϵ only, then there are non-trivial tasks (computing the query complexity), which cannot be done if ϵ is an unknown that is given as part of the input. Moreover, this phenomenon holds for properties that are natural in terms of their combinatorial structure (as they are monotone) and also in terms of their computational difficulty (as they are in $coNP$). This means that we can formally prove that in some cases knowing the error parameter ϵ in advance can help the tester in a non-trivial way. An interesting problem is whether one can find a separating property satisfying the assertions of Theorem 3.5, which belongs to P or perhaps even to a lower complexity class.

3.5 Appendix

3.5.1 Some remarks on LPS expanders:

The result of Lubotzky, Philips and Sarnak [96] can be stated as follows (see [20] for background on expander graphs)

Theorem 3.24. ([96]) *Suppose p and q are primes congruent to 1 modulo 4, where p is a quadratic residue modulo q, and put $d = p+1$ and $n = q(q^2-1)/2$. Then, there is a d-regular expander on n vertices, denoted $G_{n,d}$, with second eigenvalue $\lambda \leq 2\sqrt{d-1}$. Moreover,*

- *The chromatic number of $G_{n,d}$ is at least $\sqrt{d}/2$.*
- *The girth of $G_{n,d}$ is at least $\frac{2}{3}\log n/\log d$.*
- *$G_{n,d}$ can be constructed in time polynomial in $|V(G_{n,d})|$.*

Therefore, given integers k and g we can use the known results about the distribution of primes in arithmetic progressions, as well as the above theorem with n and d satisfying $\sqrt{d}/2 > k$ and $\frac{2}{3}\log n/\log d > g$, in order to efficiently construct the graphs satisfying the assertions of Theorem 3.17.

3.5.2 Proof of Proposition 3.6:

Clearly, if a graph property can be tested, when ϵ is given as part of the input, then for every fixed ϵ there is a tester for distinguishing between graphs satisfying $\mathcal{P}$ from those that are ϵ-far from satisfying it. To show the other direction, we need a theorem of [77] (extending a result of [6]) stating that for every ϵ and n if graph property is testable with query complexity $q(n,\epsilon)$, then it can also be tested by a so called "canonical tester", which operates by randomly selecting a set of $2q(n,\epsilon)$ vertices S, and then accepting or rejecting according to the graph spanned by S, the value of ϵ and the size of the input n.

Suppose then that for any $\epsilon > 0$ there is a tester T_ϵ that given the size of an input can distinguish between graphs satisfying $\mathcal{P}$ from those that are ϵ-far from satisfying it, such

that the query complexity of T_ϵ is at most $Q(\epsilon)$. Note, that we do not assume that the query complexity is a function of ϵ only, but just that it is upper bounded by a function of ϵ as in Definition 3.1. We turn to show that in this case there is a tester for $\mathcal{P}$ that receives ϵ as part of the input. The tester works as follows: Given n and ϵ the algorithm constructs the following families of n-vertex graphs: A, which consists of all the n-vertex graphs satisfying $\mathcal{P}$, and B, which consists of all the n-vertex graph, which are ϵ-far from satisfying $\mathcal{P}$. Starting from $q = 1$ the algorithm now goes over all the possible canonical-testers with query complexity q, and for each such tester, checks if it will accept the graphs of A with probability $2/3$, and reject the graphs of B with probability $2/3$. Recall that a canonical tester works by sampling a set of vertices and then accepting/rejecting according to the graph spanned by the sample. Therefore, when we say the the algorithm goes over all testers with query complexity q we mean that it tries all the possible $2^{2^{\binom{q}{2}}}$ ways of partitioning the set of q-vertex graphs into those that will make the canonical tester accept and those that will make it reject. Also, when we say that the algorithm checks, whether a given canonical tester accepts a graph from A with probability $2/3$, we mean that the algorithm checks if $2/3$ of the subsets of q vertices of the graph span a graph, which makes the canonical tester accept. Now, the main point is that as $\mathcal{P}$ is by assumption non-uniformly testable the algorithm will eventually find that for some $q \leq Q(\epsilon)$ there is a canonical tester T' for ϵ-testing $\mathcal{P}$ on n-vertex graphs. Once q and T' are found the algorithm executes T' on the input graph. By definition, this algorithm is a tester for $\mathcal{P}$, whose query-complexity is at most $Q(\epsilon)$. $\square$

The tester used in the above proof has two interesting features, which we have alluded to at the end of Section 3.1. First, although the query complexity of the uniform tester, which we construct in the above proof, is bounded by a function of ϵ only, it's running time is exponential in n, due to the need to go over all graphs of size n. Second, note that although the query complexity of the tester is bounded by a function of ϵ only, it is in fact a function of ϵ and n. The reason is that what the algorithm does is look for the smallest query complexity q, which is sufficient for testing $\mathcal{P}$ on n-vertex graphs. As $\mathcal{P}$ is assumed to be non-uniformly testable, we are guaranteed that for every n this quantity is bounded by some function of ϵ, which is independent of n. However, it may be the case that for fixed ϵ the optimal query complexity is different for different values of n. Therefore, for fixed ϵ the query complexity may be different for different values of n.

Chapter 4

Potpourri

4.1 The Main Results

In this chapter we prove several results that relate/use/complement the results of the previous three chapters of this part of the thesis. In the first section we prove that testing monotone properties with one-sided error may be arbitrarily difficult, that is, that for any function $Q(\epsilon)$ there are monotone properties that cannot be tested with $o(Q(\epsilon))$ queries. In the second section we prove a compactness type results in property testing which shows that if a graph is ϵ-far from satisfying an *infinite* family of hereditary properties $\mathcal{P}$, then it must be at least $\delta_{\mathcal{P}}(\epsilon)$-far from satisfying one them. In the third section we prove a result in extremal graph theory that shows that if a graph is ϵ-far from satisfying a graph property, then it contains a small induced subgraph that does not satisfy the property. In the fourth subsection we show how to extend the family of testable first order graph properties by applying the main result of Chapter 1. In the last section we show that a certain relaxation of the definition of ϵ-far cannot be used in order to allow one to test (essentially) any natural hereditary property.

4.2 A Lower Bound for Any Query Complexity

As is evident from the proof of Theorem 1.1, the upper bounds for testing a hereditary property depend on the property being tested. In other words, what we proved is that for every property $\mathcal{P}$, there is a function $Q_{\mathcal{P}}(\epsilon)$ such that $\mathcal{P}$ can be tested with query complexity $Q_{\mathcal{P}}(\epsilon)$. A natural question one may ask, is if the dependency on the specific property being tested can be removed. We rule out this possibility (even for monotone properties) by proving the following.

Theorem 4.1. *For any function $Q : (0,1) \mapsto \mathbb{N}$, there is a monotone graph property $\mathcal{P}$, which has no one-sided error property-tester with query-complexity bounded by $o(Q(\epsilon))$.*

Prior to this work, the best lower bound proved for testing a testable graph property with one-sided error was obtained in [1], where it is shown that for every non-bipartite

graph H, the query complexity of testing whether a graph does not contain a copy of H is at least $(1/\epsilon)^{\Omega(\log 1/\epsilon)}$. The fact that for every H this property is testable with one-sided error, follows from [4] and [6], and also as a special case from Theorem 1.1. As by Theorem 1.1 every monotone graph property is testable with one-sided error, Theorem 4.1 establishes that the one-sided error query complexity of testing testable graph properties, even those that are testable with one-sided error, may be *arbitrarily large*.

We turn to prove Theorem 4.1. We remind the reader that we denote by $F \mapsto K$ the fact that there is a homomorphism from F to K (see Definition 3.7). In what follows, an s-blowup of a graph K is the graph obtained from K by replacing every vertex $v_i \in V(K)$ with an independent set I_i, of size s, and replacing every edge $(v_i, v_j) \in E(K)$ with a complete bipartite graph whose partition classes are I_i and I_j. It is easy to see that a blowup of K is far from being K-free (K-free is the property of not containing a copy of K). It is also easy to see that if $F \mapsto K$, then a blowup of K is far from being F-free (see [1] Lemma 3.3). However, in this case the farness of the blowup from being F-free is a function of the size of F. As it turns out, for the proof of Theorem 4.1 we need a stronger assertion where the farness is only a function of $k = |V(K)|$. This stronger assertion was given in Lemma 3.15, which we quote for convenience.

Lemma 4.2. *Let F be a graph on f vertices with at least one edge, let K be a graph on k vertices, and suppose $F \mapsto K$ (thus, $k \geq 2$). Then, for every sufficiently large $n \geq n(f)$, an n/k-blowup of K, is $\frac{1}{2k^2}$-far from being F-free.*

For the proof of Theorem 4.1 we also need Claim 3.16, which we quote.

Claim 4.3. *Let $\mathcal{F}$ be a family of graphs, such that no $F \in \mathcal{F}$ has isolated vertices and let T be a one-sided error tester for the property of being $\mathcal{F}$-free with query complexity $Q(\epsilon, n)$. If for some $\epsilon_0 > 0$ and n, after T samples a set of vertices S of size $Q(\epsilon_0, n)$, the graph induced by S is $\mathcal{F}$-free, then T must accept the input.*

As our goal is to prove a lower bound on the query complexity we may and will assume that Q is monotone non-increasing (hence, monotone non-decreasing in $1/\epsilon$). For every such function Q we will define a property $\mathcal{P} = \mathcal{P}(Q)$ needed in order to prove Theorem 4.1. These properties can be thought of as *sparse bipartiteness* as they will be defined in terms of not containing a certain subset of the set of odd-cycles.

Let $Q : (0, 1) \mapsto \mathbb{N}$ be an arbitrary monotone non-increasing function. For such a function, let Q^i be the following i times iterated version of Q. We put $Q^1(x) = Q(x)$ and for any $i \geq 1$ define

$$Q^{i+1}(x) = 2Q\left(\frac{1}{2(Q^i(x)+2)^2}\right) + 1. \tag{4.1}$$

Define $I(Q) = \{Q^i(1/2) : i \in N\}$ and note that $I(Q)$ contains only odd integers. For a function as above, let $C(Q) = \{C_i : i \in I(Q)\}$, that is $C(Q)$ is the set of odd cycles whose lengths are the integers of the set $I(Q)$. Finally, let $\mathcal{P} = \mathcal{P}(Q)$ denote the property of not containing any of the odd-cycles of $C(Q)$ as a (not necessarily induced) subgraph.

Proof of Theorem 4.1: Given a monotone non-increasing function Q, let $\mathcal{P} = \mathcal{P}(Q)$ be the property defined above. We show that for any positive integer k for which $k-2 \in I(Q)$, any one-sided error tester that distinguishes between graphs of sufficiently large n that satisfy $\mathcal{P}$ from those that are $\frac{1}{2k^2}$-far from satisfying it, has query complexity at least $Q(1/2k^2)$. As Q is by assumption monotone non-increasing, $I(Q)$ contains infinitely many integers. Hence, for infinitely many values of ϵ, and for all large enough n, the query complexity of such a one-sided error tester is at least $Q(\epsilon)$. Note also that the set of these ϵ's approaches zero.

Fix any integer k for which $k-2 \in I(Q)$ and assume $k-2 = Q^i(1/2)$. As $I(Q)$ contains only odd integers, k is also odd. Define $\ell = Q^{i+1}(1/2)$ and recall that by (4.1), we have $\ell = 2Q(1/2k^2) + 1$. As it is clear that there is a homomorphism from C_ℓ to C_k, we get by Lemma 3.15 that for any $n \geq N(\ell)$, an n/k-blowup of C_k is $\frac{1}{2k^2}$-far from being C_ℓ-free. Denote such a blowup by G. As by definition $C_\ell \in C(Q)$, the graph G is also $\frac{1}{2k^2}$-far from satisfying $\mathcal{P}$. Also, as $k-2$ is odd, G contains no copy of C_{k-2}. In particular, G contains no member of $C(Q)$ of length less than ℓ. As property $\mathcal{P}$ is determined in terms of not containing a certain set of odd cycles, none of which has isolated vertices, we get from Claim 3.16 that a one-sided error tester must find a copy of a graph not satisfying $\mathcal{P}$, in order to determine that it does not satisfy $\mathcal{P}$. Therefore, for any $n \geq N(\ell)$ the query complexity of any tester for distinguishing between graphs of size n satisfying $\mathcal{P}$ from graphs of size n that are $\frac{1}{2k^2}$-far from satisfying it, is at least ℓ. As $\ell = 2Q(1/2k^2) + 1 \geq Q(1/2k^2)$ the proof is complete. □

An immediate consequence of Theorem 4.1 is that there is no function $Q(\epsilon)$ that upper bounds the query complexity $Q_{\mathcal{F}}(\epsilon)$, of testing the property of being $\mathcal{F}$-free for all families of graphs, $\mathcal{F}$. In other words, the dependence on the specific family of graph is unavoidable. This means that there is no function $Q(\epsilon)$ that upper-bounds the query complexity of testing all the hereditary graph properties with one-sided error. We conjecture that Theorem 4.1 can be extended to give lower bounds for two-sided error testers.

As we have commented at the beginning of this section, the proof of Theorem 4.1 heavily relies on the fact that the farness of the graph considered in Lemma 3.15 from being F-free is only a function of k. From the proof of Theorem 4.1 it should indeed be clear that if this farness had been a function of the size of F, then the length of each cycle of the family would have depended on its own size, which would result in a cycle of definitions.

4.3 A Compactness-type Result for Graph Properties

We next describe a consequence of Theorem 1.1, which does not assert the testability of some graph property, but rather one that may be useful in the general study of graph property testing. Suppose $\mathcal{P}_1, \ldots, \mathcal{P}_k$ are k graph properties that are closed under removal of edges. It is clear that if a graph G is ϵ-far from satisfying these k properties then it is at least ϵ/k-far from satisfying at least one of them. However, it is not clear that there is a fixed $\delta > 0$ such that even if $k \to \infty$, G must be δ-far from satisfying one of these properties. Our first result in this section is that such a δ does not necessarily exist.

Theorem 4.4. *For every n, there is a set of properties $\mathcal{P} = \{P_1, P_2, \ldots\}$ of n-vertex graphs, and an n-vertex graph G satisfying the following: G is $\frac{1}{10}$-far from satisfying all the properties of $\mathcal{P}$ and is yet $o(1)$-close to satisfying any single $P_i \in \mathcal{P}$.*

Proof: Consider the following set of properties: For any integer n, let $H_1, H_2, \ldots$ be some ordering of the graphs on n vertices, which contain precisely $n^{3/2}$ edges. A graph of size n is said to satisfy property $\mathcal{P}_i$ if it contains no copy of H_i. Clearly, any property $\mathcal{P}_i$ is closed under removal of edges, but not necessarily under removal of vertices. Observe, that any graph with at least $n^{3/2}$ edges does not satisfy one of the properties $\mathcal{P}_i$. Therefore, any graph G of size n, which contains $\frac{1}{5}n^2$ edges is at least $\frac{1}{10}$-far from satisfying all the properties $\mathcal{P}_i$. We claim that any such G is $\frac{\log n}{\sqrt{n}}$-close to satisfying any one of these properties. To this end, it is enough to show that for any graph H_i, we can remove at most $n^{3/2}\log n$ edges from G and thus make it H_i-free. To see this, note that as G and H_i are both of size n, G spans at most $n!$ copies of H_i. As H_i contains $n^{3/2}$ edges a randomly chosen edge of G is spanned by H_i with probability at least $n^{3/2}/\binom{n}{2} > 1/\sqrt{n}$. Thus, if we remove from G a set of $n^{3/2}\log n$ edges, were each edge is randomly and uniformly chosen from the edges of G (with repetitions), the probability that none of the edges of one of the copies of H_i in G were removed is at most $(1-1/\sqrt{n})^{n^{3/2}\log n} < 1/n!$. By the union bound, the probability that for *some* copy of H_i in G, none of its edges were removed is strictly smaller than 1. Thus, there exists a choice of $n^{3/2}\log n$ edges, whose removal from G makes it H_i-free. □

Suppose now that $\mathcal{P}$ is a set of hereditary properties. In this case it is not clear that even if $\mathcal{P}$ contains just two properties then if G is ϵ-far from satisfying the two of them, then it must be at least $\delta(\epsilon)$-far from satisfying one of them. Somewhat surprisingly, we can show that this is indeed the case even if $\mathcal{P}$ contains infinitely many properties. This can be viewed as a compactness-type result for graph properties.

Theorem 4.5. *For any (possibly infinite) set of hereditary graph properties $\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, \ldots\}$, there is a function $\delta_{\mathcal{P}} : (0,1) \mapsto (0,1)$ with the following property: If a graph G is ϵ-far from satisfying all the properties of $\mathcal{P}$, then for some i, the graph G is $\delta_{\mathcal{P}}(\epsilon)$-far from satisfying $\mathcal{P}_i$.*

Proof: For each of the hereditary properties $\mathcal{P}_i$, let $\mathcal{F}_i$ be the family of forbidden induced subgraphs of $\mathcal{P}_i$ as in Definition 1.11, and let $\mathcal{F} = \mathcal{F}_1 \bigcup \mathcal{F}_2 \bigcup \mathcal{F}_3 \bigcup \ldots$. Clearly, a graph G satisfies all the properties of $\mathcal{P}$ if and only if it is induced $\mathcal{F}$-free. Consider a graph G, which is ϵ-far from satisfying all the properties of $\mathcal{P}$. In this case G is also ϵ-far from being induced $\mathcal{F}$-free, hence, by Lemma 1.12, there is a graph $F \in \mathcal{F}$ of size $f = f_{\mathcal{F}}(\epsilon)$ such that G contains $\delta_{\mathcal{F}}(\epsilon)n^f$ induced copies of F. Note, that adding or removing an edge from G destroys at most $\binom{n}{f-2} \leq n^{f-2}$ induced copies of F. Thus, one must add or delete at least at least $\delta_{\mathcal{F}}(\epsilon)n^2$ edges to G in order to turn it into a graph containing to induced copy of F. Let i be such that $F \in \mathcal{F}_i$. We may now infer that G is $\delta_{\mathcal{F}}(\epsilon)$-far from satisfying $\mathcal{P}_i$. Finally, note that as $\mathcal{F}$ is determined by $\mathcal{P}$, we can also say that G is $\delta_{\mathcal{P}}(\epsilon)$-far from satisfying $\mathcal{P}_i$. □

4.4 An Extremal Result for All Graph Properties

Confirming a conjecture of Erdős, it was shown in [105] that if a graph is ϵ-far from being k-colorable, then it contains a non k-colorable subgraph of size $c(\epsilon)$. As we have alluded to in Subsection 1.1.1, the main technical result of Chapter 1, Lemma 1.12, immediately implies that this result can be extended to the entire family of hereditary graph properties. In fact, we can show that a similar result holds for *any* graph property.

Theorem 4.6. *For every graph property $\mathcal{P}$, there is a function $W_{\mathcal{P}}(\epsilon)$ with the following property: If G is ϵ-far from satisfying $\mathcal{P}$, then G contains an* **induced** *subgraph of size at most $W_{\mathcal{P}}(\epsilon)$, which does not satisfy $\mathcal{P}$.*

Proof of Theorem 4.6: Given any graph property $\mathcal{P}$ let $\mathcal{F}$ be the family of graphs not satisfying $\mathcal{P}$. Observe, that if a graph is ϵ-far from satisfying $\mathcal{P}$ then it is also ϵ-far from being induced $\mathcal{F}$-free and thus by Lemma 1.12 it contains an induced subgraph $F \in \mathcal{F}$ of size at most $f_{\mathcal{F}}(\epsilon)$, and by our choice of $\mathcal{F}$ the graph F does not satisfy $\mathcal{P}$. Therefore, as the function $W_{\mathcal{P}}(\epsilon)$ in the statement of Theorem 4.6 we can take the function $f_{\mathcal{F}}(\epsilon)$. □

We note that the above theorem implies that if a graph is ϵ-far from satisfying a hereditary property $\mathcal{P}$, then it contains a small proof that the graph indeed does not satisfy $\mathcal{P}$. This is because any graph that contains an induced subgraph that does not satisfy $\mathcal{P}$ cannot itself satisfy $\mathcal{P}$. Observe that this is not the case for non hereditary properties. For example, if the property is "having a clique on half the vertices of the graph" then the above theorem implies that if G is ϵ-far from satisfying this property then it contains a subgraph of size at most $c(\epsilon)$ that has no clique on half the vertices. Of course, such a subgraph does not guarantee that the entire graph has no such clique.

4.5 Testing Unbounded First-Order Graph Properties

A first order graph property is one involving the boolean operators $\wedge, \vee, \neg$, the $\forall, \exists$ quantifiers, the equality operator $=$, and the adjacency relation $\sim$. For example, the triangle-freeness property can be written as $\forall\ v_1, v_2, v_3 \neg(v_1 \sim v_2 \wedge v_2 \sim v_3 \wedge v_1 \sim v_3)$. The main result of [6] states that every first order graph property without quantification $\forall\exists$ is testable (possibly with two-sided error). The main tool in [6] was a theorem stating that any hereditary graph property, which is expressible in terms of a *finite* family of forbidden induced subgraphs is testable. Theorem 1.1 is a powerful extension of this result as it allows the family of forbidden induced subgraphs to be infinite. One may thus ask whether Theorem 1.1 can be used in order to extend the result of [6]. Theorem 4.8 below gives a positive answer to this question. To state this extension we need the following definition.

Definition 4.7. (Unbounded First-Order Properties of type $\exists\forall$) *An unbounded first order graph property of type $\exists\forall$ is of the form*

$$\exists x_1, \ldots, x_t \bigwedge_{i=1}^{\infty} \forall y_1, \ldots, y_i \ A_i(x_1, \ldots, x_t, y_1, \ldots, y_i) \tag{4.2}$$

where each $A_i(x_1, \ldots, x_t, y_1, \ldots, y_i)$ is a quantifier-free first order expression.

The main result of [6] states that any graph property that can be expressed as above while using a *single* relation A_i is testable. Using the main techniques of this chapter, we can extend this to expressions containing *infinitely* many expressions A_i.

Theorem 4.8. *Every graph property describable by an unbounded first order graph property of type $\exists\forall$ is testable (possibly with two-sided error).*

It should be noted that it is proved in [6] that there are first order graph properties with alternation of type $\forall\exists$ which are not testable, thus Theorem 4.8 is in some sense best possible.

We turn to sketch the proof of Theorem 4.8. As most of the technical details are very similar to those appearing in [6] we only discuss the main idea needed to obtain the extension of the result of [6]. We start with a useful result of [6].

Definition 4.9. (Indistinguishability) *Two graph properties $\mathcal{P}$ and $\mathcal{Q}$ are called* indistinguishable *if for every $\epsilon > 0$ there exists $N = N(\epsilon)$ satisfying the following; A graph on $n \geq N$ vertices satisfying one of the properties is never ϵ-far from satisfying the other.*

Lemma 4.10. ([6]) *If $\mathcal{P}$ and $\mathcal{Q}$ are indistinguishable graph properties, then $\mathcal{P}$ is testable if and only if $\mathcal{Q}$ is testable.*

We next define an extension of the notion of colorability. A similar notion was used in [6], where $\mathcal{F}$ was restricted to be *finite*.

Definition 4.11. ($\mathcal{F}$-colorability) *Suppose we are given c, and a (possibly infinite) family (with repetitions) $\mathcal{F}$ of graphs, each of which is provided with a c-coloring (i.e. a function from its vertex set to $\{1, \ldots, c\}$ which is not necessarily a proper c-coloring in the usual sense). A c-coloring of a graph G is called an* $\mathcal{F}$-coloring *if no member of $\mathcal{F}$ appears as an induced subgraph of G with an identical coloring. A graph G is called* $\mathcal{F}$-colorable *if there exists an $\mathcal{F}$-coloring of it.*

Note, that for any family of colored graphs $\mathcal{F}$ (finite or infinite), being $\mathcal{F}$-colorable is a hereditary graph property. We thus get the following from Theorem 1.1:

Lemma 4.12. *For any family of colored graphs $\mathcal{F}$, being $\mathcal{F}$-colorable is testable.*

Note, that by Theorem 1.1 being $\mathcal{F}$-colorable is in fact testable with one-sided error, but we do not need this stronger assertion here. The following lemma shows the relevance of the notion of $\mathcal{F}$-colorability for the proof of Theorem 4.8.

Lemma 4.13. *For every first order property $\mathcal{P}$ of the form*

$$\exists x_1, \ldots, x_t \bigwedge_{i=1}^{\infty} \forall y_1, \ldots, y_i \ A_i(x_1, \ldots, x_t, y_1, \ldots, y_s)$$

there exists a (possibly infinite) family $\mathcal{F}$, of $(2^{t+\binom{t}{2}}+1)$-colored graphs such that the property $\mathcal{P}$ is indistinguishable from the property of being $\mathcal{F}$-colorable.

Proof: (sketch) The proof uses ideas very similar to those used to prove Lemma 2.2 in [6] and is thus omitted. We briefly mention that one can use the same technique of [6] along with the fact that one is allowed to put in $\mathcal{F}$ *infinitely* many forbidden colored subgraphs. □

Proof of Theorem 4.8: Immediate from Lemmas 4.10, 4.12 and 4.13. □

4.6 On the (Im)possibility of Relaxing the Definition of ϵ-Far

Theorems 1.1 and 1.4 imply that any hereditary graph property is testable, when one uses the standard notion of ϵ-far. Suppose we forbid addition of edges and define a graph G on n vertices to be ϵ-far$_{del}$ from satisfying property $\mathcal{P}$ if one needs to delete from G at least ϵn^2 edges in order to turn it into a graph satisfying $\mathcal{P}$. We say that property $\mathcal{P}$ is testable$_{del}$ if there is a tester for distinguishing between graphs satisfying $\mathcal{P}$ from those that are ϵ-far$_{del}$ from satisfying it, whose number of queries can be upper bounded by a function of ϵ. A natural question is which graph properties are testable$_{del}$. Obviously, any hereditary property, which is also closed under removal of edges (such as k-colorability) is testable$_{del}$ as in these cases being ϵ-far$_{del}$ is equivalent to ϵ-far. The following theorem is a sharp contrast to Theorems 1.1 and 1.4.

Theorem 4.14. *For any hereditary property $\mathcal{P}$, which is not closed under removal of edges, and is satisfied by any complete graph, there is a constant $\delta = \delta(\mathcal{P}) > 0$ such that testing$_{del}$ property $\mathcal{P}$ (even with two-sided error) requires n^{δ} queries.*

Note, that any natural hereditary property, such as any of those discussed in Subsection 1.1.1, is satisfied by any complete graph, thus the above result applies to these properties. We briefly mention that we can also prove a similar statement when one allows only edge additions.

We turn to prove Theorem 4.14. Before getting to the details we first make some simple observations. Note, that if the property $\mathcal{P}$ is satisfied by all graphs then it is clearly testable. This means that if $\mathcal{P}$ is not satisfied by all graphs and is satisfied by all the cliques then it cannot be closed under removal of edges. Thus, this condition in the statement can actually be removed. Also, note that when considering the notion of ϵ-far$_{del}$ there is no sense in considering hereditary properties, which are not satisfied by some independent set, as in this case any graph with even a single independent set (say, of size 3) is arbitrarily far from satisfying the property and but finding this independent set requires $\Omega(n^2)$ queries.

Our main tool for the proof of Theorem 4.14 is the following result, which is essentially proved by Frankl and Füredi in [66].

Theorem 4.15. ([66]) *For any graph $F = (R, T)$, with $|T| = t > 0$ edges there is a constant $\delta = \delta(F)$ with the following property: For any integer n there is a graph $G_n = (V, E)$ on n vertices, which consists of $(1 - n^{-\delta})\binom{n}{2}/t$ induced copies of of F, such that no two copies of F share an edge.*

Proof of Theorem 4.14: By the discussion above we may assume that $\mathcal{P}$ has at least one forbidden induced subgraph $F = (R, T)$ and that F is not an independent set. Put $t = |T|$ and for any n let G_n be the graph, whose existence is guaranteed by Theorem 4.15. As all these graphs consist of $(1 - n^{-\delta})\binom{n}{2}/t > n^2/4t$ induced copies of F, where non of the copies share an edge, these graphs are all at least $\frac{1}{4t}$-far$_{del}$ from being induced F free. Hence, they are also at least $\frac{1}{4t}$-far$_{del}$ from satisfying $\mathcal{P}$. On the other, as we assume that any clique satisfies $\mathcal{P}$, and G contains $(1 - n^{-\delta})\binom{n}{2}$ edges, any randomized algorithm with query-complexity much smaller than n^{δ} cannot test$_{del}$ property $\mathcal{P}$ as it has a negligible probability of distinguishing between G_n, which are $\frac{1}{4t}$-far$_{del}$ from satisfying $\mathcal{P}$, and a clique of size n, which by assumption satisfies $\mathcal{P}$. □

Suppose we define ϵ-far$_{add}$ and testable$_{add}$ but now allowing only edge additions. One can easily see that simple modifications of the proof of Theorem 4.14 imply that the same lower bound can be proved for testing$_{add}$ any hereditary property, which is not closed under edge additions and which is satisfied by any edgeless graph.

Part II

On the Possibility of Small Query Complexity

Chapter 5

Testing Induced Subgraph-Freeness

5.1 The Main Results

As we have discussed in the introduction of this thesis, in the first part we tried to give general testability results. The drawback of these general results is that they supply very weak upper bound for the number of queries one has to perform in order to test even simple properties. In this chapter, as well as in the following one, we try to classify the properties that can be efficiently testable. The main focus of this chapter is to obtain a characterization of the graphs H for which the property of being induced H-free can be tested with a "small" number of queries. More precisely, throughout this chapter, as well as the next one, we call a property $\mathcal{P}$ *easily testable* if there is a one-sided error property tester for $\mathcal{P}$ whose query complexity is polynomial in $1/\epsilon$. If no such property tester exists we say that $\mathcal{P}$ is *hard to test*.

In what follows we denote by $\mathcal{P}_2, \mathcal{P}_3$ and $\mathcal{P}_4$ the paths of lengths 1, 2 and 3 (which have 2, 3 and 4 vertices, respectively), and by $\mathcal{C}_4$, the cycle of length 4. For a fixed graph H, let $\mathcal{P}^*_H$ denote the property of being induced H-free. Therefore, G satisfies $\mathcal{P}^*_H$ if and only if it contains no induced subgraph isomorphic to H. We define $\mathcal{P}_H$ to be the property of being (not necessarily induced) H-free. Therefore, G satisfies $\mathcal{P}_H$ if and only if it contains no copy of H. Thus, for example, for $H = C_4$, any clique of size at least 4 satisfies $\mathcal{P}^*_H$ but does not satisfy $\mathcal{P}_H$.

Our first result in this chapter is the following:

Theorem 5.1. *Let H be a fixed undirected graph other than $\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4, \mathcal{C}_4$ and their complements. Then, there exists a constant $c = c(H) > 0$ such that the query-complexity of any one-sided error ϵ-tester for $\mathcal{P}^*_H$ is at least*

$$\left(\frac{1}{\epsilon}\right)^{c\log(1/\epsilon)}.$$

As $\mathcal{P}_2$-freeness can obviously be tested with query complexity $\Theta(1/\epsilon)$, the following theorem, together with the above theorem, supplies a complete characterization for the

graphs H for which $\mathcal{P}^*_H$ is easily testable, except for the case of $\mathcal{P}_4$, $\mathcal{C}_4$ and its complement (the complement of $\mathcal{P}_4$ is also $\mathcal{P}_4$).

Theorem 5.2. *There is a one-sided error property tester for testing $\mathcal{P}_3$-freeness, with query complexity*

$$O(\log(1/\epsilon)/\epsilon).$$

We also prove the following theorem, which is analogous to Theorem 5.1, only with respect to directed graphs (digraphs, for short).

Theorem 5.3. *Let H be a fixed digraph on at least 5 vertices. Then, there exists a constant $c = c(H) > 0$ such that the query-complexity of any one-sided error ϵ-tester for $\mathcal{P}^*_H$ is at least*

$$\left(\frac{1}{\epsilon}\right)^{c\log(1/\epsilon)}.$$

We can actually prove a super-polynomial (in $1/\epsilon$) lower bound for the query complexity of $\mathcal{P}^*_H$ for some of the digraphs H on at most 4 vertices as well, see Subsection 5.3.1.

We finally show that Theorems 5.1 and 5.3 can also be extended to the cases of two-sided error property testers.

Theorem 5.4. *The lower bounds of Theorems 5.1 and 5.3 hold for two-sided error property testers as well.*

An interesting consequence of the above results is that the class of graphs for which $\mathcal{P}^*_H$ is easily testable, is nearly trivial (as it contains graphs on at most 4 vertices), however, it is provably not totally trivial, as $\mathcal{P}^*_{P_3}$ is easily testable. Note also the sharp dichotomy between the efficient one-sided error property-testers for $\mathcal{P}^*_{P_2}$ and $\mathcal{P}^*_{P_3}$, and the fact that for almost all the other graphs H, the property $\mathcal{P}^*_H$ has no property tester with query complexity polynomial in $1/\epsilon$ even if one is willing to settle for *two-sided* error.

Organization: The proof of Theorem 5.2 appears in Section 5.2. The lower bound proved by Theorem 5.1 is established in section 5.3. To prove this result we have to construct, for any graph H (other than the ones mentioned in the theorem) and any small $\epsilon > 0$, a graph G which is ϵ-far from being induced H-free and yet contains relatively few induced copies of H. The proof of this part, described in Section 5.3, uses the approach of [1] but requires several additional ideas. It applies certain constructions in additive number theory, based on (simple variants of) the construction of Behrend [29] of dense subsets of the first n integers without three-term arithmetic progressions. The proof of Theorem 5.3 also appears in Section 5.3. In Section 5.4 we give the proof of Theorem 5.4 which extends the lower bounds of Theorems 5.1 and 5.3 to the more general cases of two-sided error property-testers. The final section, Section 5.6, contains some concluding remarks and open problems.

Throughout this chapter we assume, whenever this is needed, that the number of vertices n of the graph or digraph G considered is sufficiently large, and that the error parameter ϵ

is sufficiently small. In order to simplify the presentation, we omit all floor and ceiling signs whenever these are not crucial, and make no attempt to optimize the absolute constants. When we later refer to a graph H as being easy/hard to test, we mean that $\mathcal{P}^*_H$ is easy/hard to test.

5.2 An Easily Testable Induced Graph Property

In this section we describe the proof of Theorem 5.2. For ease of notation, we denote by $\mathcal{P}$ the property $\mathcal{P}^*_{P_3}$, that is, being induced $\mathcal{P}_3$-free. The property-tester for $\mathcal{P}$ works as follows: it picks a random subset of, say, $10\log(1/\epsilon)/\epsilon$ vertices, and checks if there is an induced copy of $\mathcal{P}_3$ spanned by the set. It declares G to be induced $\mathcal{P}_3$-free if and only if it finds no induced copy of $\mathcal{P}_3$. If G satisfies $\mathcal{P}$, the algorithm clearly always answers correctly. We therefore only have to show that if G is ϵ-far from satisfying $\mathcal{P}$, the algorithm finds an induced copy of $\mathcal{P}_3$ with probability at least 2/3.

Let *High* denote the set of vertices of such a graph G whose degree is at least $\frac{\epsilon}{4}n$. Note that intuitively, the vertices that belong to $High$ have the highest contribution to G being ϵ-far from satisfying $\mathcal{P}$. We formulate this intuition as follows:

Claim 5.5. *Let $W \subseteq V(G)$ contain all but at most $\frac{\epsilon}{4}n$ of the vertices of* High. *Then the induced subgraph of G on W is at least $\frac{\epsilon}{2}$-far from satisfying $\mathcal{P}$.*

Proof: Assume this is not the case. Then we can make less than $\frac{\epsilon}{2}n^2$ changes within W and get a graph that contains no induced copy of $\mathcal{P}_3$ within W. We then remove all the edges that touch a vertex not in $High \cup W$ (as these vertices do not belong to *High*, there are at most $n \cdot \frac{\epsilon}{4}n$ such edges), and any edge that touches a vertex in $High \setminus W$ (there are at most $\frac{\epsilon}{4}n \cdot n$ such edges as by the assumption $|High \setminus W| \leq \frac{\epsilon}{4}n$). We thus get a graph that satisfies $\mathcal{P}$. As altogether we make less than ϵn^2 changes in G, this contradicts the assumption that G is ϵ-far from satisfying $\mathcal{P}$. □

We call a set $A \subseteq V(G)$ *Good* if all but at most $\frac{\epsilon}{4}n$ of the vertices that belong to *High* have a neighbor in A.

Claim 5.6. *A randomly chosen subset $A \subseteq V(G)$ of size $8\log(1/\epsilon)/\epsilon$ is* Good *with probability at least 7/8.*

Proof: Let A be a randomly chosen subset of size $8\log(1/\epsilon)/\epsilon$, and consider a vertex $v \in High$. As v has at least $\frac{\epsilon}{4}n$ neighbors, the probability that A does not contain any neighbor of v is at most

$$\left(1 - \frac{\epsilon}{4}\right)^{8\log(1/\epsilon)/\epsilon} \leq \epsilon^2 \leq \epsilon/32,$$

where we assumed that $\epsilon < 1/32$. As $High$ is of size at most n, we conclude that the expected number of vertices that belong to $High$ and have no neighbor in A, is at most $\frac{\epsilon}{32}n$. By Markov's inequality, with probability at least 7/8, the number of these vertices is at most $\frac{\epsilon}{4}n$. □

We will use the following simple and well known observation about the structure of induced $\mathcal{P}_3$-free graphs: A graph is induced $\mathcal{P}_3$-free if and only if it is the disjoint union of cliques.

Proof of Theorem 5.2: We first choose a random subset A of size $8\log(1/\epsilon)/\epsilon$, and assume that it is *Good*. If A contains an induced copy of $\mathcal{P}_3$ we are done. Otherwise, let W be the set of all the vertices $v \in V \setminus A$ that have at least one neighbor in A. As G is by assumption ϵ-far from satisfying $\mathcal{P}$, and A is by assumption *Good*, we get from Claim 5.5 that the induced subgraph on W is at least $\frac{\epsilon}{2}$-far from satisfying $\mathcal{P}$.

As we assumed that A contains no induced copy of $\mathcal{P}_3$, we get that there is a unique partition of A into cliques $C_1, \ldots, C_r$. If a vertex $v \in W$ is connected to $u \in C_i \subseteq A$, it follows that if W can be partitioned into cliques $D_1, \ldots, D_k$, where for $1 \leq i \leq r$, $C_i \subseteq D_i$, then v would have to belong to D_i. For each vertex $v \in W$ that is connected to $u \in C_i \subseteq A$, assign v the number i. If v is connected to vertices in A that belong to different C_i, then pick any of these numbers. This numbering induces a partition of all the vertices of W into r subsets. As W is at least $\frac{\epsilon}{2}$-far from satisfying $\mathcal{P}$, there are at least $\frac{\epsilon}{2}n^2$ pairs of vertices $u, v \in W$, such that either u and v should belong to the same D_i, but u and v are not connected, or u and v should belong to different subsets D_i, yet u and v are connected. Therefore, choosing a set B of $8/\epsilon$ randomly chosen pairs of vertices fails to find such a violating pair with probability at most $(1-\epsilon/2)^{8/\epsilon} \leq \frac{1}{8}$. By Claim 5.6, the probability of A failing to be *Good* is at most $\frac{1}{8}$, and the probability of B not containing any of the required pairs of vertices is also at most $\frac{1}{8}$. Hence, with probability at least $\frac{3}{4}$ the induced subgraph on $A \cup B$ is not induced $\mathcal{P}_3$-free. As $|A| + |B| = O(\log(1/\epsilon)/\epsilon)$ the proof is complete[1]. $\square$

5.3 Hard to Test Graphs and Digraphs

In this section we give the proofs of Theorems 5.1 and 5.3. The approach uses a construction in additive number theory, which uses the technique of Behrend [29], used to construct dense sets of integers with no three-term arithmetic progressions. A set $X \subseteq [m] = \{1, 2, \ldots, m\}$ is called *h-sum-free* if for every pair of positive integers $a, b \leq h$, if $x, y, z \in X$ satisfy the equation $ax + by = (a+b)z$ then $x = y = z$. That is, whenever $a, b \leq h$, the only solution to the equation that uses values from X, is one of the $|X|$ trivial solutions. We need the following lemma (a similar one appears in [54]):

Lemma 5.7. *For every positive integer m, there exists an h-sum-free subset $X \subset [m] = \{1, 2, \ldots, m\}$ of size at least*

$$|X| \geq \frac{m}{e^{10\sqrt{\log h \log m}}} \tag{5.1}$$

[1]Recall that we measure query complexity by the size of the sample of vertices and not the number of edge queries.

Proof: Let d and r be integers (to be chosen later) and define

$$S_r = \left\{ \sum_{i=0}^{k} x_i d^i \ : \ x_i < \frac{d}{2h} \ (0 \leq i \leq k) \ \wedge \sum_{i=0}^{k} x_i^2 = r \right\},$$

where $k = \lfloor \log m / \log d \rfloor - 1$. For the rest of the proof, the best way to view the numbers $x \in S_r$ is as represented in base d, where $x_k, \ldots, x_0$ are the "digits" of x. Also, note that by the choice of k, for any r we have $S_r \subseteq [m]$.

We claim that for every d and r, S_r is h-sum-free. Assume to the contrary that there are $x, y, z \in S_r$ that satisfy the equation $ax + by = (a+b)z$, where $a, b \leq h$ are positive integers and

$$x = \sum_{i=0}^{k} x_i d^i, \qquad y = \sum_{i=0}^{k} y_i d^i, \qquad z = \sum_{i=0}^{k} z_i d^i.$$

As by definition $x_i, y_i, z_i < d/2h$, and $a, b \leq h$ we conclude that there is no carry in the base d addition of the numbers in S_r. In other words, we have for every $0 \leq i \leq k$

$$ax_i + by_i = (a+b)z_i.$$

This means that z_i is a weighted average of x_i and y_i. Combined with the fact that the function $f(z) = z^2$ is convex, Jensen's inequality implies that

$$ax_i^2 + by_i^2 \geq (a+b)z_i^2,$$

and that the inequality is strict unless all three numbers x_i, y_i and z_i are equal. However, if for some i the inequality is strict, we have

$$a \sum_{i=0}^{k} x_i^2 + b \sum_{i=0}^{k} y_i^2 > (a+b) \sum_{i=0}^{k} z_i^2$$

which is impossible as by definition of S_r

$$\sum_{i=0}^{k} x_i^2 = \sum_{i=0}^{k} y_i^2 = \sum_{i=0}^{k} z_i^2 = r.$$

Thus, $x_i = y_i = z_i$ for all i, and S_r is indeed h-sum-free.

We complete the proof by showing that for some r, the set S_r is of the required size in (5.1). As the "digits" in any set S_r are bounded by $d/2h$, the integer r in the definition of S_r satisfies $r \leq (k+1)(d/2h)^2 < kd^2$. For the same reason, the union of the sets S_r has size $(d/2h)^{k+1} > (d/2h)^k$. It follows that for some r, the set S_r satisfies $|S_r| > (d/2h)^k / kd^2$. Setting $d = e^{\sqrt{\log m \log h}}$ (and therefore $k \approx \sqrt{\log m / \log h}$), we obtain (5.1) as needed. □

We proceed with the proofs of Theorems 5.1 and 5.3. It is convenient to start the discussion with digraphs and then obtain the results for undirected graphs as a special case,

(as they can be viewed as symmetric digraphs).

An *s-blow-up* of a digraph $H = (V(H), E(H))$ on h vertices is the digraph obtained from H by replacing each vertex $v_i \in V(H)$ by an independent set I_i of size s, and each directed edge $(v_i, v_j) \in E(H)$, by a complete bipartite directed subgraph whose vertex classes are I_i and I_j, and whose edges are directed from I_i to I_j. Note that if we take an s-blow-up of H, we get a digraph on sh vertices that contains s^h *induced* copies of H, where each vertex of the copy belongs to a different blow-up of a vertex from H (simply pick one vertex from each independent set). We call these induced copies the *special copies* of the blow-up. As each pair of vertices in the blow-up is contained in at most s^{h-2} special copies of H, it follows that adding or removing an edge from the graph can destroy at most s^{h-2} special copies of H. We conclude that one must add or remove at least $s^h/s^{h-2} = s^2$ edges from the blow-up in order to destroy all its special copies of H.

For the proofs of Theorems 5.1 and 5.3, we will need the following lemma, in which a triangle in a digraph is simply three vertices u, v, w, such that there is at least one edge between each of the three pairs.

Lemma 5.8. *For every fixed digraph H on h vertices, that contains at least one triangle, there is a constant $c = c(H) > 0$, such that for every positive $\epsilon < \epsilon_0(H)$ and every integer $n > n_0(\epsilon)$, there is a digraph G on n vertices which is ϵ-far from being induced H-free, and yet contains at most $\epsilon^{c \log(1/\epsilon)} n^h$ induced copies of H.*

Proof: Given a small $\epsilon > 0$, let m be the largest integer satisfying

$$\frac{1}{h^4 e^{10\sqrt{\log m \log h}}} \geq \epsilon. \tag{5.2}$$

It is easy to check that this m satisfies

$$m \geq \left(\frac{1}{\epsilon}\right)^{c \log(1/\epsilon)}, \tag{5.3}$$

for an appropriate $c = c(h) > 0$. Let $X \subset \{1, 2, \ldots, m\}$ be as in Lemma 5.7. Call the vertices of H $v_1, \ldots, v_h$, and let $V_1, V_2, \ldots V_h$ be pairwise disjoint sets of vertices, where $|V_i| = im$ and we denote the vertices of V_i by $\{1, 2, \ldots, im\}$, where, with a slight abuse of notation, we think of the sets V_i as being pairwise disjoint. We now construct a graph F whose vertex set is the union of the sets $V_1, \ldots, V_h$. For each j, $1 \leq j \leq m$, for each $x \in X$ and for each directed edge (v_p, v_q) of H, let $j + (p-1)x \in V_p$ have an outgoing edge pointed to $j + (q-1)x \in V_q$. In other words, for each $1 \leq j \leq m$ and $x \in X$, the graph F contains a copy of H, which is spanned by the vertices $j, j+x, j+2x, \ldots, j+(h-1)x$. Note that each of these $m|X|$ copies of H is spanned by a set of vertices that forms an arithmetic progression whose first element is j and whose difference is x. A crucial implication is that F contains $m|X|$ copies of H, such that each pair of copies have at most one common vertex. As each edge of F belongs to one of these copies, these $m|X|$ copies of H in F are in particular *induced*. In what follows we call these $m|X|$ induced copies of H in F, the *essential* copies

of H in F. Finally, define

$$s = \left\lfloor \frac{n}{|V(F)|} \right\rfloor = \left\lfloor \frac{2n}{h(h+1)m} \right\rfloor$$

and let G be the s-blow-up of F (together with some isolated vertices, if needed, to make sure that the number of vertices is precisely n). Claims 5.9 and 5.10 below complete the proof of this lemma. □

Claim 5.9. *The digraph G defined in the proof of Lemma 5.8 is ϵ-far from being induced H-free.*

Proof: The main idea of the proof is to show that adding or removing an edge from G can destroy special copies of H that belong to *at most* one of the blow-ups of the essential copies of H in F. To this end, consider two essential copies of H in F, H_1 and H_2. As was noted above, H_1 and H_2 are induced copies of H in F, which share at most one vertex in F. It follows that their corresponding blow-ups in G, denoted by T_1 and T_2, will share at most one common independent set. As T_1 and T_2 share at most one common independent set, a special copy of H in T_1 and a special copy of H in T_2 share at most one common vertex (recall that a special copy in a blow-up of H has precisely one vertex in each of the independent sets). We conclude that adding or removing an edge from G, can either destroy special copies of H that belong to T_1, or special copies of H that belong to T_2 (or not destroy any copies at all). As was explained above, in order to destroy all the special copies of an s-blow-up of H, one needs to add or remove at least s^2 edges from the blow-up. As G contains $m|X|$ blow-ups of essential copies of H, and each of these essential copies is induced in F, we conclude that one has to add or delete at least

$$s^2 m|X| = \frac{4n^2 m|X|}{h^2(h+1)^2 m^2} \geq \frac{|X|n^2}{h^4 m} \geq \frac{n^2}{h^4 e^{10\sqrt{\log m \log h}}} \geq \epsilon n^2 \tag{5.4}$$

edges in order to make G induced H-free. The second inequality follows from the lower bound on $|X|$ guaranteed by Lemma 5.7, and the third from (5.2). We conclude that G is indeed ϵ-far from being induced H-free. □

Claim 5.10. *The digraph G defined in the proof of Lemma 5.8 contains at most $\epsilon^{c \log(1/\epsilon)} n^h$ induced copies of H.*

Proof: As H contains at least one triangle, and each triangle belongs to at most $\binom{n}{h-3} \leq n^{h-3}$ copies of H, it is enough to show that G contains at most $\epsilon^{c \log(1/\epsilon)} n^3$ triangles. Consider a partition of the vertices of G into h subsets $U_1, \ldots, U_h$, where U_i contains the im independent sets that resulted from the blow-ups of the im vertices that belonged to V_i in F. Notice that if we show that the induced subgraph of G on any three of the subsets $U_1, \ldots, U_h$ contains at most $\epsilon^{c' \log(1/\epsilon)} n^3$ triangles, then the total number of triangles in G is at most $\binom{h}{3} \epsilon^{c' \log(1/\epsilon)} n^3$, which is still at most $\epsilon^{c \log(1/\epsilon)} n^3$.

Fix any three subsets U_i, U_j, U_k such that $1 \leq i < j < k \leq h$. Recall that G is a blow-up of F, and that we denote by I_v the independent set of vertices in G, which replaced the

vertex $v \in V(F)$. As there are no edges within these sets any triangle spanned by them must have exactly one vertex in each set. Note, that if the sets span a triangle whose vertices belong to the independent sets $I_x \subseteq U_i,\ I_y \subseteq U_j,\ I_z \subseteq U_z$, then as G is a blow-up of F, the vertices $x \in V_i,\ y \in V_j,\ z \in V_k$ in F must also span a triangle. Conversely, if $x \in V_i,\ y \in V_j,\ z \in V_k$ span a triangle in F, then for every choice of three vertices $u \in I_x \subseteq U_i,\ v \in I_y \subseteq U_j,\ w \in I_z \subseteq U_k$, the vertices u, v, w span a triangle in G. It follows that the number of triangles spanned by U_i, U_j, U_k is exactly s^3 times the number of triangles spanned by V_i, V_j, V_k.

If the vertices v_i, v_j, v_k, do not span a triangle in H, then by the definition of F, V_i, V_j, V_k do not span a triangle, and so do U_i, U_j, U_k in G, and we are done. If v_i, v_j, v_k span a triangle in H, then by the definition of F for any triangle spanned by V_i, V_j, V_k, there are $x, y \in X$ and $1 \leq t \leq im$, such that the three vertices of this triangle are

$$t \in V_i, \quad t + (j-i)x \in V_j, \quad t + (j-i)x + (k-j)y \in V_k.$$

The reason is that by definition of F, any edge from V_i to V_j connects some integer $t \in V_i$ to another integer $t + (j-i)x \in V_j$, where $x \in X$. The same applies also to edges connecting vertices from V_j to V_k. As this is a triangle, there must also be an edge connecting $t \in V_i$ to $t + (j-i)x + (k-j)y \in V_k$, hence there is some $z \in X$ such that

$$t + (k-i)z = t + (j-i)x + (k-j)y.$$

We conclude that the following equation in positive coefficients, whose values are at most h (recall $1 \leq i < j < k \leq h$), holds

$$(j-i)x + (k-j)y = (k-i)z.$$

As X is h-sum free, it follows that $x = y = z$. Therefore, V_i, V_j, V_k span precisely $m|X|$ triangles, which are spanned by the vertices

$$t + (i-1)x \in V_i, \qquad t + (j-1)x \in V_j, \qquad t + (k-1)x \in V_k,$$

for every possible choice of $t \in \{1, \ldots, m\}$ and $x \in X$. We conclude that U_i, U_j, U_k span

$$m|X|s^3 < m^2(n/m)^3 \leq n^3/m$$

triangles. As by (5.3), $m \geq (1/\epsilon)^{c \log(1/\epsilon)}$, the proof is complete. $\square$

The proofs of Theorems 5.1 and 5.3 now follow easily from the above lemma.

Proof of Theorem 5.1: Let H be a fixed graph on h vertices. A simple yet crucial observation is that for every graph H testing $\mathcal{P}^*_H$ is equivalent to testing $\mathcal{P}^*_{\overline{H}}$, where $\overline{H}$ is the complement of H. Note, that this relation does not hold for testing $\mathcal{P}_H$. Thus, in order to prove a lower bound for testing $\mathcal{P}^*_H$, we may prove a lower bound for testing $\mathcal{P}^*_{\overline{H}}$.

Recall that given a one-sided error ϵ-tester for testing $\mathcal{P}^*_H$ we may assume, without loss

of generality, that it queries all pairs of a uniformly at random chosen set of vertices. As the algorithm is a one-sided-error algorithm, it can report that G does not satisfy $\mathcal{P}^*_H$ only if it finds an induced copy of H in it. Observe, that if the tester picks a random subset of x vertices, and an input graph contains only $\epsilon^{c\log(1/\epsilon)}n^h$ induced copies of H, then the expected number of induced copies of H spanned by x is at most $x^h\epsilon^{c\log(1/\epsilon)}$, which is far smaller than 1 unless x exceeds $(1/\epsilon)^{c'\log(1/\epsilon)}$ for some $c' = c'(H) > 0$. It follows by Markov's inequality that the tester finds an induced copy of H with negligible probability. It is therefore enough to show that for any undirected graph H, other than $\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4, \mathcal{C}_4$ and their complements, there is a graph G on n vertices which is ϵ-far from satisfying $\mathcal{P}^*_H$, yet contains only $\epsilon^{c\log(1/\epsilon)}n^h$ induced copies of H. Combined with the first paragraph of this proof, it is enough to show this for either H or $\overline{H}$.

If $h \geq 6$, then it follows from the simplest result in Ramsey Theory (c.f., e.g., [80], page 1) that either H or $\overline{H}$ must contain a triangle. Hence, assuming that H contains a triangle, we can use Lemma 5.8 to construct a graph G on n vertices which is ϵ-far from satisfying $\mathcal{P}^*_H$ and yet contains at most $\epsilon^{c\log(1/\epsilon)}n^h$ induced copies of H. For $h = 5$, the only graph H, such that neither H nor $\overline{H}$ contains a triangle is C_5 (the cycle of length 5, whose complement is also C_5). In this case we can use the fact that C_5 is the core of itself to prove that $\mathcal{P}^*_{C_5}$ is not easily testable. See Subsection 5.3.1 for more details. As for $h = 2, 3, 4$ the only graphs H for which H and $\overline{H}$ are triangle-free are $\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4, \mathcal{C}_4$ and their complements, the proof is complete. □

Proof of Theorem 5.3: The proof is similar to the proof of Theorem 5.1. One only has to note again that for every digraph H, on at least 6 vertices, either H or $\overline{H}$ contains a triangle, and that the only digraph on 5 vertices which does not have this property is the digraph D_5 obtained from C_5, by replacing each undirected edge with two anti-parallel directed edges. We discuss this special case in Subsection 5.3.1. Though the theorem does not explicitly state it, we can also conclude from Lemma 5.8 that the same lower bound applies for any digraph H on 3 or 4 vertices such that either H or $\overline{H}$ contains a triangle. In Subsection 5.3.1 we discuss more digraphs for which we can obtain similar bounds. □

5.3.1 Graphs which are cores of themselves

In this subsection we briefly argue how to use the results of [11], that appear in the next chapter, in order to obtain lower bounds for some digraphs on 3,4 and 5 vertices. We first need some definitions. A homomorphism from digraph H to digraph K is a mapping $\varphi : V(H) \mapsto V(K)$ that maps edges of H to edges of K, i.e. $(u, v) \in E(H) \Rightarrow (\varphi(u), \varphi(v)) \in E(K)$. The *core* of a digraph H, is the smallest *subgraph* of H (with respect to number of edges) to which there is a homomorphism from H. In [11] the authors establish a lower bound similar to those of Theorems 5.1 and 5.2 for testing $\mathcal{P}_H$ for any digraph H whose core contains at least one cycle of length at least 3. As in the proof of Lemma 5.8, the main ingredient of the proof (Lemma 8 in [11]) is a construction of a digraph that is ϵ-far from being H-free and yet contains relatively few copies of H. Though it is not explicitly stated in [11], in case H is the core of itself, the constructed graph is actually also ϵ-far from being

induced H-free, and contains relatively few *induced* copies of H. Thus we can use the result of [11] to obtain similar lower bounds for any digraph H on 3,4 or 5 vertices such that either H or $\overline{H}$ is the core of itself and contains a cycle of length at least 3. This in particular holds for C_5, and therefore also for D_5, as testing $\mathcal{P}_{C_5}$ is a special case of testing $\mathcal{P}_{D_5}$. As was noted in [11], any directed cycle C that contains a non equal number of forward edges and backward edges is the core of itself. Thus, any digraph on 4 vertices that contains such a cycle of length 4 (e.g. a Hamilton cycle) is the core of itself, and we can use the result of [11] to obtain a lower bound for this case as well.

5.4 Two-Sided Error Testers

For the proof of Theorem 5.4 we apply Yao's principle [115], by constructing, for every fixed graph H, for which a lower bound was established in Theorems 5.1 and 5.3, two distributions D_1 and D_2, where D_1 consists of graphs which are ϵ-far from satisfying $\mathcal{P}^*_H$ with probability $1 - o(1)$ (where the $o(1)$ term tends to 0 as ϵ tends to zero), while D_2 consists of graphs which satisfy $\mathcal{P}^*_H$. We then show that any deterministic algorithm, which makes a small number of queries (adaptively) cannot distinguish with non-negligible probability between D_1 and D_2. We prove Theorem 5.4 for the case of digraphs, as it is clear that the case of undirected graphs follows as a special case. For the case of H being the graph obtained from C_5 by replacing each edge by a cycle of length 2, we can use the fact that this graph is the core of itself (as we did for one sided error in Subsection 5.3.1) to prove that $\mathcal{P}^*_{C_5}$ has no two-sided ϵ-tester with query complexity polynomial in $1/\epsilon$. We thus assume that H is a graph on at least 6 vertices. As in the proofs of Theorems 5.1 and 5.3, testing $\mathcal{P}^*_H$ with two-sided error has the same query complexity as testing $\mathcal{P}^*_{\overline{H}}$, thus we assume that H contains at least one triangle.

Proof of Theorem 5.4: Let H be a fixed digraph which contains at least one triangle. Given n and ϵ, let X, m and the sets V_i be as in the proof of Lemma 5.8. Construct the digraph F just as in the proof of Lemma 5.8, and remember that it consists of $m|X|$ pairwise edge disjoint copies of H which we called the essential copies of H in F (though it may well contain additional copies of H).

To construct D_1 which consists of digraphs that are ϵ-far from satisfying $\mathcal{P}^*_H$ with high probability, we first construct F'_1 by removing each of the $m|X|$ essential copies of H, randomly and independently, with probability $1 - 1/|E(H)|$. We then create G_1 by taking an s blow up of F'_1, adding isolated vertices, if needed. Finally, D_1 consists of all randomly permuted copies of such digraphs G_1. It follows from a standard Chernoff bound, that with probability $1 - o(1)$, at least $m|X|/2|E(H)|$ essential copies of H are left in F'_1, where the $o(1)$ term tends to 0, as ϵ tends to 0. Similar to the derivation of (5.4), it is easy to show that if $m|X|/2|E(H)|$ of these copies of H are left in F'_1, the graph G_1 is ϵ-far from satisfying $\mathcal{P}^*_H$. It follows that with probability $1 - o(1)$, a member of D_1 is ϵ-far from satisfying $\mathcal{P}^*_H$.

The distribution D_2 of digraphs that satisfy $\mathcal{P}^*_H$, is defined by first constructing F'_2 by randomly, independently and uniformly picking from each of the $m|X|$ essential copies of H a single edge, and removing all the other edges of that copy. We then create G_2 by taking

an s blow up of F_2' adding isolated vertices, if needed. Finally, D_2 consists of all randomly permuted copies of such digraphs G_2. The main argument of Lemma 5.8, states that the graph F defined in the lemma contains only triangles whose three edges belong to one of the essential copies of H. Hence, keeping a single edge from each of these copies results in a triangle free graph, and in particular all the graphs in G_2 satisfy $\mathcal{P}_H^*$.

As in the proof of Lemma 5.8, denote by I_v the independent set of vertices in G_1 (or G_2) that replaces the vertex $v \in V(F)$. Now consider a set of vertices S in G_1 (or G_2) and its natural projection to a subset of $V(F)$ (namely, for each vertex $u \in I_v$ we consider the vertex v in F) which we also denote by S with a slight abuse of notation. Suppose S has the property that it does not contain more than two vertices from any one of the essential copies of H.

If this property holds, then each edge spanned by S is contained in a different essential copy of H. Therefore, each edge has probability $1/|E(H)|$ of being in F_1', and these probabilities are mutually independent. Similarly, each such edge has probability $1/|E(H)|$ of being in F_2' and these probabilities are also mutually independent. It follows that in this case, sampling a digraph G from D_1, and looking at the induced digraph on a set S with the above property, has *exactly* the same distribution as sampling a digraph G from D_2, and looking at the induced digraph on S.

In order to apply Yao's principle and thus complete the proof, we have to show that no deterministic algorithm can distinguish between the distributions D_1 and D_2 with constant probability. To this end, it is clearly enough to show that with probability $1 - o(1)$, any deterministic algorithm that looks at a digraph spanned by less than $(1/\epsilon)^{c' \log 1/\epsilon}$ vertices, has *exactly* the same probability of seeing any digraph regardless of the distribution from which the digraph was chosen. By the discussion in the previous paragraph, this can be proved by establishing that, with high probability, a small set of vertices does not contain three vertices from the same essential copy of H. For a fixed ordered set of three vertices in S, consider the event that they all belong to the same essential copy of H. The first two vertices determine all the vertices of one of these copies uniquely. Now, the conditional probability that the third vertex is also a vertex of the same copy is $h/|V(F)| \leq 1/m$. By the union bound, the probability that the required property is violated is at most

$$|S|^3/m \leq |S|^3 \epsilon^{c \log 1/\epsilon}.$$

This quantity is $o(1)$ as long as $|S| = o((1/\epsilon)^{\frac{c}{3} \log 1/\epsilon})$, where here we applied the lower bound on the size of m given in (5.3). Therefore, if the algorithm has query complexity $o((1/\epsilon)^{c' \log 1/\epsilon})$ for some absolute positive constant c', it has probability $1 - o(1)$ of looking at a subset on which the distributions D_1 and D_2 are identical, thus, the probability that it distinguishes between D_1 and D_2 is $o(1)$. $\square$

A slightly more complicated argument than the above can give two distributions D_1 and D_2, such that the graphs in D_1 are *always* ϵ-far from satisfying $\mathcal{P}_H^*$, while the graphs in D_2 always satisfy $\mathcal{P}_H^*$. The idea is to first partition the $m|X|$ essential copies of H into groups of size $|E(H)|$ assuming for simplicity that $|E(H)|$ divides $m|X|$. To create D_1, we randomly pick from each group of $|E(H)|$ copies of H a single copy, and delete all its

edges. To create D_2, we do exactly the same as we did in the proof of Theorem 5.4. It is easy to appropriately modify the proof above in order to show that any deterministic algorithm with query complexity $o((1/\epsilon)^{c \log 1/\epsilon})$ can not distinguish between D_1 and D_2. As this argument has no qualitative advantage, we described the simpler one given above.

5.5 Additional Results

In this section we discuss some additional results that were not included in this thesis. It is natural to ask if the results of this chapter can be extended to k-uniform hypergraphs [2]. As a dense k-uniform hypergraph (k-graph for short) has $\Theta(n^k)$ edges, we say that a k-graph is ϵ-far from satisfying a property if one has to add/delete at least ϵn^k edges in order to get a a k-graph satisfying the property. We define testers for properties of k-graph in the obvious way. Given the results of this chapter it seems natural to try and extend them to k-graphs. Specifically, we can ask for which k-graphs H, it is possible to test the property of being induced H-free with a polynomial (in $1/\epsilon$) number of queries. It is clear that when H is a single edge (on k vertices) the property is easily testable. In a joint paper with Noga Alon [13], we have shown that aside from the case when H is an edge as well as a unique 3-graph on 4 vertices with 2 edges, all the other k-graphs H are such that the property of being induced H-free is hard to test. The proof of this result is significantly more involved compared to the proof of Theorem 5.3. The techniques we applied in [13] were also used in another joint work with Noga Alon [19] to solve a special case of a conjecture of Brown, Erdős and Sós from 1973. Let $f_k(n, v, e)$ denote the maximum number of edges in a k-uniform hypergraph on n vertices, which does not contain e edges spanned by v vertices. [38] and [39] raised the problem of estimating $f_k(n, e(k-r)+r+1, e)$ for fixed integers e and $2 \leq r < k$. Ruzsa and Szemerédi [110] have resolved the case $r = 2$, and $k = e = 3$ by showing that $n^{2-o(1)} < f_3(n, 6, 3) = o(n^2)$. Erdős, Frankl and Rödl extend their result to $r = 2$, $e = 2$ and any $k > 2$ and by showing that $n^{2-o(1)} < f_k(n, 3(k-2)+3, 3) = o(n^2)$. Using the techniques of [13] along with several additional ideas we have further extended the above results to arbitrary $2 \leq r < k$ and $e = 3$ by showing that $n^{k-o(1)} < f_k(n, 3(k-r)+r+1, 3) = o(n^k)$.

5.6 Concluding Remarks and Open Problems

- As in the case of $\mathcal{P}_H$, there is a huge gap between the general upper bounds for testing $\mathcal{P}^*_H$ that were established in [6], and the lower bounds in this chapter. It would be very interesting, and probably challenging, to improve any of these bounds. Even in the seemingly simplest case of H being a triangle, we do not know how to improve these bounds.

- Another interesting open problem is to complete the characterizations of easily testable properties $\mathcal{P}^*_H$ for undirected graphs H, by solving the cases of $H = P_4, C_4$ (recall

[2] A k-uniform hypergraph $G = (V, E)$ has vertex set V and edge set E where every edge $e \in E$ contains k distinct vertices from V. Thus, standard (simple) graphs are just 2-uniform hypergraphs.

that testing the complement of C_4 is equivalent to testing C_4). The case of testing $\mathcal{P}^*_{P_4}$ seems the simplest one to resolve, since there are known structural results, that characterize induced $\mathcal{P}_4$-free graphs. These graphs are also known as Complement Reducible graphs, or Cographs for short, and they are precisely the graphs formed from a single vertex under the closure of the operations of union and complement, see [43] and [97]. Cographs arise naturally in such application areas as examination scheduling and automatic clustering of index terms. Cographs have a unique tree representation called a Cotree. It might be possible to use this characterization, and the unique tree representation in order to design an efficient tester for $\mathcal{P}^*_{P_4}$.

- Combining Theorem 5.3 and Subsection 5.3.1, the only unclassified digraphs on 3 vertices are the graph obtained from $\mathcal{P}_3$ by replacing one edge with two anti-parallel edges, and the other by a single edge, and the graph obtained from $\mathcal{P}_3$ by replacing both edges with two anti-parallel edges. As all the digraphs on at least 5 vertices are hard to test, the only remaining unclassified digraphs are the digraphs H on 4 vertices, such that neither H nor $\overline{H}$ contains a triangle, and neither H nor $\overline{H}$ contains a cycle of length 4 that is the core of itself (e.g. the graph obtained from either C_4 or $\mathcal{P}_4$ by replacing each edge with two anti-parallel edges). It will be interesting to classify these cases as well.

- There is an interesting possible connection between the problem of graph isomorphism and testing $\mathcal{P}^*_H$. It is known (see [44]) that for any graph $H \in \{\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4, \mathcal{C}_4\}$, the graph isomorphism problem can be solved in polynomial time for induced H-free graphs. Moreover, for any other H, any instance of the graph isomorphism problem can be reduced to an instance that is induced H-free. Thus, in some sense, the problem on induced H-free graphs, for H other than $\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4$ and $\mathcal{C}_4$, is *isomorphism hard*. It might be interesting to understand if this connection is indeed meaningful.

Chapter 6

Testing Subgraph-Freeness in Directed Graphs

6.1 The Main Results

In this chapter we continue the investigation of the graph properties that can be tested with a small number of queries. Our main investigation in this chapter is testing properties of directed graphs (= digraphs for short), and we obtain a characterization of the directed graphs D for which the property of being D-free can is easily testable. We briefly note that the basic definitions we need for digraphs are the natural extensions of the definitions we have thus far used for undirected graphs. For example a property of digraphs is a family of digraphs closed under isomorphism, and a digraph is ϵ-far from satisfying property $\mathcal{P}$ if one must add/remove at least ϵn^2 directed edges in order to make the graph satisfy $\mathcal{P}$. For a fixed connected digraph H (with at least one edge), let $\mathcal{P}_H$ denote the property of being H-free. Therefore, G satisfies $\mathcal{P}_H$ if and only if it contains no (not necessarily induced) subgraph isomorphic to H. Our first result in this chapter is that for every fixed digraph H, the property $\mathcal{P}_H$ is testable.

Theorem 6.1. *For every fixed digraph H, the property $\mathcal{P}_H$ is testable with one-sided error.*

The proof relies on a variant of the regularity lemma of Szemérédi [112] adapted for directed graphs, which we formulate and prove. This version of the regularity lemma might prove useful for other problems. The application for getting the strong-testability of each property $\mathcal{P}_H$ is similar to the proof for the undirected case, given (implicitly) in [4], see also [6], [1].

The one-sided ϵ-tester for $\mathcal{P}_H$ for arbitrary digraphs H, has query-complexly bounded by a function which, though independent of the size of the input digraph G, has a huge dependency on ϵ and the size of H. For some digraphs H, however, there are more efficient ϵ-testers; for example, if H is a single directed edge, it is easy to see that there is a one-sided ϵ-tester for $\mathcal{P}_H$, which makes only $\Theta(1/\epsilon)$ queries. A natural question is therefore, to decide for which digraphs H can one design a one-sided error property tester for $\mathcal{P}_H$, whose

query complexity would be bounded by a polynomial in $1/\epsilon$. In what follows we call $\mathcal{P}_H$ *easily testable* if there is a one-sided error property-tester for $\mathcal{P}_H$ whose query complexity is polynomial in $1/\epsilon$. If such a property tester does not exist we say that $\mathcal{P}_H$ is *hard to test*.

Our main result here is a precise characterization of all digraphs H for which $\mathcal{P}_H$ is easily testable. We further show that the same characterization applies to two-sided error ϵ-testers as well. As a special case of the argument we conclude that for an undirected graph H, $\mathcal{P}_H$ has a two-sided ϵ-tester whose query complexity is polynomial in $1/\epsilon$ if and only if H is bipartite. This settles an open problem raised in [1]. Somewhat surprisingly, it turns out that if $\mathcal{P}_H$ is easily testable, then it has a two-sided error property-tester that samples only $\Theta(1/\epsilon)$ vertices, although any one-sided error ϵ-tester for $\mathcal{P}_H$ has to sample at least $(1/\epsilon)^{d/2}$ vertices, where d is the average degree of H.

Before continuing let us introduce the following standard terminology. We call a directed cycle of length 2, a 2-*cycles*. We call a cycle obtained from an undirected cycle by directing its edges an *oriented cycle*. An oriented cycle in which all edges point to the same direction is a *directed cycle*. *Oriented paths* and *directed paths* are defined in an analogous manner. A digraph is an *oriented tree* if it does not contain any oriented cycle. A digraph is *bipartite* if it does not contain any oriented cycle of odd length.

The characterization of the digraphs H, for which $\mathcal{P}_H$ is easily testable, relies on some properties of digraph homomorphisms and cores of digraphs. Let H and K be two digraphs. A function φ mapping vertices of H to vertices of K is a *homomorphism* if it satisfies $(u,v) \in E(H) \Rightarrow (\varphi(u), \varphi(v)) \in E(K)$. The *core* of a digraph H is the *subgraph* K of H with the smallest number of edges, for which there is a homomorphism from H to K. We can clearly assume that the core does not contain isolated vertices. It is also easy to see that this notion is well defined in the sense that up to isomorphism the core is unique. We refer the reader to [28] and [85] for more background and references on digraph homomorphisms, and to [84] for more information and references on cores of graphs. Our main result is the following precise characterization of the digraphs H for which testing $\mathcal{P}_H$ with one-sided error, has query complexity polynomial in $1/\epsilon$. Here, and throughout the chapter, we measure query-complexity by the number of vertices sampled, assuming we always examine all edges spanned by them.

Theorem 6.2. *Let H be a fixed connected digraph on h vertices, and let K be its core.*
(i) If K is a 2-cycle, then for every $\epsilon > 0$, there is a one-sided error ϵ-tester for $\mathcal{P}_H$ whose query-complexity is bounded by

$$O((1/\epsilon)^{h/2}).$$

(ii) If K is an oriented tree, then for every $\epsilon > 0$ there is a one-sided error ϵ-tester for $\mathcal{P}_H$ whose query-complexity is bounded by

$$O((1/\epsilon)^{h^2}).$$

(iii) If H is not as in (i), (ii), then there exists a constant $c = c(H) > 0$ such that the

query-complexity of any one-sided error ϵ-tester for $\mathcal{P}_H$ is at least

$$\left(\frac{1}{\epsilon}\right)^{c\log(1/\epsilon)}.$$

A special case of the first part of the above theorem improves the previous result from [1] which had query complexity $O((1/\epsilon)^{h^2})$.

We also prove the following theorem, that says that in case H is a tree, we can design an optimal ϵ-tester for $\mathcal{P}_H$.

Theorem 6.3. *If H is an oriented tree, then there is a one-sided error ϵ-tester for $\mathcal{P}_H$, with optimal query complexity*

$$\Theta(1/\epsilon).$$

The result in the last part of Theorem 6.2 can be extended to two-sided error ϵ-testers as well.

Theorem 6.4. *Let H be a fixed digraph on h vertices, and let K be its core.*
(i) If K is a 2-cycle or an oriented tree, then the property $\mathcal{P}_H$ has a two-sided error ϵ-tester with optimal query complexity

$$\Theta(1/\epsilon).$$

(ii) If K is neither a directed 2-cycle, nor an oriented tree, then there exists a constant $c = c(H) > 0$ such that the query-complexity of any two-sided error ϵ-tester for $\mathcal{P}_H$ is at least

$$\left(\frac{1}{\epsilon}\right)^{c\log(1/\epsilon)}.$$

It is not difficult to show, by considering an appropriate random digraph, that the one-sided error query complexity of $\mathcal{P}_H$ for any digraph H with average degree d is at least $(\frac{1}{\epsilon})^{d/2}$. Therefore, the first part of the theorem exhibits an interesting difference between the query complexity of the best one-sided and the best two-sided error ϵ-testers of $\mathcal{P}_H$ for many digraphs H. The second part of Theorem 6.4 implies a similar result for undirected non bipartite graphs, thus solving a problem raised in [1].

As is apparent from the statement of Theorem 6.2, the characterization of the digraphs H for which $\mathcal{P}_H$ is easily testable, is far more complicated than the characterization for undirected graphs, which states that $\mathcal{P}_H$ is easily testable if and only if H is bipartite. The characterization for undirected graphs is also simple in the sense that one can check it in polynomial time. It turns out that the characterization for digraphs is not complicated by chance, and in fact we show that the problem of deciding whether for a given digraph H, the property $\mathcal{P}_H$ is easily testable, is NP-complete. This fact follows easily by combining Theorem 6.2 with a theorem of Hell, Nesetril, and Zhu [85] about cores of digraphs.

Note, that although this implies that the problem of deciding if $\mathcal{P}_H$ is easily testable is hard for large digraphs H, this problem is interesting for small fixed digraphs as well, and for those the decision is simple. Thus, for example, Theorem 6.2 implies that the property $\mathcal{P}_C$ has a *polynomial* query complexity in $1/\epsilon$ for the oriented cycle C on the vertices $v_1, \ldots, v_{2k}$,

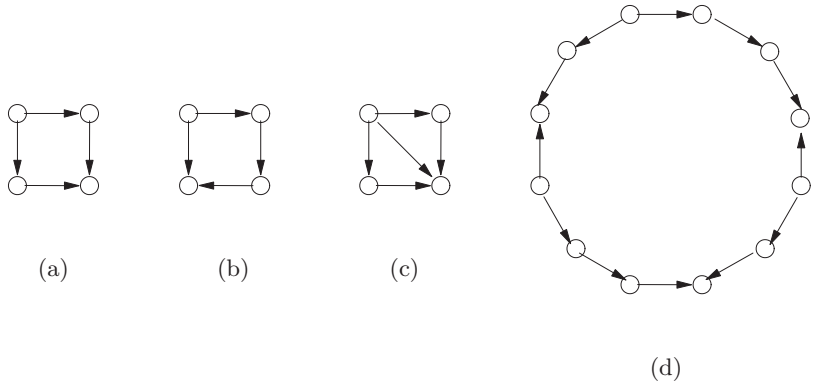

Figure 6.1: (a) Core is a path (b) Core is the entire digraph (c) Core is a triangle (d) Core is the entire digraph although the graph is balanced.

that consists of two edge-disjoint directed paths from v_1 to v_{k+1} (see Figure 6.1 (a)), as each path is a core of C. Theorem 6.2 also implies that the property $\mathcal{P}_{C'}$ has a *non-polynomial* query complexity in $1/\epsilon$ for every oriented cycle C' that is obtained from the above cycle C, by changing the direction of *any* single edge (see Figure 6.1 (b)), because in this case the core of C' is the entire digraph. This example shows that the testability of $\mathcal{P}_H$ does not rely solely on the structure of H as an undirected graph. Additional comments on this subject appear in Section 6.7.

Organization: The chapter is organized as follows: In Section 6.2, we modify some of the ideas used in the proof of Szemerédi's regularity lemma for undirected graphs, in order to prove a more general result that applies also to digraphs. In Section 6.3 we apply the above lemma in order to prove Theorem 6.1.

The main result consists of two parts. The first one (Theorem 6.2, parts (i),(ii)) appears in Section 6.4, and is proved using probabilistic arguments and tools from extremal graph theory. Unlike the corresponding result for undirected graphs, the techniques required here are rather complicated, and apply some delicate arguments. In this section we also prove Theorem 6.3. To prove the third part of Theorem 6.2, we have to construct, for any digraph H as in (iii) and any small $\epsilon > 0$, a digraph G which is ϵ-far from being H-free and yet contains relatively few copies of H. The proof of this part, described in Section 6.5, uses the approach of [1], but requires some additional ideas. It applies some properties of digraph homomorphisms as well as certain constructions in additive number theory, based on (simple variants of) the construction of Behrend [29] of dense subsets of the first n integers without three-term arithmetic progressions. In Section 6.6 we describe the proof of Theorem 6.4. We assume, throughout these three sections, that the underlying undirected graph of the digraph H considered is connected. In the final section, Section 6.7, we observe that it is easy to extend the results to the disconnected case and discuss the complexity of the problem of deciding whether for a given input digraph H, $\mathcal{P}_H$ is polynomially testable.

This final section contains some concluding remarks and open problems as well.

Throughout the chapter we assume, whenever this is needed, that the number of vertices n of the digraph G is sufficiently large, and that the error parameter ϵ, is sufficiently small. In order to simplify the presentation, we omit all floor and ceiling signs whenever these are not crucial, and make no attempt to optimize the absolute constants.

6.2 A Regularity Lemma for Digraphs

6.2.1 Statement of the new lemma

In this section we prove a regularity lemma for digraphs, by using some of the ideas in the proof of Szemerédi's regularity lemma for undirected graphs. For the proof of Szemerédi's regularity lemma the reader is referred to the original proof in [112], and to [48] which was used as a reference for the proof here. In order to state the lemma we need some definitions. Let $G = (V, E)$ be a digraph, and let $X, Y \subseteq V$ be disjoint. Let $\overrightarrow{E}(X, Y)$ denote the set of edges going from X to Y, and let $\overleftarrow{E}(X, Y)$ denote the set of edges going from Y to X. Let $\overline{E}(X, Y)$ denote the set of pairs of edges that form 2-cycles between X and Y. Define

$$\overrightarrow{d}(X,Y) := \frac{|\overrightarrow{E}(X,Y)|}{|X||Y|}, \qquad \overleftarrow{d}(X,Y) := \frac{|\overleftarrow{E}(X,Y)|}{|X||Y|}, \qquad \overline{d}(X,Y) := \frac{|\overline{E}(X,Y)|}{|X||Y|}$$

the *directed densities* of the pair (X, Y). Observe that all three densities of any pair are real numbers between 0 and 1. Given some $\epsilon > 0$, we call a pair (A, B) of disjoint sets $A, B \subseteq V$ *ϵ-regular* if all $X \subseteq A$ and $Y \subseteq B$ with

$$|X| \geq \epsilon|A| \quad \text{and} \quad |Y| \geq \epsilon|B|,$$

satisfy

$$|\overrightarrow{d}(X,Y) - \overrightarrow{d}(A,B)| \leq \epsilon, \qquad |\overleftarrow{d}(X,Y) - \overleftarrow{d}(A,B)| \leq \epsilon, \qquad |\overline{d}(X,Y) - \overline{d}(A,B)| \leq \epsilon.$$

We will later need the following trivial claim about a regular pair (A, B). The claim simply says that if we take a large enough subset $Y \subseteq B$, then for most vertices in the other side, Y behaves almost like B. In order to state the claim we need the following notation which will be used later as well: $\overrightarrow{N}_Y(v)$ is the set of vertices $y \in Y$ for which $(v, y) \in E$, $\overleftarrow{N}_Y(v)$ is the set of vertices $y \in Y$ for which $(y, v) \in E$ and $\overline{N}_Y(v)$ is the set of vertices $y \in Y$ for which (v, y) is a 2-cycle.

Claim 6.5. *Let (A, B) be an ϵ-regular pair with densities $\overrightarrow{d}, \overleftarrow{d}$ and $\overline{d}$, and let $Y \subseteq B$ be of size at least $\epsilon|B|$. Then for all but at most $3\epsilon|A|$ vertices $v \in A$, the inequalities $\overrightarrow{N}_Y(v) \geq (\overrightarrow{d} - \epsilon)|Y|$, $\overleftarrow{N}_Y(v) \geq (\overleftarrow{d} - \epsilon)|Y|$ and $\overline{N}_Y(v) \geq (\overline{d} - \epsilon)|Y|$ hold.*

Proof: Assume that for some X, such that $|X| \geq 3\epsilon|A|$, for all $v \in X$ at least one of the inequalities does not hold. Then for some $Z \subseteq X$, such that $|Z| \geq \epsilon|A|$, for all $v \in Z$ the

same inequality does not hold. Hence, the pair (Z, Y) contradicts the ϵ-regularity of the pair (A, B). $\square$

Consider a partition $\{V_0, V_1, \ldots, V_k\}$ of V in which one set V_0 has been singled out as an exceptional set (V_0 may be empty). We call such a partition an *ϵ-regular partition* of a digraph G if it satisfies the following three conditions:

(i) $|V_0| \leq \epsilon|V|$;

(ii) $|V_1| = \ldots = |V_k|$;

(iii) all but at most ϵk^2 of the pairs (V_i, V_j) with $1 \leq i < j \leq k$ are *ϵ-regular*.

Our objective is to prove the following generalization of Szemerédi's regularity lemma.

Lemma 6.6. *For every $\epsilon > 0$ and every $m \geq 1$ there exists an integer $DM = DM(m, \epsilon)$ such that every digraph of order at least m admits an ϵ-regular partition $\{V_0, V_1, \ldots, V_k\}$ with $m \leq k \leq DM$.*

The statement of the lemma for symmetric digraphs, that is, digraphs in which (u, v) is a directed edge if and only if (v, u) is a directed edge, is equivalent to the statement of the regularity lemma for undirected graphs.

6.2.2 The regularity lemma for undirected graphs

We start with the regularity lemma for undirected graphs, and some of the definitions used in the course of its proof. In the context of undirected graphs there is only one density between a pair of disjoint subsets $A, B \subseteq V$, and it is defined as $d(A, B) := |E(A, B)|/|A||B|$, where $E(A, B)$ is the set of edges between A and B. A pair of disjoint sets $A, B \subseteq V$ is *ϵ-regular* if all $X \subseteq A$ and $Y \subseteq B$ with $|X| \geq \epsilon|A|$ and $|Y| \geq \epsilon|B|$, satisfy $|d(X, Y) - d(A, B)| \leq \epsilon$.

An *ϵ-regular partition* is defined in a way analogous to the definition of a regular partition for digraphs. The following is Szemerédi's regularity lemma for undirected graphs

Lemma 6.7. [112] *For every $\epsilon > 0$ and every $m \geq 1$ there exists an integer $M = M(m, \epsilon)$ such that every graph of order at least m admits an ϵ-regular partition $\{V_0, V_1, \ldots, V_k\}$ with $m \leq k \leq M$.*

The proof for undirected graphs uses the following definitions that will be used in our proof as well. Let $G = (V, E)$ be a graph and $n = |V|$. For disjoint sets $A, B \subseteq V$ we define

$$q(A, B) = \frac{|A||B|}{n^2} d^2(A, B).$$

For a partition $P = \{C_1, \ldots, C_k\}$ of V we let

$$q(P) = \sum_{i<j} q(C_i, C_j).$$

However, if $P = \{C_0, C_1, \ldots, C_k\}$ has an exceptional set C_0, we treat C_0 as a set of singletons and define

$$q(P) = q(P'),$$

where $P' = \{C_1, \ldots, C_k\} \cup \{\{v\} : v \in C_0\}$.

It can be easily shown that for any partition P,

$$q(P) \leq \frac{1}{2}. \tag{6.1}$$

We say that a partition P' *refines* a partition P, if any (non exceptional) set in P is the union of some sets in P'. We will also need the following lemmas from [48] that establish relations between partitions and their refinements.

Lemma 6.8. *If P and P' are partitions of V, and P' refines P, then $q(P') \geq q(P)$.*

Lemma 6.9. *Let $0 \leq \epsilon \leq 1/4$ and let $P = \{C_0, C_1, \ldots, C_k\}$ be a partition of V, with exceptional set C_0, of size $|C_0| \leq \epsilon n$ and $|C_1| = \ldots = |C_k|$. If P is not ϵ-regular, then there is a partition $P' = \{C'_0, C'_1, \ldots, C'_\ell\}$ of V with exceptional set C'_0, where $k \leq \ell \leq k4^k$, such that $|C'_0| \leq |C_0| + n/2^k$,$C_0 \subseteq C'_0$, all other sets C'_i have equal size, and*

$$q(P') \geq q(P) + \epsilon^5/2.$$

Comment: Although the above claim in [48] does not explicitly state it, the partition P' is a refinement of P.

Note that combining Lemma 6.9 with (6.1), the proof of the regularity lemma for undirected graphs is immediate (up to some technicalities). We can apply Lemma 6.9 over and over again until we get an ϵ-regular partition. This must happen after at most $1/\epsilon^5$ iterations.

6.2.3 The proof of Lemma 6.6

Given a digraph $G = (V, E)$, and a partition of V, $P = \{C_1, \ldots, C_k\}$, consider a partition of E into 3 (not necessarily disjoint) sets

$$\overrightarrow{E} = \{(u, v) \in E : u \in C_i, v \in C_j, i < j\},$$

$$\overleftarrow{E} = \{(u, v) \in E : u \in C_i, v \in C_j, i > j\},$$

$$\overline{E} = \{(u, v) \in E : (v, u) \in E, u \in C_i, v \in C_j, i \neq j\}.$$

Now we can view a partition P as three different partitions $\overrightarrow{P}, \overleftarrow{P}, \overline{P}$, of undirected graphs (all three partition V in the same way, but the sets of edges among the partition sets are different). The first is obtained by removing any edge that does not belong to $\overrightarrow{E}$, and considering the directed edges as undirected. The second is obtained by removing any

edge that does not belong to $\overleftarrow{E}$, and again considering the directed edges as undirected. The third is obtained by removing any edge that does not belong to $\overline{E}$, and considering each cycle of length 2 as an undirected edge. We can also define the values $q(\overrightarrow{P})$, $q(\overleftarrow{P})$ and $q(\overline{P})$, as the function $q(\cdot)$ on a partition of V with edge sets $\overrightarrow{E}$, $\overleftarrow{E}$ and $\overline{E}$ respectively, by considering the directed edges and cycles of length 2, as undirected edges.

The key observation now, is that if the above three partitions are ϵ-regular in the context of undirected graphs, then P is an ϵ-regular partition in the context of directed graphs. Thus we can view the task of obtaining an ϵ-regular partition in a digraph, as the task of obtaining a partition that is ϵ-regular in the sense of undirected graphs, over three subsets of E. We next refer to $\overrightarrow{P}$, $\overleftarrow{P}$ and $\overline{P}$ sometimes not as a specific partition, but as the *set* of partitions of $\overrightarrow{E}$, $\overleftarrow{E}$ and $\overline{E}$ respectively, obtained in the course of creating the ϵ-regular partition.

Proof of Lemma 6.6: Let $G = (V, E)$ be given. For any partition P of V, we can define the partitions $\overrightarrow{P}$, $\overleftarrow{P}$ and $\overline{P}$ as described above. Also note that all three values $q(\overrightarrow{P}), q(\overleftarrow{P})$ and $q(\overline{P})$ are always at most $1/2$ by (6.1). Thus we can apply Lemma 6.9, circularly once for each partition until all three are ϵ-regular. For example, when we apply Lemma 6.9 to $\overrightarrow{E}$, we choose a new partition of V, according to the previous $\overrightarrow{P}$, and this induces a new partition of $\overleftarrow{P}$ and $\overline{P}$ as well. By the condition of Lemma 6.9 and the comment following it, this cannot happen more than $s = 3 \cdot 1/\epsilon^5 = 3/\epsilon^5$ times, before we obtain an ϵ-regular partition of the digraph G. Observe that, for example, when we apply Lemma 6.9 to $\overrightarrow{P}$, we do not necessarily increase $q(\overleftarrow{P})$ by $\epsilon^5/2$ (In fact, it might even be the case that $\overleftarrow{P}$ was an ϵ-regular partition of $\overleftarrow{E}$ and now it is not!), but by Lemma 6.8 and the comment following Lemma 6.9, we also do not decrease its value. Hence, in each iteration one of the values $q(\overrightarrow{P}), q(\overleftarrow{P}), q(\overline{P})$ is increased by at least $\epsilon^5/2$, while the other two do not decrease. An important technicality is that as the definitions of the partitions $\overrightarrow{P}$, $\overleftarrow{P}$ and $\overline{P}$ depend on the serial numbers given to the partition sets of $V(G)$ (see beginning of the subsection), we must make sure that if, for example, edge (u, v) was part of partition $\overrightarrow{P}$ then it does not "move" to another partition, say, $\overleftarrow{P}$. To this end, we can simply give consecutive serial numbers in the new partition, to all the subsets of a set that belongs to the previous partition.

We are left only with the simple technicalities of making sure that C_0 does not get too large, and of defining the function $DM(m, \epsilon)$. These are straightforward, and are left to the reader. See ,e.g., [48] pages 159-160. □

Note that our process for obtaining the regular partition does *not* apply the regularity lemma for undirected graphs recursively, and that the bound for the function $DM(\epsilon, k)$ in the lemma for digraphs is similar to the bound of the function $M(\epsilon, k)$ in the lemma for undirected graphs, that is, both

are towers of 2's of height $O(1/\epsilon^5)$. By a result of Gowers [74], both functions must grow at least as fast as a tower of 2's of height $poly(1/\epsilon)$.

6.3 Testing for Arbitrary Subgraphs

In this section we use our version of Szemerédi's regularity lemma, Lemma 6.6 from the previous section, in order to prove Theorem 6.1. To this end, we prove the following lemma, which is similar to previously known results for undirected graphs. See, for example, Theorem 2.1 in [90], and Lemma 3.2 in [6].

Lemma 6.10. *For every fixed ϵ and h, there is a positive constant $c(h,\epsilon)$ with the following property: for every fixed digraph H of size h, and for every digraph G of a large enough size n that is ϵ-far from being H-free, G contains at least $c(h,\epsilon)n^h$ copies of H.*

Proof: Let ϵ_1 be a constant whose value will be decided later. On inputs $1/\epsilon_1$ and ϵ_1, Lemma 6.6 returns an ϵ_1-regular partition with $|V_0| \leq \epsilon_1 n$ and partition sets $V_1,\ldots,V_t$, $|V_i| = k$ such that $1/\epsilon_1 \leq t \leq DM(1/\epsilon_1,\epsilon_1)$. Obtain from G the digraph G' by removing the following sets of edges:

- Edges that touch V_0. There are at most $(\epsilon_1 n)^2 + 2\epsilon_1 n^2 < 3\epsilon_1 n^2$ edges of this type.
- Edges within some set V_i. There are at most $t(n/t)^2 = n^2/t \leq \epsilon_1 n^2$ such edges.
- Edges between non ϵ_1-regular sets. There are at most $\epsilon_1 t^2 \cdot 2n^2/t^2 \leq 2\epsilon_1 n^2$ such edges.
- If for some pair of partition sets, one of the densities $\overrightarrow{d}\,, \overleftarrow{d}\,, \overline{d}$ is less than $\epsilon/4$, remove all corresponding edges (i.e. all edges that define that density). There are at most $\binom{t}{2}\epsilon n^2/t^2 \leq \epsilon n^2/2$ such edges.

Altogether we have removed less than $\epsilon n^2/2 + 6\epsilon_1 n^2$ edges from G. Thus, as G is ϵ-far from being H-free, for *any* $\epsilon_1 \leq \epsilon/13$ the digraph G' is obtained from G by removing less that ϵn^2 edges, and therefore still contains a copy of H. Moreover, for each directed edge (u,v) in H, u and v belong to an ϵ_1-regular pair $(U,V), u \in U, v \in V$, such that $\overrightarrow{d}\,(U,V) \geq \epsilon/4$. The same applies to a pair of edges $(u,v),(v,u)$ in H but this time with respect to the density $\overline{d}(U,V)$.

Having established the existence of one such H, we show that there are actually many more copies of H, provided that ϵ_1 is sufficiently small. Let $u_1,\ldots,u_h$ be the vertices of the copy of H in G, and assume that $u_i \in V_{\sigma(i)}$. We wish to show that for a small enough $\epsilon_1 \leq \epsilon/13$ we can build $c(h,\epsilon_1)n^h$ copies of H, where for each copy, u_i will belong to $V_{\sigma(i)}$. This would imply the lemma.

For our scheme to work we need to take $\epsilon_1 \leq \epsilon/13$ small enough that it satisfies,

$$(3h+1)\epsilon_1 \leq (\epsilon/4 - \epsilon_1)^h. \tag{6.2}$$

Note, that we must also take $\epsilon_1 \leq \epsilon/13$ so that we will be able to assume the properties of G' discussed above. Also, note that the value of ϵ_1 is a function of ϵ and h only, and is independent of n.

The idea is to build the copies iteratively, where in iteration $1 \leq i \leq h$, we find many candidates to play the role of u_i. To this end, we keep a set $C_{i,j} \subseteq V_{\sigma(i)}$, which includes

the vertices that may play the role of u_i after we have already found vertices for $u_1, \ldots, u_j$. Initially, $C_{i,0} = V_{\sigma(i)}$, $|C_{i,0}| = k$. Consider the stage when we come to select the vertices that will play the role of u_j. When we select a vertex to be u_j we have to update the sets $C_{i,j}$. For example, if for $i > j$ (u_j, u_i) is an edge of H, then after selecting v to be u_j we have to update $C_{i,j} = \overrightarrow{N}_{C_{i,j-1}}(v)$. The updates are equivalent for the other two cases where there is an edge (u_i, u_j) and when there are two edges $(u_i, u_j), (u_j, u_i)$.

The crucial observation now, is that we made sure that all edges of H go between ϵ_1-regular pairs, and moreover we have a relatively high density in the direction of these edges. Therefore, if $|C_{i,j-1}| \geq \epsilon_1 |V_{\sigma(i)}|$ then by Claim 6.5 all but at most $3\epsilon_1 |V_{\sigma(j)}|$ vertices in $V_{\sigma(j)}$ are such that the three inequalities of Claim 6.5 hold (with $d = \epsilon/4$ and $\epsilon = \epsilon_1$). That is,

$$|C_{i,j}| \geq (\epsilon/4 - \epsilon_1)|C_{i,j-1}|. \tag{6.3}$$

As H contains h vertices, and each $i > j$ excludes at most $3\epsilon_1 |V_{\sigma(j)}|$ from being u_j, then altogether we have at least $|C_{j,j-1}| - 3\epsilon_1 h |V_{\sigma(j)}|$ candidates for the role of u_j. For our scheme to work we must make sure that $|C_{i,j}| \geq \epsilon_1 |V_{\sigma(i)}|$ so that we may apply Lemma 6.5. But, by our previous assumptions the following holds for any $i > j$,

$$|C_{i,j}| - 3\epsilon_1 h |V_{\sigma(j)}| \geq (\epsilon/4 - \epsilon_1)^h k - 3\epsilon_1 hk \geq \epsilon_1 k.$$

The first inequality follows from (6.3) and the second from (6.2). We thus get that $|C_{i,j}| \geq \epsilon_1 k = \epsilon_1 |V_{\sigma(i)}|$ as needed. In particular $|C_{j,j-1}| \geq \epsilon_1 k$, thus we have $\epsilon_1 k$ choices when we come to choose u_j. Finally as Lemma 6.6 partitions V into a constant number of sets we get that,

$$k = \frac{n - |V_0|}{t} \geq \frac{n(1 - \epsilon_1)}{DM(1/\epsilon_1, \epsilon_1)}$$

Thus, for each iteration i, we have at least

$$\epsilon_1 k = \frac{\epsilon_1 (1 - \epsilon_1) n}{DM(1/\epsilon_1, \epsilon_1)}$$

choices for u_i. Therefore, as ϵ_1 is a function of ϵ and h only by (6.2), G' contains at least

$$\left(\frac{\epsilon_1 (1 - \epsilon_1)}{DM(1/\epsilon_1, \epsilon_1)} \right)^h n^h = c(h, \epsilon) n^h$$

copies of H. As G' is a subgraph of G, G contains at least as many copies. □

The proof of Theorem 6.1 now follows easily.

Proof of Theorem 6.1: The tester simply picks, say, $4/c(h, \epsilon)$ sets of vertices of G, where each set consists of h vertices, at random. If at least one of these sets spans a copy of H, it reports that G is not H-free, else, it declares that G is H-free. If G is H-free, then the algorithm will certainly report that this is the case. If G is ϵ-far from being H-free then, by the above lemma, the algorithm will find a copy of H with probability at least $2/3$. □.

6.4 Easily Testable Digraphs

In this section we prove parts (i) and (ii) of Theorem 6.2 as well as Theorem 6.3. We first show that the property of being H-free is easily testable, whenever the core of H is a 2-cycle. We then prove the same for all digraphs H for which the core of H is a tree. In Section 6.5 we show that for any other digraph H, the property of being H-free is hard to test.

We next prove that if the core of a digraph H is a 2-cycle, then testing H-freeness has query complexity polynomial in $1/\epsilon$. Observe, that the core of a digraph cannot be a bipartite digraph with at least one 2-cycle, and not be a 2-cycle, because there is a homomorphism from any such digraph to a 2-cycle.

Proof of Theorem 6.2, part (i): Let H be a bipartite digraph with at least one 2-cycle, with color classes of size s and t, and assume $s \leq t$. Our tester samples some c/ϵ^s vertices, for an appropriate $c = c(s,t)$, and reports that G is not H-free if and only if there is a copy of H spanned by a subset of these vertices. Clearly, if G is H-free, the algorithm will report this is the case. If G is ϵ-far from being H-free it must contain at least ϵn^2 cycles of length 2, as otherwise we can remove an edge from each of these 2-cycles and obtain an H-free digraph (using the fact that H contains a 2-cycle), while removing less than ϵn^2 edges. Now, consider the undirected graph G', obtained from G by putting an edge (u,v) in G' if and only if (u,v) is a 2-cycle in G. We show how to find in G' a set of vertices that spans a copy of $K_{s,t}$. From the definition of G', it implies that in G the same set spans a copy of H.

Randomly and independently, pick s vertices (with repetitions). The expected number of vertices that are connected to all the chosen vertices is

$$\sum_v \left(\frac{d_v}{n}\right)^s \geq n\left(\frac{\sum_v d_v}{n^2}\right)^s \geq n(2\epsilon)^s,$$

where d_v is the degree of v, the first inequality follows from convexity of the function x^s, and the second from our assumption that G' contains at least ϵn^2 edges.

It follows that with probability at least $\frac{1}{2}(2\epsilon)^s$, at least $\frac{1}{2}(2\epsilon)^s n$ vertices are adjacent to all the s chosen vertices, as otherwise the expectation would have been smaller than $n(2\epsilon)^s$. Therefore, after $10/(2\epsilon)^s$ rounds in which s vertices are chosen, with probability at least 15/16 at least $\frac{1}{2}(2\epsilon)^s n$ of the vertices are adjacent to all the s vertices chosen in one of the rounds. Fix these s vertices. If we now choose another vertex, it has probability at least $\frac{1}{2}(2\epsilon)^s$ of being adjacent to all these s vertices. We conclude that the expected number of additional vertices that we need to sample, in order to find t vertices that are connected to the s fixed ones, is at most $2t/(2\epsilon)^s$. By Markov's inequality, after sampling $8t/(2\epsilon)^s$ vertices, the probability of not finding a set of t vertices that is connected to all the s vertices is at most 1/4. The algorithm has probability at most 1/16 of failing to find the s vertices in the first step, a probability of at most 1/4 of failing to find the t vertices in the second step, and a probability of $o(1)$ that in each of the two steps, the chosen set does not consist of distinct vertices (notice that we sampled with repetitions). Altogether, the failure probability is at most 1/3, hence, the algorithm finds a copy of $K_{s,t}$ with probability

at least 2/3. As for the sample size, the first part uses a sample of size $10s/(2\epsilon)^s$, while the second is of size $8t/(2\epsilon)^s$. Altogether, we use a sample of size $O((1/\epsilon)^s) = O((1/\epsilon)^{h/2})$. This completes the proof of Theorem 6.1, part (i). □

Comment: By the above proof, every digraph G on sufficiently many vertices with $\Omega(n^2)$ 2-cycles, contains a copy of every fixed bipartite digraph. Therefore, there is a very simple and efficient **two-sided error** algorithm for testing $\mathcal{P}_H$, for every H whose core is a 2-cycle, which simply samples $O(1/\epsilon)$ pairs of vertices and accepts iff they span no edge.

We now proceed with the proof of Theorem 6.2 part (ii). In the proof we will use the following construction of a digraph G' obtained from a digraph G which is ϵ-far from being H-free. The process is described with respect to some tree K, which is a connected subgraph of H. We therefore denote $G' = G'(G, K)$. The reason to make the description general is that we will later use it with respect to different trees. Let G be a digraph that is ϵ-far from being H-free, and let K be some subtree of H. Let us also name the vertices of K as $1, \ldots, t$. We define the digraph $G' = G'(G, K)$ in the following constructive manner with respect to K: assign each vertex v of G a list $L(v)$ containing the numbers $1, \ldots, t$. This list should eventually contain $i \in \{1, 2, \ldots, t\}$ if and only if there is a homomorphism $\varphi : K \mapsto G'$ in which $\varphi(i) = v$. We also define $N^+(v, i)$ to be the set of vertices u, for which there is an edge (v, u), and $i \in L(u)$. We define $N^-(v, i)$ analogously only with respect to incoming edges into v. The process executes the following two operations while it can: (i) If for some directed edge (i, j) in K, there is a vertex v in G, for which $i \in L(v)$ and $|N^+(v, j)| < \frac{\epsilon}{2t}n$, remove all edges $\{(v, u) : u \in N^+(v, j)\}$, remove i from $L(v)$, and update all the sets $N^-(\cdot, i)$ of vertices in G. (ii) If for some directed edge (i, j) in K, there is a vertex v in G, for which $j \in L(v)$ and $|N^-(v, i)| < \frac{\epsilon}{2t}n$, remove all edges $\{(u, v) : u \in N^-(v, i)\}$, remove j from $L(v)$, and update all the sets $N^+(\cdot, j)$ of vertices in G.

Lemma 6.11. *If G is ϵ-far from being H-free, and K is a connected subgraph of H which is a tree, then the digraph $G' = G'(G, K)$ described above satisfies the following properties: (1) It contains a copy of K. (2) $i \in L(v)$ if and only if there is a homomorphism $\varphi : K \mapsto G'$ for which $\varphi(i) = v$.*

Proof: As K is a subgraph of H, and G is ϵ-far from being H-free, we may show that G' satisfies (1), simply by showing that the above process for obtaining G', does so by removing less than ϵn^2 edges. To this end, consider any vertex v. Each execution of items (i) and (ii) removes an element from $L(v)$, therefore we can execute them at most t times on v. As in each execution we remove less than $\frac{\epsilon}{2t}n$ edges, it follows that the process removes less than ϵn edges that touch v, and altogether less than ϵn^2 edges.

To prove (2) we first prove the implication that asserts that if $i \notin L(v)$ then there is no homomorphism $\varphi : K \mapsto G'$ for which $\varphi(i) = v$. We proceed by induction on m, the number of steps of the process. At the beginning, all the lists are full, therefore the desired property trivially holds. Assume it holds for m steps and consider step $m+1$: if we execute (i), then some i was removed from some $L(v)$, after removing all edges that go from v to vertices $N^+(v, j)$ for some j that is a neighbor of i in K. It follows from the induction hypothesis,

that no homomorphism can map j to an out-neighbor of v, and therefore, as i and j are neighbours in K, no homomorphism can map i to v. The case of executing (ii) is identical. To prove the second implication, assume that at the end of the process, for some vertex v, we have $i \in L(v)$ but there is no homomorphism $\varphi : K \mapsto G'$ for which $\varphi(i) = v$. Let K' be the largest connected subgraph of K that contains i, for which there is a homomorphism $\varphi : K' \mapsto G'$ that satisfies $\varphi(i) = v$ and for all $j \in K'$ $j \in L(\varphi(j))$. As K is connected, there is some vertex $i' \in K'$ that is connected by an edge to $j' \in K \setminus K'$ in K. By the maximality of K', there is no edge connecting $\varphi(i')$ to a vertex q for which $j' \in L(q)$. This is impossible, as it means that the process should have removed i' from $L(\varphi(i'))$. □

We now turn to the proof of Theorem 6.2, part (ii). The proof is based on a variant of a powerful probabilistic technique, which may be called *dependent random choice*, and which has already found several recent combinatorial applications. See, e.g., [9] and some of its references. Given a subset of vertices $V_i \subseteq V(G)$ and a vertex $v \in V(G)$, let $N(v, i)$ denote the set of neighbors of v within V_i. We need the following lemma.

Lemma 6.12. *Let $G = (V, E)$ be an undirected graph on n vertices, and let $V_1, V_2, \ldots, V_{d+1}$ be (not necessarily disjoint) subsets of V. Put $\alpha = |V_1|/n$. Assume that for every vertex $v \in V_1$ and for every $2 \leq k \leq d+1$, $|N(v,k)| \geq \epsilon |V_k|$. Then, sampling $32h \log(1/\delta)/(\alpha\epsilon^d)$ vertices from G, finds with probability at least $1-\delta$, an h-tuple of distinct vertices $s = \{v_1, \ldots, v_h\} \subseteq V_1$, that satisfies*

$$\left| \bigcap_{i=1}^{h} N(v_i, k) \right| \geq \frac{1}{4}\epsilon^{dh}|V_k|, \quad \forall\ 2 \leq k \leq d+1. \tag{6.4}$$

Proof: The result is trivial for $h = 1$, and we thus assume that $h \geq 2$. For $2 \leq k \leq d+1$, choose uniformly and independently a vertex t_k from each set V_k. Let X be the set of vertices $v \in V_1$, for which $t_k \in N(v, k)$ for all $2 \leq k \leq d+1$. For each $v \in V_1$, let X_v be an indicator random variable for the event that $v \in X$. It follows from the assumption on the large number of neighbours of each vertex of V_1 in each set V_k, that

$$E(|X|) = \sum_{v \in V_1} E(X_v) \geq \epsilon^d |V_1|.$$

By Jensen's inequality, it follows that

$$E(|X|^h) \geq E(|X|)^h \geq \epsilon^{dh}|V_1|^h.$$

Therefore, there is an expected number of at least $\epsilon^{dh}|V_1|^h$ h-tuples $s = (v_1, \ldots, v_h)$ (where the vertices v_i are not necessarily distinct) of vertices in V_1, with the property that $t_k \in N(v_i, k)$, for all $2 \leq k \leq d+1$ and $1 \leq i \leq h$. We now turn to show, that the expected number of these h-tuples that violate (6.4) is small. To this end, define Z to be the set of all h-tuples $s \in V_1^h$, that do not satisfy (6.4), and let Y be the set of all members of Z that

lie in X^h. For each $s \in Z$ let Y_s denote the indicator random variable for the event that $s \in X^h$. Note that $|Y| = \sum_{s \in Z} Y_s$. Thus

$$E(|Y|) = \sum_{s=(v_1,\ldots,v_h)\in Z} E(Y_s) = \sum_{s \in Z} \prod_{k=2}^{d+1} \frac{\left|\bigcap_{i=1}^{h} N(v_i,k)\right|}{|V_k|} \leq \sum_{s \in V_1^h} \frac{1}{4}\epsilon^{dh} \leq \frac{1}{4}\epsilon^{dh}|V_1|^h,$$

where the first inequality follows from our assumption that for some k, $\left|\bigcap_{i=1}^{h} N(v_i,k)\right| < \frac{1}{4}\epsilon^{dh}|V_k|$. We conclude that,

$$E(\frac{1}{2}|X|^h - |Y|) = \frac{1}{2}E(|X|^h) - E(|Y|) \geq \frac{1}{2}\epsilon^{dh}|V_1|^h - \frac{1}{4}\epsilon^{dh}|V_1|^h = \frac{1}{4}\epsilon^{dh}|V_1|^h.$$

Therefore, there is some choice of $t_2, \ldots, t_{d+1}$, for which the sets X and Y satisfy,

$$|X|^h - |Y| \geq \frac{1}{2}|X|^h + \frac{1}{4}\epsilon^{dh}|V_1|^h.$$

Fix one such choice of $t_2, \ldots, t_{d+1}$. The above inequality implies that more than half of the h-tuples in X^h satisfy (6.4), and that X is of size at least $\frac{1}{4^{1/h}}\epsilon^d|V_1| \geq \frac{\alpha}{2}\epsilon^d n$. Therefore, a randomly chosen vertex from G, has probability at least $\frac{\alpha}{2}\epsilon^d$ to lie in X. It follows that the expected number of samples needed to find an h-tuple from X is at most $2h/(\alpha\epsilon^d)$. Hence, by Markov's inequality, choosing $8h/(\alpha\epsilon^d)$ random vertices, finds an h-tuple from X with probability at least $\frac{3}{4}$. As at least half of the h-tuples in X^h satisfy (6.4), it follows that with probability at least $\frac{3}{8}$ we find an h-tuple satisfying (6.4). This is not necessarily an h-tuple of distinct vertices. But the probability of finding an h-tuple with non distinct vertices is $o(1)$, as $|X| = \Omega(n)$. Therefore with probability at least $\frac{1}{4}$ we find an h-tuple of distinct vertices satisfying (6.4). Thus, choosing $32h\log(1/\delta)/(\alpha\epsilon^d)$ vertices finds such an h-tuple with probability at least $1-\delta$ as needed. □

Proof of Theorem 6.2, part (ii): As in the proof of part (i), (and as can be done for any one-sided property tester for a problem which is closed under taking induced subgraphs), the algorithm simply samples the stated number of vertices randomly and reports that G is H-free if and only if it finds no copy of H on them. Clearly, if G is H-free, the answer is correct. Let G be ϵ-far from being H-free, and let K denote the core of H which is, by assumption, a tree. Number the vertices of K by $1, \ldots, k$ in a BFS order, and let h_i be the number of vertices of H that are mapped to $i \in \{1, 2, \ldots, k\}$. Note that if i and j are neighbors in K, it does not necessarily hold, that all the vertices of H that are mapped to i, are adjacent to all the vertices of H that are mapped to j, but it does hold, that all existing edges are in the same direction. We will show however, that we can find a subgraph of G whose vertex set consists of subsets $|U_1| = h_1, \ldots, |U_k| = h_k$ such that if $(i,j) \in E(K)$ then all the vertices of U_i are connected to all the vertices of U_j. Such a subgraph clearly contains a copy of H.

Let $N(i)$ be the neighbours of vertex i in K, that appear after it in the BFS order, and

$d_i = |N(i)|$. Apply the process described before the proof of Lemma 6.11 with respect to K, that is, obtain $G' = G'(G, K)$. It follows from Lemma 6.11 that G' contains a copy of K. Let $v_1, \ldots, v_k$ be such a copy. By Lemma 6.11, for all $1 \leq i \leq k$, $i \in L(v_i)$. Denote by V_i the set of vertices u_i for which $i \in L(u_i)$. Clearly $v_i \in V_i$. In order to make the presentation simple, from now until the end of the proof, we will not specify the direction of an edge between $u_i \in V_i$ and $u_j \in V_j$, although we will always be speaking about an edge that is directed as the direction of an edge between i and j in K.

Let $N(1) = \{2, \ldots, d_1 + 1\}$ be the d_1 neighbors of vertex 1 in K, hence, G' contains the edges $(v_1, v_2), \ldots, (v_1, v_{d_1+1})$. From the definition of the process for obtaining G', it follows that for every $2 \leq i \leq d_1 + 1$, there are at least $\frac{\epsilon}{2h}n$ vertices $u_1 \in V_1$, for which there is an edge (u_1, v_i) and $1 \in L(u_1)$, and in particular, $|V_1| \geq \frac{\epsilon}{2h}n$. It follows again from the definition of the process, that for every $u_1 \in V_1$, and for every $2 \leq i \leq d_1 + 1$, u_1 has at least $\frac{\epsilon}{2h}n$ neighbors in V_i, implying that $|V_i| \geq \frac{\epsilon}{2h}n$. As $|V_i| \leq n$, it follows that, *each* vertex in V_1 has at least $\frac{\epsilon}{2h}|V_i|$ neighbours in *each* V_i. We can continue this way to conclude that for $1 \leq i \leq k$, $|V_i| \geq \frac{\epsilon}{2h}n$, and that *every* $u_i \in V_i$ has at least $\frac{\epsilon}{2h}|V_j|$ neighbors in V_j, for *every* $j \in N(i)$. Finally note that as G' is a subgraph of G, all of the above applies also to G.

The previous paragraph implies, that we can apply Lemma 6.12 on the sets $V_1, \ldots, V_{d_1+1}$, with $\delta = \frac{1}{4h}$, $\alpha = \frac{\epsilon}{2h}$, $h = h_1$ and ϵ being $\epsilon/(2h)$, to conclude that sampling some $c_1(h)/(\epsilon^{d_1+1})$ vertices of G, finds, with probability at least $1 - \frac{1}{4h}$, an h_1-tuple s_1, of distinct vertices from V_1, such that for $2 \leq j \leq d_1 + 1$ they have at least $c'_1(h)\epsilon^{h_1 d_1}|V_j| \geq c"_1(h)\epsilon^{h_1 d_1 + 1}n$ common neighbors in V_j. The actual constants $c_1(h), c'_1(h), c''_1(h)$ as well as the constants that will appear at the rest of this proof can be derived from the statement of Lemma 6.12 and are omitted in order to keep the presentation simple. For $2 \leq j \leq d_1 + 1$, denote by V'_j this set of common neighbors of the vertices of s_1. Now each V'_j is of size at least $c"_1(h)\epsilon^{h_1 d_1 + 1}n$. By construction of G', every vertex in V_j, has at least $\frac{\epsilon}{2h}|V_t|$ neighbours in V_t, for every $t \in N(j)$. As $V'_j \subseteq V_j$, the same also applies to the vertices of V'_j. For $2 \leq j \leq d_1 + 1$, we can now apply Lemma 6.12 to V'_j as follows. Take $\delta = \frac{1}{4h}$, $\alpha = |V'_j|/n \geq c"_1(h)\epsilon^{h_1 d_1 + 1}$, $h = h_j$, $d = d_j$ and ϵ as before. We conclude that sampling $c_2(h)/(\epsilon^{d_j + d_1 h_1 + 1})$ finds, with probability at least $1 - \frac{1}{4h}$, an h_j-tuple s_j of distinct vertices from V'_j, with the property, that all the vertices of s_1 are adjacent to all the vertices of s_j, and the vertices of s_j have at least $c'_2(h)\epsilon^{d_j h_j}|V_t|$ common neighbors in V_t, for every $t \in N(j)$.

We now turn to generalizing the above for all $1 \leq i \leq k$, but before doing so we must take care of the following minor technicality; we must make sure that we do not sample the same vertex twice when we look for the copy of H, as it must consist of distinct vertices. We therefore remove from each V'_j the previously used vertices. As H is of fixed size, each V'_j is still of essentially its previous size.

Observe, that as each vertex in V_i has at least $\frac{\epsilon}{2h}|V_t|$ neighbours in V_t, for every required t, and we made sure that we do not sample the same vertex twice, we can safely generalize the above sampling technique as follows. For every $2 \leq i \leq k$, let p_i be the (single) neighbor of i in K that precedes it in the BFS order. Therefore, for every $2 \leq i \leq k$ we can sample some $c_3(h)/(\epsilon^{d_i + d_{p_i} h_{p_i} + 1})$ vertices, to find, with probability at least $1 - \frac{1}{4h}$,

an h_i-tuple s_i, with the properties, that every member in s_{p_i} is adjacent to every member of s_i, and the vertices of s_i have at least $c'_3(h)\epsilon^{d_i h_i}|V_t|$ common neighbors in V_t for every $t \in N(i)$. Observe, that as $k \leq h$, the probability that at least one of these k samples failed is at most $k/4h \leq 1/4$. Therefore, with probability at least $3/4$ we have found k sets $s_1, \ldots, s_k$ of sizes $h_1, \ldots, h_k$, respectively, such that for every edge (i,j) in K, we have all the edges going from s_i to s_j. This digraph clearly contains a copy of H, as needed. As for the total number of vertices sampled, note that we do not sample more than h times the size of the largest sample we use. The first sample, the one used to find s_1 is of size $c_1(h)/(\epsilon^{d_1+1}) = O((1/\epsilon)^{d_1+1})$. For $2 \leq i \leq k$, we use a sample of size $O((1/\epsilon)^{d_i + d_{p_i} h_{p_i} + 1})$. If we define $\overline{h} = \max_{2 \leq i \leq k}\{d_i + d_{p_i} h_{p_i} + 1\}$, then the total sample size is $O((1/\epsilon)^{\overline{h}})$. As it is clear that for every tree of size h, $\overline{h} \leq h^2$, we conclude that our ϵ-tester has indeed a query complexity of $O((1/\epsilon)^{h^2})$. $\square$

It is worth observing that in the proof of Theorem 6.2 part (ii), we did not explicitly use the fact that the core of the considered digraph H is a tree. Rather, we only needed the fact that $V(H)$ can be homomorphically mapped to some subgraph which is a tree. However, one can easily see that if such a homomorphism exists, then the core of H must also be a tree. We now turn to prove Theorem 6.3, that states that in case H is an oriented tree, we can design an optimal one-sided error ϵ-tester that simply samples a subset of $O(1/\epsilon)$ vertices, and checks if they span a copy of H.

Proof of Theorem 6.3: If G is H-free, the algorithm clearly reports it. Let G be ϵ-far from being H-free. Consider a DFS ordering of the vertices of H, and number the vertices of H accordingly $1, \ldots, h$. It follows that vertex i has exactly one neighbor from $1, \ldots, i-1$. Apply the process described before the proof of Lemma 6.11 with respect to H itself, that is, obtain $G' = G'(G, H)$. It follows from Lemma 6.11 that G' contains a copy of H. Let $v_1, \ldots, v_h$ be such a copy. By Lemma 6.11, for all $1 \leq i \leq h$, $i \in L(v_i)$. Without loss of generality, assume H contains the edge $(1,2)$. Therefore G' contains an edge (v_1, v_2), and by Lemma 6.11 $1 \in L(v_1)$ and $2 \in L(v_2)$. From the definition of the process for obtaining G', it follows that there are at least $\frac{\epsilon}{2h}n$ vertices u_1, for which there is an edge (u_1, v_2) and $1 \in L(u_1)$. It follows again from the definition of the process, that for each such u_1, there are at least $\frac{\epsilon}{2h}n$ vertices u_2 for which there is an edge (u_1, u_2) and $2 \in L(u_1)$. We can continue this way inductively to conclude that for every homomorphism mapping the subgraph of H spanned by the vertices $1, \ldots, i$ into G', there are at least $\frac{\epsilon}{2h}n$ possibilities for extending this homomorphism, to a homomorphism from the subgraph of H spanned by $1, \ldots, i+1$ into G'. As H is of fixed size, and n is assumed to be large enough, it follows that for each *injective* homomorphism mapping the subgraph of H spanned by the vertices $1, \ldots, i$ into G', there are at least $\frac{\epsilon}{2h}n - i \geq \frac{\epsilon}{3h}n$ possibilities for extending this injective homomorphism, to an injective homomorphism from the subgraph spanned by $1, \ldots, i+1$ into G'. Finally, observe that as G' is a subgraph of G, all the above applies also to G.

We now turn to the actual proof. We show that a random subset of $9h^2/\epsilon$ vertices, contains a copy of H with probability at least $2/3$. We choose this set one vertex at a time (with repetitions). From the above discussion, it follows that each randomly chosen vertex

v, has probability at least $\epsilon/3h$ of having the property that there is a copy of H in G in which v plays the role of vertex 1. More generally, it follows from the above discussion, that for every $1 \leq i \leq h-1$, if we have found vertices $v_1, \ldots, v_{i-1}$ having the property that there is a copy of H in G in which $v_1, \ldots, v_{i-1}$ play the role of vertices $1, \ldots, i-1$, then there are at least $\frac{\epsilon}{3h}n$ vertices u in G, such that there is a copy of H in G, in which $v_1, \ldots, v_{i-1}$ play the role of $1, \ldots, i-1$ respectively, and u plays the role of v_i. Therefore, each randomly chosen vertex has probability at least $\epsilon/3h$ of decreasing the number of vertices that are required in order to complete a copy of H, *regardless* of any history. By linearity of expectation, and the fact that the expected number of trials needed to find each new vertex is geometrically distributed, it follows that the expected number of trials needed to find a copy of H is $3h^2/\epsilon$. By Markov's inequality, it follows that the probability of not finding a copy of H after $9h^2/\epsilon$ trials, is at most $1/3$, as needed. Note, that the failure probability is in fact exponentially small in h/ϵ, but we do not need this stronger estimate here.

To show that the result is optimal, we show how to construct, for every tree H, a digraph G_H, that is ϵ-far from being H-free, yet in order to find a copy of H, one must sample $\Omega(1/\epsilon)$ vertices of G_H. Given a tree H of size h, construct a digraph G_H as follows: Let K be the core of H (which is obviously a tree), and let k denote its size. We also denote by t the number of vertices that are mapped to vertex k of K in a homomorphism from H to K. The digraph G_H contains $k-1$ sets of vertices $V_1, \ldots, V_{k-1}$ of size $\frac{n-\epsilon 2kn}{k-1}$ each, and one subset V_k of size $\epsilon 2kn$. For each edge (i,j) in K, G_H contains an edge (v_i, v_j) for every $v_i \in V_i$ and $v_j \in V_j$. To show that G_H is ϵ-far from being H-free, observe that there are

$$(\epsilon 2kn)^t \left(\frac{n-\epsilon 2kn}{k-1}\right)^{h-t}$$

natural homomorphisms from H into G_H, and at least half of them are injective (there are $o(n^h)$ homomorphisms that are not injective), that is, at least half of them define a copy of H. On the other hand, each edge e in G_H, is in the image of at most

$$(\epsilon 2kn)^{t-1} \left(\frac{n-\epsilon 2kn}{k-1}\right)^{h-t-1}$$

of these homomorphisms from H to K. Therefore, for a large enough n, one must remove at least

$$\frac{1}{2}(\epsilon 2kn)^t \left(\frac{n-\epsilon 2kn}{k-1}\right)^{h-t} \cdot (\epsilon 2kn)^{1-t} \left(\frac{n-\epsilon 2kn}{k-1}\right)^{1-h+t} \geq \epsilon kn \frac{n-\epsilon 2kn}{k-1} \geq \epsilon n^2$$

edges, in order to make G_H H-free, and hence G_H is ϵ-far from being H-free. The first multiplicand comes from the number of copies of H in G_H which is at least half of the number of homomorphisms from H to G_H, while the second comes from the number of copies of H which share a given edge. In order to establish that a digraph is not H-free, a one-sided error property-tester must find a copy of H. Now, by the minimality of K, each copy of H in G_H must have a vertex from V_k. Therefore, in order to find a copy of H

with probability 2/3, one must find a vertex in V_k with at least this probability. As proved by Goldreich and Trevisan in [77], we may assume without loss of generality that any one sided error property tester for $\mathcal{P}_H$ samples uniformly at random a subset of vertices, and answers by only inspecting edges spanned by this set. Finally, to find a vertex from V_k with probability at least 2/3, one must sample uniformly at random at least $\Omega(1/\epsilon)$ vertices. Thus, we obtain a lower bound of $\Omega(1/\epsilon)$ as required. □

6.5 Hard to Test Digraphs

In this section we apply the approach used in [1], together with some additional ideas, in order to prove Theorem 6.2 part (iii). This approach uses techniques from additive number theory, based on the construction of Behrend [29] of dense sets of integers with no three-term arithmetic progressions, together with some properties of homomorphisms of digraphs.

A linear equation with integer coefficients

$$\sum a_i x_i = 0 \tag{6.5}$$

in the unknowns x_i is *homogeneous* if $\sum a_i = 0$. If $X \subseteq M = \{1, 2, \ldots, m\}$, we say that X *has no non-trivial solution to* (6.5), if whenever $x_i \in X$ and $\sum a_i x_i = 0$, it follows that all x_i are equal. Thus, for example, X has no nontrivial solution to the equation $x_1 - 2x_2 + x_3 = 0$ if and only if it contains no three-term arithmetic progression. The following lemma is proved in [1] (Lemma 3.1), following the method of [29]:

Lemma 6.13. *For every fixed integer $r \geq 2$ and every positive integer m, there exists a subset $X \subset M = \{1, 2, \ldots, m\}$ of size at least*

$$|X| \geq \frac{m}{e^{10\sqrt{\log m \log r}}}$$

with no non-trivial solution to the equation

$$x_1 + x_2 + \ldots + x_r = r x_{r+1}. \tag{6.6}$$

Let $C = (v_1, \ldots, v_{r+1}, v_1)$ be an arbitrary *oriented* cycle of length $r + 1$. We next apply the construction in the above lemma to construct, for every integer $r + 1 \geq 3$, a relatively dense digraph consisting of pairwise edge disjoint copies of C, which does not contain too many copies of C of a special structure (see statement of lemma below). Let m be an integer, let $X \subset \{1, 2, \ldots m\}$ be a set satisfying the assertion of Lemma 6.13, and define, for each $1 \leq i \leq r + 1$, the set V_i to consist of the vertices $\{1, 2, \ldots im\}$ where, with a slight abuse of notation, we think on the sets $V_1, \ldots, V_{r+1}$ as being pairwise disjoint. The reason we use this notation is that we will next refer to the vertices of these sets as integers. In order to avoid confusion, when we will later on refer to a vertex we will always state to which of the sets $V_1, \ldots, V_{r+1}$ it belongs.

Let $T = T(X, C)$ be the family of all $r + 1$-partite digraphs on the classes of vertices $V_1, V_2, \ldots, V_{r+1}$, whose edges are defined as follows: For each j, $1 \leq j \leq m$, and for each

$x \in X$ the vertices $j \in V_1, j+x \in V_2, j+2x \in V_3, \ldots, j+rx \in V_{r+1}$ form an oriented cycle of length $r+1$ in this order, whose edges are directed as the edges of C. Therefore, if C contains the directed edge (v_i, v_{i+1}), then $(j+(i-1)x, j+ix)$ is an edge from V_i to V_{i+1} for all $1 \leq j \leq m, x \in X$, in any member of T. If C contains the reverse edge (v_{i+1}, v_i), then $(j+ix, j+(i-1)x)$ is an edge from V_{i+1} to V_i for all $1 \leq j \leq m, x \in X$ in any member of T. The same applies to the edges between V_1 and V_{r+1}. If (v_i, v_{i+1}) is an edges in C, then any digraph in T does not contain any additional edges going from V_i to V_{i+1}. If (v_{i+1}, v_i) is an edge in C, then any digraph in T does not contain any additional edges going from V_{i+1} to V_i. The same applies to V_1, V_{r+1}. Besides the above set of edges and restrictions, the members of T may contain any other edges between V_i, V_j.

Lemma 6.14. *For every integer $r \geq 2$, and every m, any member of $T(X, C)$ defined above has precisely $m|X|$ ($< m^2$) copies of the cycle C, such that the vertex that plays the role of v_i in the copy of C, belongs to V_i.*

Proof: We only have to show that any member of T does not contain any additional copies of C, for which the vertex that plays the role of v_i in the copy of C, belongs to V_i. Let C' be such a copy of C. Therefore, there are $j \leq m$ and elements $x_1, x_2, \ldots, x_{r+1} \in X$, such that the vertices of the cycle are $j \in V_1, j+x_1 \in V_2, j+x_1+x_2 \in V_3, \ldots, j+x_1+x_2+\ldots+x_r \in V_{r+1}$ and $x_1 + x_2 + \ldots + x_r = rx_{r+1}$ (remember that all edges between V_1 and V_{r+1} are of the form $(j, j+rx)$ or $(j+rx, j)$). However, by the definition of X this implies that $x_1 = x_2 = \ldots = x_{r+1}$, implying the desired result. $\Box$

Comment: Note that the members of $T(X, C)$ may contain many additional copies of C, which do not satisfy the restriction described in the statement of the lemma.

An s-blow-up of a digraph $K = (V(K), E(K))$ is the digraph obtained from K by replacing each vertex of K by an independent set of size s, and each edge e of K by a complete bipartite directed subgraph whose vertex classes are the independent sets corresponding to the ends of the edge, and whose edges are directed according to the direction of e.

Lemma 6.15. *Let $H = (V(H), E(H))$ be a digraph with h vertices, let $K = (V(K), E(K))$ be another digraph on at most h vertices, and let $T = (V(T), E(T))$ be an s-blow-up of K. Suppose there is a homomorphism*

$$\varphi : V(H) \mapsto V(K)$$

from H to K and suppose $s \geq h$. Let $R \subset E(T)$ be a subset of the set of edges of T, and suppose that each copy of H in T contains at least one edge of R. Then

$$|R| \geq \frac{|E(T)|}{|E(K)||E(H)|} > \frac{|E(T)|}{h^4}.$$

Proof: Let $g : V(H) \mapsto V(T)$ be a random injective mapping obtained by defining, for each vertex $v \in V(K)$, the images of the vertices in $\varphi^{-1}(v) \in V(H)$ randomly, in a one-to-one

fashion, among all s vertices of T in the independent set that corresponds to the vertex v. Obviously, g maps adjacent vertices of H into adjacent vertices of T, and hence the image of g contains a copy of H in T. Each edge of H is mapped to one of the corresponding s^2 edges of T according to a uniform distribution, and hence the probability it is mapped onto a member of R does not exceed $|R|/s^2$. It follows that the expected number of edges of H mapped to members of R is at most $\frac{|R||E(H)|}{s^2}$, and as, by assumption, this random variable is always at least 1, we conclude that $\frac{|R||E(H)|}{s^2} \geq 1$. The desired result follows, since $s^2 = |E(T)|/|E(K)|$. $\square$

Claim 6.16. *If K, the core of H, is neither a tree nor a 2-cycle, then K contains an oriented cycle C of length at least 3. Moreover, any homomorphism from H to K, maps a copy of C from H to the copy of C in K.*

Proof: Let k denote the number of vertices of K, and let us number its vertices $\{v_1, v_2, \ldots, v_k\}$ such that the first $r+1 \geq 3$ vertices $v_1, v_2, \ldots v_{r+1}$ form an oriented cycle C in this order. One such cycle must exist as K is by assumption neither a tree nor a 2-cycle. Remember, that as was explained in the discussion before the proof of Theorem 6.2, part (i), the core cannot have only 2-cycles, and not be a 2-cycle. By the minimality of K, every homomorphism φ of K into itself must be an automorphism, that is $(u, v) \in E(K) \Leftrightarrow (\varphi(u), \varphi(v)) \in E(K)$ (otherwise H would have a homomorphism into a subgraph with a smaller number of edges). We claim that *any* homomorphism of H into K maps a copy of C from H to the vertices $v_1, v_2, \ldots v_{r+1}$ of K. Indeed, any homomorphism of H into K, induces also a homomorphism of K into K. Therefore, some $r+1$ vertices of K are mapped to $v_1, v_2, \ldots, v_{r+1}$, and these vertices must span a cycle in K and therefore in H, as this homomorphism is an automorphism from K to K by the previous argument. $\square$

Lemma 6.17. *For every fixed digraph $H = (V(H), E(H))$ on h vertices whose core is neither an oriented tree nor a 2-cycle, there is a constant $c = c(H) > 0$, such that for every positive $\epsilon < \epsilon_0(H)$ and every integer $n > n_0(\epsilon)$, there is a digraph G on n vertices which is ϵ-far from being H-free, and yet contains at most $\epsilon^{c \log(1/\epsilon)} n^h$ copies of H.*

Proof: Let K be the core of H, and let k denote the number of vertices of K. Also, let us number its vertices $\{v_1, v_2, \ldots, v_k\}$ such that the first $r+1 \geq 3$ vertices $v_1, v_2, \ldots v_{r+1}$ form an oriented cycle C in this order as guaranteed by Claim 6.16. Given a small $\epsilon > 0$, let m be the largest integer satisfying

$$\epsilon \leq \frac{1}{h^8 e^{10\sqrt{\log m \log h}}}. \tag{6.7}$$

It is easy to check that this m satisfies

$$m \geq \left(\frac{1}{\epsilon}\right)^{c \log(1/\epsilon)} \tag{6.8}$$

for an appropriate $c = c(h) > 0$. Let $X \subset \{1, 2, \ldots, m\}$ be as in Lemma 6.13. We next define a digraph F from K in a way similar to the one described in the paragraph preceding

Lemma 6.14. Let $V_1, V_2, \ldots V_k$ be pairwise disjoint sets of vertices, where $|V_i| = im$ and we denote the vertices of V_i by $\{1, 2, \ldots, im\}$. For each j, $1 \leq j \leq m$, for each $x \in X$ and for each directed edge (v_p, v_q) of K, let $j + (p-1)x \in V_p$ have an outgoing edge pointed to $j + (q-1)x \in V_q$. In other words, F consists of $m|X|$ copies of K, where the vertices of each copy form an arithmetic progression whose first element is j and whose difference is x. It follows that each pair of these copies shares at most one vertex in F. In particular, these copies are edge disjoint. It thus follows that the number of edges in F satisfies

$$|E(F)| = m|X||E(K)|.$$

Note that the induced subgraph of F on the union of the first $(r+1)$ vertex classes, belongs to the family of digraphs $T(X, C)$ considered in Lemma 6.14, where $C = (v_1, \ldots, v_{r+1}, v_1)$ is the oriented cycle on the first $r+1$ vertices of K, which was defined above. Finally, define

$$s = \left\lfloor \frac{n}{|V(F)|} \right\rfloor = \left\lfloor \frac{2n}{k(k+1)m} \right\rfloor$$

and let G be the s-blow-up of F (together with some isolated vertices, if needed, to make sure that the number of vertices is precisely n). Note that the number of edges of G satisfies,

$$|E(G)| = \frac{4n^2|E(F)|}{k^2(k+1)^2m^2} = \frac{4n^2|X||E(K)|}{k^2(k+1)^2m} \geq \frac{n^2|X||E(K)|}{k^4m} \geq \frac{n^2|E(K)|}{k^4 e^{10\sqrt{\log m \log r}}} \tag{6.9}$$

where the last inequality follows from the lower bound on $|X|$ that is guaranteed by Lemma 6.13.

Since F consists of $m|X|$ edge disjoint copies of K, G consists of pairwise edge disjoint s-blow-ups of K, hence, by Lemma 6.15, one has to delete at least a fraction of $1/h^4$ of its edges to destroy all copies of H in it. Therefore, one must delete at least

$$\frac{1}{h^4} \cdot |E(G)| \geq \frac{n^2|E(K)|}{h^4k^4e^{10\sqrt{\log m \log r}}} \geq \frac{n^2|E(K)|}{h^8e^{10\sqrt{\log m \log h}}} \geq \epsilon n^2 \tag{6.10}$$

edges in order to destroy all copies of H. The first inequality follows from (6.9), the second from the fact that $r \leq h$ and $k \leq h$ and the third from (6.7). We conclude that G is ϵ-far from being H-free.

We next claim that any copy of H in G must contain a copy of C such that for $1 \leq i \leq r+1$, the vertex that plays the role of v_i belongs to the blow-up of the vertices of V_i. To see this, note that there is a natural homomorphism of G onto K, obtained by first mapping G homomorphically onto F (by mapping each class of s vertices into the vertex of F to which it corresponds), and then by mapping all vertices of V_i to v_i. This homomorphism maps each copy of H in G homomorphically into K, and hence, by Claim 6.16, maps a copy of C that belongs to the considered digraph H, to the first $r+1$ vertices of K. The definition of the homomorphism thus implies the assertion of the claim.

As the vertex that plays the role of v_i in the copy of C must belong to the blow-up of the vertices of V_i for $1 \leq i \leq r+1$, it follows from Lemma 6.14 that the number of such

cycles is at most

$$m^2 s^{r+1} = m^2 \left(\frac{2n}{k(k+1)m} \right)^{r+1} \leq n^{r+1}/m,$$

and this implies that the total number of copies of H in G does not exceed $n^h/m = \epsilon^{c \log(1/\epsilon)} n^h$, implying the desired result. □

Proof of Theorem 6.2, part (iii): Let H be a digraph on h vertices whose core is neither an oriented tree nor a 2-cycle, and suppose $\epsilon > 0$. Given a one-sided error ϵ-tester for testing H-freeness we may assume, without loss of generality, that it queries all pairs of a uniformly at random chosen set of vertices (otherwise, as explained in [6], every time the algorithm queries about a vertex pair we make it query also about all pairs containing a vertex of the new pair and a vertex from previous queries. See also [77] for a more detailed proof of this statement.) As the algorithm is a one-sided-error algorithm, it can report that G is not H-free only if it finds a copy of H in it. By Lemma 6.17 there is a digraph G on n vertices which is ϵ-far from being H-free and yet contains at most $\epsilon^{c \log(1/\epsilon)} n^h$ copies of H. The expected number of copies of H inside a uniformly at random chosen set of x vertices in such a digraph is at most $x^h \epsilon^{c \log(1/\epsilon)}$, which is far smaller than 1 unless x exceeds $(1/\epsilon)^{c' \log(1/\epsilon)}$ for some $c' = c'(H) > 0$, implying the desired result. □

6.6 Two-Sided Error Testers

In this section we present the proof of Theorem 6.4. Applying the second part of the theorem for the case of undirected graphs, shows that if H is an undirected, non-bipartite graph, then there is no two-sided ϵ-tester for testing H-freeness whose query complexity is smaller than $(1/\epsilon)^{c \log 1/\epsilon}$ for an appropriate $c = c(H) > 0$. This settles an open problem raised in [1]. For the proof we need the following easy application of a theorem of Erdős from [52].

Lemma 6.18. *Let H be a fixed digraph on h vertices, let K be its core, and denote by k the size of K. For every constant $0 < \gamma < 1$ and for every sufficiently large n, every digraph G on n vertices that contains γn^k copies of K, contains also a copy of H.*

Proof: Let φ be a homomorphism from $V(H)$ to $V(K)$, denote by $t_1, \ldots, t_k$ the vertices of K, and let $S_1, \ldots, S_k$ be the sets $\varphi^{-1}(t_1), \ldots, \varphi^{-1}(t_k)$, respectively. Define a k-uniform hypergraph T as follows: take a random partition of $V(G)$ into k subsets, $V_1, \ldots, V_k$, where each vertex of G is chosen uniformly and independently to be in one of the groups. For each copy of K in G, in which the vertices $u_{i_1}, \ldots, u_{i_k}$ play the role of $t_1, \ldots, t_k$ respectively, put an edge in T that contains $u_{i_1}, \ldots, u_{i_k}$ if and only if $u_{i_1} \in V_1, \ldots, u_{i_k} \in V_k$. Observe, that by linearity of expectation, if G contains γn^k copies of K, the expected number of edges in T is $\gamma k^{-k} n^k$. Therefore, one partition which defines at least this many edges must exist. Fix one such partition, and the hypergraph T' which it defines. In [52] it is proved that any k-uniform hypergraph on n vertices with at least $n^{k-h^{1-k}}$ edges, contains a copy of a complete k-partite k-uniform hypergraph, where each partition class is of size h. It follows that for large enough n, T' contains a copy of such hypergraph on some hk vertices

$\{v_1^1, \ldots, v_h^1\} \subseteq V_1, \ldots, \{v_1^k, \ldots, v_h^k\} \subseteq V_k$. It is now easy to see that G must contain a copy of H where for the role of the vertices of S_i we can choose any $|S_i|$ vertices from $\{v_1^i, \ldots, v_h^i\}$. □

Proof of Theorem 6.4, part (i): Let H be a fixed digraph with core K, and let k be the size of K. If K is a 2-cycle, then a two-sided error ϵ-tester for testing $\mathcal{P}_H$ with query complexity $O(1/\epsilon)$ was described in the comment following the proof of Theorem 6.2 part (i). Assume now that K is an oriented tree. Our two-sided error ϵ-tester for $\mathcal{P}_H$ works as follows: Given a digraph G, the algorithm samples c/ϵ vertices, for an appropriate c, and reports that the digraph is not H-free if and only if they span a copy of K. We turn to show that the algorithm answers correctly with probability at least 2/3. Assume G is ϵ-far from being H-free. Then it is clearly also ϵ-far from being K-free, therefore applying Theorem 6.3 to $\mathcal{P}_K$, we conclude that a randomly chosen set of c/ϵ vertices, with an appropriate c, finds a copy of K with probability at least 2/3. Assume G does not contain a copy of H. It follows from Lemma 6.18 that it contains $o(n^k)$ copies of K, and therefore a randomly chosen set of any constant size (independent of n), and in particular of size $O(1/\epsilon)$, has probability $o(1)$ of finding a copy of K.

To show that the result is optimal, we apply Yao's principle [115]. We first prove the case of K being an oriented tree. Applying Yao's principle to our setting, we first have to define for every n, two distributions of digraphs D_1, D_2, where all the digraphs in D_1 are ϵ-far from being H-free, and all the digraphs in D_2 are H-free. In order to define the two distributions we use the digraph G_H whose description appears at the end of the proof of Theorem 6.3. Note that this digraph is constructed using the core K, which is a tree. D_1 is a uniform distribution on all the $n!$ digraphs that are obtained from G_H by a permutation of its vertices. By the computation at the end of the proof of Theorem 6.3 it follows that all the digraphs in D_1 are ϵ-far from being H-free. To define D_2 we first define G'_H to be the digraph that is obtained from G_H by removing all the edges that touch V_k (see the definition of G_H). D_2 is now a uniform distribution on all the $n!$ digraphs that are obtained from G'_H by a permutation of its vertices. As G'_H is clearly H-free, all the digraphs in D_2 are H-free. To finish the proof we must show that no deterministic algorithm that samples less than $\Omega(1/\epsilon)$ vertices (adaptively) can tell the difference between these two distributions with probability that exceeds, say, 1/3. Recall that by the definition of G_H and G'_H, as long as the algorithm does not look at a vertex from V_k, it sees the *same* digraph. As V_k is of size $\epsilon 2kn$, the probability that a deterministic algorithm that samples less than, say, $1/(10\epsilon k)$ vertices finds a vertex from V_k is smaller than 1/3. Therefore, with probability at least 2/3 the two distributions D_1, D_2 will look identical to any deterministic algorithm sampling less than $\Omega(1/\epsilon)$ vertices, as needed.

The proof for the case of K being a 2-cycle is analogous, and involves taking a permutation of a complete bi-directed bipartite graph on vertex sets of sizes $\epsilon 4n$ and $n - \epsilon 4n$, and a digraph with no edges. The rest of the details are left to the reader. □

A close inspection at the proofs of Theorem 6.3 and Theorem 6.2 part (i), shows that if G is ϵ-far from being H-free, and the core of H, K, is either a 2-cycle or an oriented tree,

then sampling $O(1/\epsilon)$ vertices finds a copy of K with probability $1 - o(1)$ where the $o(1)$ term tends to 0 as ϵ tends to zero. On the other, the proof of Theorem 6.4, part (i), shows that if G is H-free, then the algorithm does not find a copy of K with probability $1 - o(1)$ where the $o(1)$ term tends to 0 as n tends to infinity (even if $\epsilon > 0$ is relatively large). Therefore, in some sense the test has "almost" one-sided error, as even for large values of ϵ the failure probability in case G is H-free is still $o(1)$, as n tends to infinity.

Proof of Theorem 6.4, part (ii): Let H be a fixed digraph whose core K is neither a directed 2-cycle nor an oriented tree. We apply Yao's principle again in order to prove the lower bound.

Given n and ϵ, let X, m and the sets V_i be as in the proof of Lemma 6.17. Construct the digraph F just as in the proof of Lemma 6.17, and remember that it consists of $m|X|$ pairwise edge disjoint copies of K (though it may well contain additional copies of K). Recall, also, that K contains a cycle C of length $r + 1 \geq 3$, and that each copy of K in F contains a copy of this cycle in which the i-th vertex lies in V_i for all $1 \leq i \leq r + 1$. Let $\mathcal{C}$ denote the set of these edge disjoint copies of C, and note that by Lemma 6.14 there are no other copies of C in F, in which the i-th vertex lies in V_i, besides the $m|X|$ members of $\mathcal{C}$.

To construct D_1 which consists of digraphs that are ϵ-far from being H-free with probability $1 - o(1)$, we first construct F_1' by removing each of the $m|X|$ edge disjoint cycles that belong to $\mathcal{C}$ with probability $\frac{1}{r+1}$. We then create G_1 by taking an s blow up of F_1' adding isolated vertices, if needed. Finally, D_1 consists of all randomly permuted copies of such digraphs G_1. It follows from a standard Chernoff bound, that with probability $1 - o(1)$, at least $m|X|(1 - 2/(r + 1))$ copies of C are left in F_1', where the $o(1)$ term tends to 0, as ϵ tends to 0. Similar to the derivation of (6.10), it is easy to show that if $m|X|/2(r + 1)$ of these copies of C are left in F_1', the digraph G_1 is ϵ-far from being H-free. It follows that with probability $1-o(1)$, a member of D_1 is ϵ-far from being H-free. The distribution D_2 of digraphs that are H-free, is defined by first constructing F_2' by removing from each member $C \in \mathcal{C}$ one randomly chosen edge (out of the $r + 1$ edges of the cycle). We then create G_2 by taking an s blow up of F_2' adding isolated vertices, if needed. Finally, D_2 consists of all randomly permuted copies of such digraphs G_2, which are clearly H-free.

Now consider a set of vertices S in G_1 (or G_2) and its natural projection to a subset of $V(F)$, which we also denote by S with a slight abuse of notation. Suppose S has the property that it does not contain more than two vertices from any one of the copies of C that belong to $\mathcal{C}$.

If this property holds, then each edge spanned by S is contained in a different copy of $C \in \mathcal{C}$ (if it is contained in such a cycle at all). Therefore, each edge that lies in such a cycle, has probability $1 - \frac{1}{r+1}$ of being in F_1', and these probabilities are mutually independent. Similarly, each such edge has probability $1 - \frac{1}{r+1}$ of being in F_2' and these probabilities are also mutually independent. It follows that sampling a digraph G from D_1, and looking at the induced digraph on a set S with the above property, has *exactly* the same distribution as sampling a digraph G from D_2, and looking at the induced digraph on S.

To complete the proof we have to show that no deterministic algorithm can distinguish between the distributions D_1 and D_2 with constant probability. To this end, it is clearly

enough to show that any deterministic algorithm that looks at a digraph spanned by less than $(1/\epsilon)^{c' \log 1/\epsilon}$ vertices, has essentially the same probability of seeing any digraph regardless of the distribution from which the digraph was chosen. By the discussion in the previous paragraph, this can be proved by establishing that, with high probability, a small set of vertices does not contain three vertices from the same copy of C. For a fixed ordered set of three vertices in S, consider the event that they all belong to the same copy of C. The first two vertices determine all the vertices of one of these copies uniquely. Now, the conditional probability that the third vertex is also a vertex of the same copy is $(r+1)/|V(F)| \leq r/m$. By the union bound, the probability that the required property is violated is at most

$$r|S|^3/m \leq r|S|^3 \epsilon^{c \log 1/\epsilon}.$$

This quantity is $o(1)$ as long as $|S| = o((1/\epsilon)^{\frac{c}{3} \log 1/\epsilon})$, where here we applied the lower bound on the size of m given in (6.8). Therefore, if the algorithm has query complexity $o((1/\epsilon)^{c' \log 1/\epsilon})$ for some absolute positive constant c', it has probability $1 - o(1)$ of looking at a subset on which the distributions D_1 and D_2 are identical, thus, the probability that it distinguishes between D_1 and D_2 is $o(1)$. $\square$

A slightly more complicated argument than the above can give two distributions D_1 and D_2, such that the digraphs in D_1 are *always* ϵ-far from being H-free, while the digraphs in D_2 are always H-free. The idea is to first partition the $m|X|$ copies of C into pairs, assuming for simplicity that $m|X|$ is even. To create D_1, we randomly pick from each pair of copies of C a single copy, and delete two randomly chosen edges from this copy. To create D_2, we do exactly the same as we did in the proof above. It is easy to appropriately modify the proof above in order to show that any deterministic algorithm with query complexity $o((1/\epsilon)^{c \log 1/\epsilon})$ cannot distinguish between D_1 and D_2 (see [11] for more details). As this argument has no qualitative advantage, we described the simpler one given above.

Observe that for digraphs H whose core K is neither an oriented tree nor a 2-cycle, we can give the above lower bound for testing $\mathcal{P}_H$, but no better upper bound than the one given by Theorem 6.1. However, following the arguments in the proof of Theorem 6.4 (i), it follows that the query complexity of testing $\mathcal{P}_H$ with two-sided error is at most the query complexity of testing $\mathcal{P}_K$ with two-sided error. Thus, for example, the query complexity of testing the digraph in Figure 6.1 (c) with two-sided error, is at most the query complexity of testing its induced oriented triangle with two-sided error.

6.7 Concluding Remarks and Open Problems

- We have shown that for any digraph H, the property $\mathcal{P}_H$ of being H-free is testable with one-sided error. In order to prove this result we have first proved a regularity lemma for digraphs, which generalizes Szemerédi's regularity lemma for undirected graphs. This lemma might prove useful for tackling other problems as well. We also gave a precise characterization of all digraphs H for which $\mathcal{P}_H$ is easily testable, and showed that the same characterization applies to two-sided error ϵ-testers as well,

where here the complexity is polynomial in $1/\epsilon$ if and only if it is $\Theta(1/\epsilon)$. We have addressed the case when H is an oriented tree, and gave an optimal one-sided error ϵ-tester with query complexity $\Theta(1/\epsilon)$ for this case.

- It is not difficult to generalize Theorem 6.2 to the case of disconnected digraphs. Let H be a disconnected graph whose components we denote by $H_1, \ldots, H_t$, and whose cores we denote by $K_1, \ldots, K_t$. Note that if G is ϵ-far from being H-free, then for all i, it is also ϵ-far from being H_i-free. If $K_1, \ldots, K_t$ are all either trees or 2-cycles, then running the testers for $H_1, \ldots, H_t$ will find disconnected copies of each of $H_1, \ldots, H_t$, and therefore a copy of H. This test will obviously have query complexity polynomial in $1/\epsilon$, and therefore in this case $\mathcal{P}_H$ is easily testable. If at least one of the cores is neither a tree nor a 2-cycle then the core of H is neither a tree nor a 2-cycle, hence, it follows directly from the proof of Theorem 6.2 part (iii) (note that Lemma 6.17 and the proof of Theorem 6.2 part (iii) do not assume that H is connected) that $\mathcal{P}_H$ is not easily testable. Note finally that the above applies also to the case of two-sided error, thus Theorem 6.4 can also be extended to the case of disconnected digraphs.

- Hell, Nesetril and Zhu proved in [85] that the problem of deciding if the core of a given input digraph is a tree is NP-complete. This, together with Theorem 6.2 imply the following.

  Proposition 6.19. *The problem of deciding whether for a given digraph H, the property $\mathcal{P}_H$ is easily testable, is NP-complete.*

  Therefore, there is no polynomially testable characterization of the digraphs H for which $\mathcal{P}_H$ is easily testable (though for every small, fixed H, Theorem 6.2 can be easily used to decide if H is such a digraph). One interesting class of digraphs for which the problem is solvable in polynomial time, is the class of oriented cycles. An oriented cycle is *balanced* if the number of forward edges is equal to the number of backward edges. It is not difficult to see that if an oriented cycle C is not balanced, then the core of C is C itself, (see, e.g., Figure 6.1 (b)). However the converse is not true, and while there are balanced cycles whose core is a path, (see, e.g., Figure 6.1 (a)), there are also balanced cycles C whose core is C itself, (see, e.g., Figure 6.1 (d)). It is therefore interesting to observe that the problem of deciding whether the core of a given cycle C is C itself or an induced path in it, can be solved in polynomial time using dynamic programming. The details are left to the reader.

  A digraph H is balanced iff every oriented cycle in it is balanced. It is not difficult to see that a digraph H is balanced iff there is a homomorphism mapping H into an oriented tree, and this happens iff there is a homomorphism mapping H into a directed path. It thus follows, by Theorem 6.2, that if H is not balanced then $\mathcal{P}_H$ cannot be tested by a polynomial number of queries (but the converse is not true in general.)

- Lemma 6.11 implies that if G is ϵ-far from satisfying $\mathcal{P}_H$, and the core of H is a tree K of size k, then G contains $\Omega(\epsilon^k n^k)$ copies of K. Having this, we could have

used results from the theory of supersaturated graphs and hypergraphs (see [56]) to conclude that there exists a one-sided error ϵ-tester for $\mathcal{P}_H$ which uses a sample of size $O((1/\epsilon)^{O(h^k)})$. (An alternative way to deduce this, is to change the statement of Lemma 6.18 and prove that G contains $c(\gamma)n^h$ copies of H for some constant $c(\gamma)$, and not just one). However, our proof of Theorem 6.2 part (ii) given here provides a far more efficient ϵ-tester that uses a sample of size only $O((1/\epsilon)^{h^2})$. By applying the techniques of [56] we can show that for every fixed digraph H with h vertices whose core K (which is not necessarily a tree) has k vertices, any digraph on n vertices containing at least δn^k copies of the core K, contains at least $\Omega(\delta^{O(h^k)}n^h)$ copies of H.

- Lemma 6.11 implies that if G is ϵ-far from satisfying $\mathcal{P}_H$, and H is a tree of size h, then G contains $\Omega(\epsilon^h n^h)$ copies of H. This can be seen to be essentially optimal by considering an appropriate random digraph. We omit the details.

  As there are many copies of H, we conclude that sampling h vertices finds a copy of H with probability $\Omega(\epsilon^h)$. It follows that one can test $\mathcal{P}_H$ simply by sampling $\Theta((1/\epsilon)^h)$ samples of h vertices each. However, in Theorem 6.3 we show that a sample of size $O(1/\epsilon)$ suffices. The reason is that sampling h vertices in $O((1/\epsilon)^h)$ rounds fails to take into account all the h-tuples that lie in the sample. In a sample of size $\Theta(1/\epsilon)$ there are $\Theta((1/\epsilon)^h)$ subsets of size h, and it turns out that if we consider all of them, we get essentially the same result as sampling $\Theta((1/\epsilon)^h)$ subsets of size h. In general, showing that if G is ϵ-far from being H-free then it contains $f(\epsilon)n^h$ copies of H, and then designing a ϵ-tester that samples $1/f(\epsilon)$ subsets of size h, usually fails to achieve the query complexity of more efficient ϵ-testers. In many cases, the difference can be substantial, as in our case. In addition, our proof of a test that uses a sample of size $O(1/\epsilon)$ gives a somewhat different proof that for any oriented tree H with h vertices, a digraph that is ϵ-far from being H-free, contains $\Omega(\epsilon^h n^h)$ copies of H.

- Testing H-freeness for H being the complete bipartite undirected graph $K_{s,t}$, is another example of the above mentioned phenomenon. In [1], an ϵ-tester for $K_{s,t}$-freeness which uses a sample of size $O((1/\epsilon)^{st})$ has been established, simply by showing that the graph must contain $\Omega(\epsilon^{st}n^{s+t})$ copies of $K_{s,t}$. Our method here improves this result and shows that a sample of size $O((1/\epsilon)^{\min(s,t)})$ suffices. This nearly matches a lower bound of $\Omega((1/\epsilon)^{\min(s,t)/2})$ which follows by considering an appropriate random graph (see the full version of [10].)

Part III

Algorithmic Results Related to Property Testing

Chapter 7

Additive Approximation for Edge-Deletion Problems

7.1 The Main Results

7.1.1 An algorithm for any monotone property

Our main focus in this part of the thesis is in approximation algorithms for the edit distance of a graph from satisfying some (monotone) graph property. For a graph property $\mathcal{P}$, let $\mathcal{P}_n$ denote the set of graphs on n vertices, which satisfy $\mathcal{P}$. Given two graphs on n vertices, G and G', we denote by $\Delta(G, G')$ the edit distance between G and G', namely the smallest number of edge additions and/or deletions that are needed in order to turn G into G'. For a given property $\mathcal{P}$, we want to denote how far is a graph G from satisfying $\mathcal{P}$. For notational reasons it will be more convenient to normalize this measure so that it is always in the interval $[0, 1]$ (actually $[0, \frac{1}{2}]$). We thus define

Definition 7.1. **($E_{\mathcal{P}}(G)$)** *For a graph property $\mathcal{P}$ and a graph G on n vertices, let*

$$E_{\mathcal{P}}(G) = \min_{G' \in \mathcal{P}_n} \frac{\Delta(G, G')}{n^2} .$$

In words, $E_{\mathcal{P}}(G)$ is the minimum edit distance of G to a graph satisfying $\mathcal{P}$ after normalizing it by a factor of n^2.

Our first main result in this chapter states that for any graph property $\mathcal{P}$, which belongs to the large, natural and well studied family of monotone graph properties, it is possible to derive efficient approximations of $E_{\mathcal{P}}$.

Theorem 7.2. *For any fixed $\epsilon > 0$ and any monotone property $\mathcal{P}$ there is a* **deterministic** *algorithm that given a graph G on n vertices computes in time $O(n^2)$ a real E satisfying $|E - E_{\mathcal{P}}(G)| \leq \epsilon$.*

Note, that the running time of our algorithm is of type $f(\epsilon)n^2$, and can in fact be improved to linear in the size of the input by first counting the number of edges, taking

$E = 0$ in case the graph has less than ϵn^2 edges. We note that Theorem 7.2 was not known for many monotone properties. In particular, such an approximation algorithm was not even known for the property of being triangle-free and more generally for the property of being H-free for any non-bipartite H.

Theorem 7.2 is obtained via a novel structural graph theoretic technique. One of the applications of this technique (roughly) yields that every graph G, can be approximated by a small weighted graph W, in such a way that $E_{\mathcal{P}}(G)$ is approximately the optimal solution of a certain related problem (explained precisely in Section 7.3) that we solve on W. The main usage of this new structural-technique in this chapter is in proving Lemmas 7.20 and 7.21, which lie at the core of the proof of Theorem 7.2. This new technique, which may very well have other algorithmic and graph-theoretic applications, applies a result of Alon, Fischer, Krivelevich and Szegedy [6], which is a strengthening of Szemerédi's Regularity Lemma [112]. We then use an efficient algorithmic version of the regularity lemma, which also implies an efficient algorithmic version of the result of [6], in order to transform the existential structural result into the algorithm stated in Theorem 7.2.

We further use our structural result in order to prove the following concentration-type result regarding the edit distance of subgraphs of a graph.

Theorem 7.3. *For every ϵ and any monotone property $\mathcal{P}$ there is a $d = d(\epsilon, \mathcal{P})$ with the following property: Let G be any graph and suppose we randomly pick a subset D, of d vertices from $V(G)$. Denote by G' the graph induced by G on D. Then,*

$$Prob[\ |E_{\mathcal{P}}(G') - E_{\mathcal{P}}(G)| > \epsilon] < \epsilon\ .$$

An immediate implication of the above theorem is the following,

Corollary 7.4. *For every $\epsilon > 0$ and any monotone property $\mathcal{P}$ there is a* **randomized** *algorithm, which given a graph G computes in time $O(1)$ a real E satisfying $|E - E_{\mathcal{P}}(G)| \leq \epsilon$ with probability at least $1 - \epsilon$.*

We stress that there are some computational subtleties regrading the implementation of the algorithmic results discussed above. Roughly speaking, one should define how the property $\mathcal{P}$ is "given" to the algorithm and also whether ϵ is a fixed constant or part of the input. These issues are discussed in Section 7.5.

7.1.2 On the possibility of better approximations

Theorem 7.2 implies that it is possible to efficiently approximate the distance of an n vertex graph from any monotone graph property $\mathcal{P}$, to within an error of ϵn^2 for any $\epsilon > 0$. A natural question one can ask is for which monotone properties it is possible to improve the additive error to $n^{2-\delta}$ for some fixed $\delta > 0$. In the terminology of Definition 7.1, this means to approximate $E_{\mathcal{P}}$ to within an additive error of $n^{-\delta}$ for some $\delta > 0$. Our second main result in this chapter is a precise characterization of the monotone graph properties for which such a $\delta > 0$ exists[1].

[1] We assume henceforth that $\mathcal{P}$ is not satisfied by all graphs.

Theorem 7.5. *Let $\mathcal{P}$ be a monotone graph property. Then,*

1. *If there is a bipartite graph that does not satisfy $\mathcal{P}$, then there is a fixed $\delta > 0$ for which it is possible to approximate $E_{\mathcal{P}}$ to within an additive error of $n^{-\delta}$ in polynomial time.*

2. *On the other hand, if all bipartite graphs satisfy $\mathcal{P}$, then for any fixed $\delta > 0$ it is NP-hard to approximate $E_{\mathcal{P}}$ to within an additive error of $n^{-\delta}$.*

While the first part of the above theorem follows easily from the known results about the Turán numbers of bipartite graphs (see, e.g., [113]), the proof of the second item involves various combinatorial tools. These include Szemerédi's Regularity Lemma, and a new result in Extremal Graph Theory, which is stated in Theorem 7.30 (see Section 7.6) that extends the main result of [36] and [27]. We also use the basic approach of [2], which applies spectral techniques to obtain an NP-hardness result by embedding a blow-up of a sparse instance to a problem, in an appropriate dense pseudo-random graph. Theorem 7.30 and the proof technique of Theorem 7.5 may be useful for other applications in graph theory and in proving hardness results. As in the case of Theorem 7.2, the second part of Theorem 7.5 was not known for many specific monotone properties. For example, prior to this work it was not even known that it is NP-hard to *precisely* compute $E_{\mathcal{P}_{K_4}}$, where $\mathcal{P}_{K_4}$ is the property of being K_4-free [2]. More generally, the only non-bipartite graphs H for which it was known that computing $E_{\mathcal{P}_H}$ is NP-hard, where $\mathcal{P}_H$ is the property of being H-free, are the odd-cycles that were studied by Yannakakis [115].

7.1.3 Related work

Our main results form a natural continuation and extension of several research paths that have been extensively studied. Below we survey some of them.

Approximations of graph-modification problems: As we have previously mentioned many practical optimization problems in various research areas can be posed as the problem of computing the edit-distance of a certain graph from satisfying a certain property. Cai [40] has shown that for any hereditary property, which is expressible by a finite number of forbidden induced subgraphs, the problem of computing the edit distance is fixed-parameter tractable. Khot and Raman [87] proved that for some hereditary properties $\mathcal{P}$, finding in a given graph G, a subgraph that satisfies $\mathcal{P}$ is fixed-parameter tractable, while for other properties finding such a subgraph is hard in an appropriate sense (see [87]).

Note that Theorem 7.2 implies that if the edit distance (in our case, number of edge removals) of a graph from a property is $\Omega(n^2)$, then it can be approximated to within any *multiplicative* constant $1 + \epsilon$.

[2] K_t is a complete graph (clique) of size t.

Hardness of edge-modification problems: Natanzon, Shamir and Sharan [99] proved that for various hereditary properties, such as being Perfect and Comparability, computing $E_{\mathcal{P}}$ is NP-hard and sometimes even NP-hard to approximate to within some constant. Yannakakis [115] has shown that for several graph properties such as outerplanar, transitively orientable, and line-invertible, computing $E_{\mathcal{P}}$ is NP-hard. Asano [24] and Asano and Hirata [25] have shown that properties expressible in terms of certain families of forbidden minors or topological minors are NP-hard.

The NP-completeness proofs obtained by Yannakakis in [115], were add-hoc arguments that applied only to specific properties. Yannakakis posed in [115] as an open problem, the possibility of proving a general NP-hardness result for computing $E_{\mathcal{P}}$ that will apply to a general family of graph properties. Theorem 7.5 achieves such a result even for the seemingly easier problem of approximating $E_{\mathcal{P}}$.

Approximation schemes for "dense" instances: Fernandez de la Vega [58] and Arora, Karger and Karpinski [23] showed that many of the classical NP-complete problems such as MAX-CUT and MAX-3-CNF have a PTAS when the instance is dense, namely if the graph has $\Omega(n^2)$ edges or the 3-CNF formula has $\Omega(n^3)$ clauses. Approximations for dense instances of Quadratic Assignment Problems, as well as for additional problems, were obtained by Arora, Frieze and Kaplan [22]. Frieze and Kannan [70] obtained approximations schemes for several dense graph theoretic problems via certain matrix approximations. Alon, Fernandez de la Vega, Kannan and Karpinski [5] obtained results analogous to ours for any dense Constraint-Satisfaction-Problem via certain sampling techniques. It should be noted that all the above approximation schemes are obtained in a way similar to ours, that is, by first proving an *additive* approximation, and then arguing that in case the optimal solution is large (that is, $\Omega(n^2)$ in case of graphs, or $\Omega(n^3)$ in case of 3-CNF) the small additive error translates into a small multiplicative error.

All the above approximation results apply to the family of so called Constraint-Satisfaction-Problems. In some sense, these problems can express graph properties for which one imposes restrictions on **pairs** of vertices, such as k-colorability. These techniques thus fall short from applying to properties as simple as Triangle-freeness, where the restriction is on triples of vertices. The techniques we develop in order to obtain Theorem 7.2 enable us to handle restrictions that apply to *arbitrarily* large sets of vertices.

We briefly mention that $E_{\mathcal{P}}$ is related to packing problems of graphs. In [82] and [116] it was shown that by using linear programming one can approximate the packing number of a graph. In Section 7.9 we explain why this technique does not allow one to approximate $E_{\mathcal{P}}$.

Algorithmic applications of Szemerédi's Regularity Lemma: The authors of [4] gave a polynomial time algorithmic version of Szemerédi's Regularity Lemma. They used it to prove that Theorem 7.2 holds for the k-colorability property. The running time of their algorithm was improved by Kohayakawa, Rödl and Thoma [89]. Frieze and Kannan [69] further used the algorithmic version of the regularity lemma, to obtain approximation schemes for additional graph problems.

Theorem 7.2 is obtained via the algorithmic version of a strengthening of the standard regularity lemma, which was proved in [6], and it seems that these results cannot be obtained using the standard regularity lemma.

Tolerant Property-Testing: In standard Property-Testing one wants to distinguish between the graphs G that satisfy a certain graph property $\mathcal{P}$, or equivalently those G for which $E_{\mathcal{P}}(G) = 0$, from those that satisfy $E_{\mathcal{P}}(G) > \epsilon$. The main goal in designing property-testers is to reduce their query-complexity, namely, minimize the number of queries of the form "are i and j connected in the input graphs?".

Parnas, Ron and Rubinfeld [103] introduced the notion of Tolerant Property-Testing, where one wants to distinguish between the graphs G that satisfy $E_{\mathcal{P}}(G) < \delta$ from those that satisfy $E_{\mathcal{P}}(G) > \epsilon$, where $0 \leq \delta < \epsilon \leq 1$ are some constants. Recently, there have been several results in this line of work. Specifically, Fischer and Newman [64] have recently shown that if a graph property is testable with number of queries depending on ϵ only, then it is also tolerantly testable for any $0 \leq \delta < \epsilon \leq 1$ and with query complexity depending on $|\epsilon - \delta|$. Combining this with Theorem 1.1 implies that any monotone property is tolerantly testable for any $0 \leq \delta < \epsilon \leq 1$ and with query complexity depending on $|\epsilon - \delta|$. Note, that Corollary 7.4 implicitly states the same. In fact, the algorithm implied by Corollary 7.4 is the "natural" one, where one picks a random subset of vertices S, and approximates $E_{\mathcal{P}}(G)$ by computing $E_{\mathcal{P}}$ on the graph induced by S. The algorithm of [64] is far more complicated. Furthermore, due to the nature of our algorithm if the input graph satisfies a monotone property $\mathcal{P}$, namely if $E_{\mathcal{P}}(G) = 0$, we will always detect that this is the case. The algorithm of [64] may declare that $E_{\mathcal{P}}(G) > 0$ even if $E_{\mathcal{P}}(G) = 0$.

Organization: The proofs of the main results of this chapter, Theorems 7.2 and 7.5, are independent of each other. Sections 7.2, 7.3, 7.4 and 7.5 contain the proofs relevant to Theorem 7.2 and Sections 7.6, 7.7 and 7.8 contain the proofs relevant to Theorem 7.5.

In Section 7.2 we introduce the basic notions of regularity and state the regularity lemmas that we use for proving Theorem 7.2 and some of their standard consequences. In Section 7.3 we give a high level description of the main ideas behind our algorithms. We also state the main structural graph theoretic lemmas, Lemmas 7.20 and 7.21, which lie at the core of these algorithms. The proofs of these lemmas appear in Section 7.4. In Section 7.5 we give the proof of Theorems 7.2 and 7.3 as well as a discussion about some subtleties regarding the implementation of these algorithms.

Section 7.6 contains a high-level description of the proof of Theorem 7.5 as well as a description of the main tools, which we apply in this proof. In Section 7.7 we prove a new Extremal Graph-Theoretic result, which lies at the core of the proof of Theorem 7.5. In Section 7.8 we give the detailed proof of Theorem 7.5.

The final Section 7.9 contains some concluding remarks and open problems. Throughout the chapter, whenever we relate, for example, to a function $f_{3.1}$, we mean the function f defined in Lemma/Claim/Theorem 3.1.

7.2 Regularity Lemmas and their Algorithmic Versions

In this section we discuss the basic notions of regularity, some of the basic applications of regular partitions and state the regularity lemmas that we use in the proof of Theorems 7.2 and 7.3. See [90] for a comprehensive survey on the regularity-lemma. Some of the material appearing in this section overlaps with the material presented in Section 1.2. For completeness and self-containment of the presentation of this chapter, we give here a slightly different version of some of the notions related to the regularity lemma. In particular, while the presentation in Section 1.2 is oriented towards an application of Lemma 1.14, the presentation here is oriented towards an application of Lemma 7.14. Also, note that the notion of an $\mathcal{E}$-regular partition, which is defined below, was not used in Section 1.2.

We start by recalling some basic definitions. For every two nonempty disjoint vertex sets A and B of a graph G, we define $e(A,B)$ to be the number of edges of G between A and B. The *edge density* of the pair is defined by $d(A,B) = e(A,B)/|A||B|$.

Definition 7.6. (γ-regular pair) *A pair (A,B) is* γ-regular*, if for any two subsets $A' \subseteq A$ and $B' \subseteq B$, satisfying $|A'| \geq \gamma|A|$ and $|B'| \geq \gamma|B|$, the inequality $|d(A',B') - d(A,B)| \leq \gamma$ holds.*

Throughout the chapter we will make an extensive use of the notion of graph homomorphism, which we turn to formally define.

Definition 7.7. (Homomorphism) *A homomorphism from a graph F to a graph K, is a mapping $\varphi : V(F) \mapsto V(K)$ that maps edges to edges, namely $(v,u) \in E(F)$ implies $(\varphi(v),\varphi(u)) \in E(K)$.*

In what follows, $F \mapsto K$ denotes the fact that there is a homomorphism from F to K. We will also say that a graph H is homomorphic to K if $H \mapsto K$. Note, that a graph H is homomorphic to a complete graph of size k if and only if H is k-colorable.

Let F be a graph on f vertices and K a graph on k vertices, and suppose $F \mapsto K$. Let G be a graph obtained by taking a copy of K, replacing every vertex with a sufficiently large independent set, and every edge with a random bipartite graph of edge density d. It is easy to show that with high probability, G contains a copy of F (in fact, many). The following lemma shows that in order to infer that G contains a copy of F, it is enough to replace every edge with a "regular enough" pair. Intuitively, the larger f and k are, and the sparser the regular pairs are, the more regular we need each pair to be, because we need the graph to be "closer" to a random graph. This is formulated in the lemma below. Several versions of this lemma were previously proved in papers using the regularity lemma (see [90]).

Lemma 7.8. *For every real $0 < \eta < 1$, and integers $k, f \geq 1$ there exist $\gamma = \gamma_{7.8}(\eta,k,f)$, and $N = N_{7.8}(\eta,k,f)$ with the following property. Let F be any graph on f vertices, and let $U_1,\ldots,U_k$ be k pairwise disjoint sets of vertices in a graph G, where $|U_1| = \ldots = |U_k| \geq N$. Suppose there is a mapping $\varphi : V(F) \mapsto \{1,\ldots,k\}$ such that the following holds: If (i,j) is an edge of F then $(U_{\varphi(i)}, U_{\varphi(j)})$ is γ-regular with density at least η. Then $U_1,\ldots,U_k$ span a copy of F.*

Remark 7.9. *Observe that the function $\gamma_{7.8}(\eta, k, f)$ may and will be assumed to be monotone non-increasing in k and f and monotone non-decreasing in η. Therefore, it will be convenient to assume that $\gamma_{7.8}(\eta, k, f) \leq \eta^2$. Similarly, we will assume that $N_{7.8}(\eta, k, f)$ is monotone non-decreasing in k and f. Also, for ease of future definitions (in particular those given in (7.2)) set $\gamma_{7.8}(\eta, k, 0) = N_{7.8}(\eta, k, 0) = 1$ for any $k \geq 1$ and $0 < \eta < 1$.*

A partition $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$ of the vertex set of a graph is called an *equipartition* if $|V_i|$ and $|V_j|$ differ by no more than 1 for all $1 \leq i < j \leq k$ (so in particular each V_i has one of two possible sizes). The *order* of an equipartition denotes the number of partition classes (k above). A *refinement* of an equipartition $\mathcal{A}$ is an equipartition of the form $\mathcal{B} = \{V_{i,j} \mid 1 \leq i \leq k,\ 1 \leq j \leq l\}$ such that $V_{i,j}$ is a subset of V_i for every $1 \leq i \leq k$ and $1 \leq j \leq l$.

Definition 7.10. (γ-regular equipartition) *An equipartition $\mathcal{B} = \{V_i \mid 1 \leq i \leq k\}$ of the vertex set of a graph is called γ-regular if all but at most $\gamma\binom{k}{2}$ of the pairs $(V_i, V_{i'})$ are γ-regular.*

The Regularity Lemma of Szemerédi can be formulated as follows.

Lemma 7.11. ([112]) *For every m and $\gamma > 0$ there exists $T = T_{7.11}(m, \gamma)$ with the following property: If G is a graph with $n \geq T$ vertices, and $\mathcal{A}$ is an equipartition of the vertex set of G of order at most m, then there exists a refinement $\mathcal{B}$ of $\mathcal{A}$ of order k, where $m \leq k \leq T$ and $\mathcal{B}$ is γ-regular.*

$T_{7.11}(m, \gamma)$ may and is assumed to be monotone non-decreasing in m and monotone non-increasing in γ. Szemerédi's original proof of Lemma 7.11 was only existential as it supplied no efficient algorithm for obtaining the required equipartition. Alon et. al. [4] were the first to obtain a polynomial time algorithm for finding the equipartition, whose existence is guaranteed by lemma 7.11. The running time of this algorithm was improved by Kohayakawa et. al. [89] who obtained the following result.

Lemma 7.12. ([89]) *For every fixed m and γ there is an $O(n^2)$ time algorithm, which given an equipartition $\mathcal{A}$ finds equipartition $\mathcal{B}$ as in Lemma 7.11.*

Our main tool in the proof of Theorem 7.2 is Lemma 7.14 below, proved in [6]. This lemma can be considered a strengthening of Lemma 7.11, as it guarantees the existence of an equipartition and a refinement of this equipartition, which poses stronger properties compared to those of the standard γ-regular equipartition. This stronger notion is defined below.

Definition 7.13. ($\mathcal{E}$-regular equipartition) *For a function $\mathcal{E}(r) : \mathbb{N} \mapsto (0, 1)$, a pair of equipartitions $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$ and its refinement $\mathcal{B} = \{V_{i,j} \mid 1 \leq i \leq k,\ 1 \leq j \leq l\}$, are said to be $\mathcal{E}$-regular if*

1. *For all $1 \leq i < i' \leq k$, for all $1 \leq j, j' \leq l$ but at most $\mathcal{E}(k)l^2$ of them, the pair $(V_{i,j}, V_{i',j'})$ is $\mathcal{E}(k)$-regular.*

2. *All* $1 \leq i < i' \leq k$ *but at most* $\mathcal{E}(0)\binom{k}{2}$ *of them are such that for all* $1 \leq j, j' \leq l$ *but at most* $\mathcal{E}(0)l^2$ *of them* $|d(V_i, V_{i'}) - d(V_{i,j}, V_{i',j'})| < \mathcal{E}(0)$ *holds.*

It will be very important for what follows to observe that in Definition 7.13 we may use an arbitrary *function* rather than a fixed γ as in Definition 7.10 (such functions will be denoted by $\mathcal{E}$ throughout the chapter). The following is one of the main results of [6].

Lemma 7.14. ([6]) *For any integer* m *and function* $\mathcal{E}(r) : \mathbb{N} \mapsto (0, 1)$ *there is* $S = S_{7.14}(m, \mathcal{E})$ *such that any graph on at least* S *vertices has an* $\mathcal{E}$*-regular equipartition* $\mathcal{A}$, $\mathcal{B}$ *where* $|\mathcal{A}| = k \geq m$ *and* $|\mathcal{B}| = kl \leq S$.

In order to make the presentation self contained we briefly review the proof of Lemma 7.14. Fix any m and function $\mathcal{E}$ and put $\zeta = \mathcal{E}(0)$. Partition G into m arbitrary subsets of equal size and denote this equipartition by $\mathcal{A}_0$. Put $M = m$. Iterate the following task: Apply Lemma 7.11 on $\mathcal{A}_{i-1}$ with $m = |\mathcal{A}_{i-1}|$ and $\gamma = \mathcal{E}(M)/M^2$ and let $\mathcal{A}_i$ be the refinement of $\mathcal{A}_{i-1}$ returned by Lemma 7.11. If $\mathcal{A}_{i-1}$ and $\mathcal{A}_i$ form an $\mathcal{E}$-regular equipartition stop, otherwise set $M = |\mathcal{A}_{i-1}|$ and reiterate. It is shown is [6] that after at most $100/\zeta^4$ iterations, for some $1 \leq i \leq 100/\zeta^4$ the partitions $\mathcal{A}_{i-1}$ and $\mathcal{A}_i$ form an $\mathcal{E}$-regular equipartition. Moreover, detecting an i for which this holds is very easy, that is, can be done in time $O(n^2)$ (see the proof in [6]). Note, that one can thus set the integer $S_{7.14}(m, \mathcal{E})$ to be the order of $\mathcal{A}_i$. In particular, the following is an immediate implication of the above discussion.

Proposition 7.15. *If* m *is bounded by a function of* ϵ *only, then for any* $\mathcal{E}$ *the integer* $S = S_{7.14}(m, \mathcal{E})$ *can be upper bounded by a function of* ϵ *only.*

The ϵ in the above proposition will be the ϵ from the task of approximating $E_{\mathcal{P}}$ within an error of ϵ in Theorem 7.2. Also, in our application of Lemma 7.14 the function $\mathcal{E}$ will (implicitly) depend on ϵ. For example, it will be convenient to set $\mathcal{E}(0) = \epsilon$. However, it follows from the definition of $S_{7.14}(m, \mathcal{E})$ given above that even in this case it is possible to upper bound $S_{7.14}(m, \mathcal{E})$ by a function of ϵ only.

In order to design our algorithm we will need to obtain the equipartitions $\mathcal{A}$ and $\mathcal{B}$, which appear in the statement of Lemma 7.14. However, note that by the overview of the proof of Lemma 7.14 given above, in order to obtain this partition one can use Lemma 7.12 as an efficient algorithm for obtaining the regular partitions. Moreover, by Proposition 7.15 whenever we apply either $\mathcal{E}$ or Lemma 7.12 we are guaranteed that m (which in the above overview was M) is upper bounded by some function of ϵ and γ is lower bounded by some function of ϵ. This means that each of the at most $100/\zeta^4$ applications of Lemma 7.15 takes $O(n^2)$ time. We thus get the following:

Proposition 7.16. *If* m *is bounded by a function of* ϵ *only, then for any* $\mathcal{E}$ *there is an* $O(n^2)$ *algorithm for obtaining the equipartitions* $\mathcal{A}$ *and* $\mathcal{B}$ *of Lemma 7.14.*

7.3 Overview of the Proof of the Algorithmic Result

We start with a convenient way of handling a monotone graph property.

Definition 7.17. (Forbidden Subgraphs) *For a monotone graph property $\mathcal{P}$, define $\mathcal{F} = \mathcal{F}_{\mathcal{P}}$ to be the set of graphs which are minimal with respect to not satisfying property $\mathcal{P}$. In other words, a graph F belongs to $\mathcal{F}$ if it does not satisfy $\mathcal{P}$, but any graph obtained from F by removing an edge or a vertex, satisfies $\mathcal{P}$.*

As an example of a family of forbidden subgraphs, consider $\mathcal{P}$ which is the property of being 2-colorable. Then $\mathcal{F}_{\mathcal{P}}$ is the set of all odd-cycles. Clearly, a graph satisfies $\mathcal{P}$ if and only it contains no member of $\mathcal{F}_{\mathcal{P}}$ as a (not necessarily induced) subgraph. We say that a graph is $\mathcal{F}$-free if it contains no (not necessarily induced) subgraph $F \in \mathcal{F}$. Clearly, for any family $\mathcal{F}$, being $\mathcal{F}$-free is a monotone property. Thus, the monotone properties are precisely the graph properties, which are equivalent to being $\mathcal{F}$-free for some family $\mathcal{F}$. In order to simplify the notation, it will be simpler to talk about properties of type $\mathcal{F}$-free rather than monotone properties. To avoid confusion we will henceforth denote by $E_{\mathcal{F}}(G)$ the value of $E_{\mathcal{P}}(G)$, where $\mathcal{F} = \mathcal{F}_{\mathcal{P}}$ as above.

The main idea we apply in order to obtain the algorithmic results of this chapter is quite simple; given a graph G, a family of forbidden subgraphs $\mathcal{F}$ and $\epsilon > 0$ we use Lemma 7.14 with appropriately defined parameters in order to construct in $O(n^2)$ time a weighted complete graph W, of size depending on ϵ but **independent** of the size of G, such that a solution of a certain "related" problem on W gives a good approximation of $E_{\mathcal{F}}(G)$. As W will be of size independent of the size of G, we may and will use exhaustive search in order to solve the "related" problem on W. In what follows we give further details on how to define W and the "related" problem that we solve on W.

We start with the simplest case, where the property is that of being triangle-free, namely $\mathcal{F} = \{K_3\}$. Let W be some weighted complete graph on k vertices and let $0 \leq w(i,j) \leq 1$ denote the weight of the edge connecting i and j in W. Let $E_{\mathcal{F}}(W)$ be the natural extension of the definition of $E_{\mathcal{F}}(G)$ to weighted graphs, namely, instead of just counting how many edges should be removed in order to turn G into an $\mathcal{F}$-free graph, we ask for the edge set of minimum weight with the above property. Let G be a k-partite graph on n vertices with partition classes $V_1, \ldots, V_k$ of equal size n/k. Suppose for every $i < j$ we have $d(V_i, V_j) = w(i,j)$ (recall that $d(V_i, V_j)$ denotes the edge density between V_i and V_j). In some sense, W can be considered a weighted approximation of G, but to our investigation a more important question is whether W can be used in order to estimate $E_{\mathcal{F}}(G)$? In other words, is it true that $E_{\mathcal{F}}(G) \approx E_{\mathcal{F}}(W)$?

It is easy to see that $E_{\mathcal{F}}(G) \leq E_{\mathcal{F}}(W)$. Indeed, given a set of edges S, whose removal turns W into a triangle free graph, we simply remove all edges connecting V_i and V_j for every $(i,j) \in S$. The main question is whether the other direction is also true. Namely, is it true that if it is possible to remove αn^2 from G and thus make it triangle free, then it is possible to remove from W a set of edges of total weight approximately αk^2 and thus make it triangle-free? If true this will mean that by computing $E_{\mathcal{F}}(W)$ we also approximately compute $E_{\mathcal{F}}(G)$. Unfortunately, this assertion is false in general, as the minimal number of edge modifications that are enough to make G triangle-free, may involve removing *some* and not *all* the edges connecting a pair (V_i, V_j), and in W we can remove only edges and not parts of them. It thus seems natural to ask what kind of restrictions should we impose on

G (or more precisely on the pairs (V_i, V_j)) such that the above situation will be impossible, namely, that the optimal way to turn G into a triangle free graph will involve removing either none or all the edges connecting a pair (V_i, V_j) (up to some small error). This will clearly imply that we also have $E_{\mathcal{F}}(G) \approx E_{\mathcal{F}}(W)$.

One natural restriction is that the pairs (V_i, V_j) would be random bipartite graphs. While this restriction indeed works it is of no use for our investigation as we are trying to design an algorithm that can handle arbitrary graphs and not necessarily random graphs. One is thus tempted to replace random bipartite graph with γ-regular pairs for some small enough γ. Unfortunately, we did not manage to prove that there is a small enough $\gamma > 0$ ensuring that even if all pairs (V_i, V_j) are γ-regular then $E_{\mathcal{F}}(G) \approx E_{\mathcal{F}}(W)$. In order to circumvent this difficulty we use the stronger notion of $\mathcal{E}$-regularity defined in Section 7.2. As it turns out, if one uses an appropriately defined function $\mathcal{E}$, then if all pairs (V_i, V_j) are $\mathcal{E}(k)$-regular, one can infer that $E_{\mathcal{F}}(G) \approx E_{\mathcal{F}}(W)$. This result is (essentially) formulated in Lemma 7.20.

In the above discussion we considered the case $\mathcal{F} = \{K_3\}$. So suppose now that $\mathcal{F}$ is an arbitrary (possibly infinite) family of graph. Suppose we use a weighted complete graph W on k vertices as above in order to approximate some k-partite graph. The question that naturally arises at this stage is what problem should we try to solve on W in order to get an approximation of $E_{\mathcal{F}}(G)$. It is easy to see that G may be very far from being $\mathcal{F}$-free, while at the same time W can be $\mathcal{F}$-free, simply because $\mathcal{F}$ does not contain graphs of size at most k. As an example, consider the case, where the property is that of containing no copy of the complete bipartite graph with two vertices in each side, denoted $K_{2,2}$. Now, if G is the complete bipartite graph $K_{n/2,n/2}$ then it is very far from being $K_{2,2}$-free. However, in this case W is just an edge, which spans no copy of $K_{2,2}$.

It thus seems that we must solve a *different* problem on W. To formulate this problem we need the following definitions.

Definition 7.18. ($\mathcal{F}$-homomorphism-free) *For a family of graphs $\mathcal{F}$, a graph W is called $\mathcal{F}$-homomorphism-free if $F \not\to W$ for any $F \in \mathcal{F}$.*

We now define a measure analogous to $E_{\mathcal{F}}$ but with respect to making a graph $\mathcal{F}$-homomorphism-free. Note that we focus on weighted graphs.

Definition 7.19. ($\mathcal{H}_{\mathcal{F}}(W)$) *For a family of graphs $\mathcal{F}$ and a weighted complete graph W on k vertices, let $\mathcal{H}'_{\mathcal{F}}(W)$ denote the minimum total weight of a set of edges, whose removal from W turns it into an $\mathcal{F}$-homomorphism-free graph. Define, $\mathcal{H}_{\mathcal{F}}(W) = \mathcal{H}'_{\mathcal{F}}(W)/k^2$.*

Note, that in Definition 7.18 the graph W is an unweighed not necessarily complete graph. Also, observe that when $\mathcal{F} = \{K_3\}$ then we have $\mathcal{H}_{\mathcal{F}}(W) = E_{\mathcal{F}}(W)$. As it turns out, the "right" problem to solve on W is to compute $\mathcal{H}_{\mathcal{F}}(W)$. This is formulated in the following key lemma, whose proof appears in Section 7.4:

Lemma 7.20. (The Key Lemma) *For every family of graphs $\mathcal{F}$, there are functions $N_{7.20}(k, \epsilon)$ and $\gamma_{7.20}(k, \epsilon)$ with the following property[3]: Let W be any weighted complete*

[3]The functions $N_{7.20}(k, \epsilon)$ and $\gamma_{7.20}(k, \epsilon)$ will also (implicitly) depend on $\mathcal{F}$.

graph on k vertices and let G be any k-partite graph with partition classes $V_1, \ldots, V_k$ of equal size such that

1. $|V_1| = \ldots = |V_k| \geq N_{7.20}(k, \epsilon)$.
2. *All pairs (V_i, V_j) are $\gamma_{7.20}(k, \epsilon)$-regular.*
3. *For every $1 \leq i < j \leq k$ we have $d(V_i, V_j) = w(i, j)$.*

Then, $E_{\mathcal{F}}(G) \geq \mathcal{H}_{\mathcal{F}}(W) - \epsilon$.

It is easy to argue as we did above and prove that $E_{\mathcal{F}}(G) \leq \mathcal{H}_{\mathcal{F}}(W)$ in Lemma 7.20 (see the proof of Lemma 7.21), however we will not need this (trivial) direction. It is important to note that while Lemma 7.20 is very strong as it allows us to approximate $E_{\mathcal{F}}(G)$ via computing $\mathcal{H}_{\mathcal{F}}(W)$ (recall that W is intended to be very small compared to G) its main weakness is that it requires the regularity between each of the pairs to be a function of k, which denotes the number of partition classes, rather than depending solely on the family of graphs $\mathcal{F}$. We note that even if $\mathcal{F} = \{K_3\}$ as discussed above, we can only prove Lemma 7.20 with a regularity measure that depends on k. This supplies some explanation as to why Lemma 7.11 (the standard regularity lemma) is not sufficient for our purposes; note that the input to Lemma 7.11 is some fixed $\gamma > 0$ and the output is a γ-regular equipartition with number of partition classes that depends on γ (the function $T_{7.11}(m, \gamma)$). Thus, even if all pairs are γ-regular, this γ may be very large when considering the number of partition classes returned by Lemma 7.11 and the regularity measure which Lemma 7.20 requires. Hence, the standard regularity lemma cannot help us with applying Lemma 7.20. In order to overcome this problem we use the notion of $\mathcal{E}$-regular partitions and the stronger regularity-lemma given in Lemma 7.14, which, when appropriately used, allows us to apply Lemma 7.20 in order to obtain Lemma 7.21 below, from which Theorem 7.2 follows quite easily. The proof of this lemma appears in Section 7.4.

Lemma 7.21. *For any $\epsilon > 0$ and family of graphs $\mathcal{F}$ there are functions $N_{7.21}(r)$ and $\mathcal{E}_{7.21}(r)$ satisfying the following[4]: Suppose a graph G has an $\mathcal{E}_{7.21}$-regular equipartition $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$, $\mathcal{B} = \{V_{i,j} \mid 1 \leq i \leq k,\ 1 \leq j \leq l\}$, where*

1. $k \geq 1/\epsilon$.
2. $|V_{i,j}| \geq N_{7.21}(k)$ *for every* $1 \leq i \leq k$ *and* $1 \leq j \leq l$.

Let W be a weighted complete graph on k vertices with $w(i, j) = d(V_i, V_j)$. Then,

$$|E_{\mathcal{F}}(G) - \mathcal{H}_{\mathcal{F}}(W)| \leq \epsilon \ .$$

Using the algorithmic version of Lemma 7.14, which is given in Proposition 7.16, we can rephrase the above lemma in a more algorithmic way, which is more or less the algorithm of Theorem 7.2: Given a graph G we use the $O(n^2)$ time algorithm of Proposition 7.16

[4]The functions $N_{7.21}(r)$ and $\mathcal{E}_{7.21}(r)$ will also (implicitly) depend on ϵ and $\mathcal{F}$.

in order to obtain the equipartition described in the statement of Lemma 7.21. We then construct the graph W as in Lemma 7.21, and finally use exhaustive search in order to precisely compute $\mathcal{H}_{\mathcal{F}}(W)$. By Lemma 7.21, this gives a good approximation of $E_{\mathcal{F}}(G)$. The proof of Theorem 7.2 appears in Section 7.5.

7.4 Proofs of Structural Lemmas

In this section we apply our new structural technique in order to prove Lemmas 7.20 and 7.21. Regretfully, it is hard to precisely state what are the ingredients of this technique. Roughly speaking, it uses the notion of $\mathcal{E}$-regularity in order to partition the edges of a graph into a bounded number of edge sets, which have regular-partitions that are almost identical[5] and more importantly, the regularity-measure of each of the bipartite graph in each of the edge sets can be a function of the number of clusters.

We start this section with some definitions that will be very useful for the proof of Lemma 7.20. These notions were used in Subsection 1.3.1 and the reader can find more details and intuition regarding them in that subsection.

Definition 7.22. *For any (possibly infinite) family of graphs $\mathcal{F}$, and any integer r let $\mathcal{F}_r$ be the following set of graphs: A graph R belongs to $\mathcal{F}_r$ if it has at most r vertices and there is at least one $F \in \mathcal{F}$ such that $F \mapsto R$.*

Definition 7.23. *For any family of graphs $\mathcal{F}$ and integer r for which $\mathcal{F}_r \neq \emptyset$, define*

$$\Psi_{\mathcal{F}}(r) = \max_{R \in \mathcal{F}_r} \min_{\{F \in \mathcal{F}: F \mapsto R\}} |V(F)|. \tag{7.1}$$

Define $\Psi_{\mathcal{F}}(r) = 0$ if $\mathcal{F}_r = \emptyset$. Therefore, $\Psi_{\mathcal{F}}(r)$ is monotone non-decreasing in r.

Practicing definitions, note that if $\mathcal{F}$ is the family of odd cycles, then $\mathcal{F}_k$ is precisely the family of non-bipartite graphs of size at most k. Also, in this case $\Psi_{\mathcal{F}}(k) = k$ when k is odd, and $\Psi_{\mathcal{F}}(k) = k - 1$ when k is even. The "right" way to think of the function $\Psi_{\mathcal{F}}$ is the following: Let R be a graph of size at most k and suppose we are guaranteed that there is a graph $F' \in \mathcal{F}$ such that $F' \mapsto R$ (thus $R \in \mathcal{F}_k$). Then by this information only and *without* having to know the structure of R itself, the definition of $\Psi_{\mathcal{F}}$ implies that there is a graph $F \in \mathcal{F}$ of size at most $\Psi_{\mathcal{F}}(k)$, such that $F \mapsto R$.

The function $\Psi_{\mathcal{F}}$ has a critical role in the proof of Lemma 7.20. While proving this lemma we will use Lemma 7.8 in order to derive that some k sets of vertices, which are regular enough, span some graph $F \in \mathcal{F}$. Roughly speaking, the main difficulty will be that we will not know the size of F, and as a consequence will not know the regularity measure between these sets that is sufficient for applying Lemma 7.8 on these k sets (this quantity is $\gamma_{7.8}(\eta, k, |V(F)|)$). However, we *will* know that there is *some* $F' \in \mathcal{F}$, which is spanned by these sets. The function $\Psi_{\mathcal{F}}(r)$ will thus be very useful as it supplies an upper bound for the size of the smallest $F \in \mathcal{F}$, which is spanned by these sets. See Proposition 7.25, where $\Psi_{\mathcal{F}}(r)$ has a crucial role.

[5] Two regular partitions $V_1, \ldots, V_k$ and $U_1, \ldots, U_k$ are identical if $d(V_i, V_j) = d(U_i, U_j)$

Proof of Lemma 7.20: Given ϵ and k let

$$T = T(k,\epsilon) = T_{7.11}(k, \gamma_{7.8}(\epsilon/2, k, \Psi_{\mathcal{F}}(k))). \tag{7.2}$$

We prove the lemma with $\gamma_{7.20}(k,\epsilon)$ and $N_{7.20}(k,\epsilon)$ satisfying

$$\gamma_{7.20}(k,\epsilon) = \min(\epsilon/2,\ 1/T), \tag{7.3}$$

$$N_{7.20}(k,\epsilon) = T \cdot N_{7.8}(\epsilon/2, k, \Psi_{\mathcal{F}}(k)) \tag{7.4}$$

Suppose G is a graph on n vertices, in which case each set V_i is of size $\frac{n}{k}$. We may thus show that one must remove at least $\mathcal{H}_{\mathcal{F}}(W)\cdot n^2 - \epsilon n^2$ edges from G in order to make it $\mathcal{F}$-free. To this end, it is enough to show that if there is a graph G', which is obtained from G by removing less than $\mathcal{H}_{\mathcal{F}}(W)\cdot n^2 - \epsilon n^2$ edges and spans no $F \in \mathcal{F}$ then it is possible to remove from W a set of edges of total weight less than $\mathcal{H}_{\mathcal{F}}(W)\cdot k^2$ and obtain a graph W', which is $\mathcal{F}$-homomorphism-free. This will obviously be a contradiction.

Assume such a G' exists and apply Lemma 7.11 on it with $\gamma = \gamma_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k))$ and $m = k$ (we use $m = k$ as G is already partitioned into k subsets $V_1, \ldots, V_k$). For the rest of the proof we denote by $V_{i,1}, \ldots, V_{i,l}$ the partition of V_i, which Lemma 7.11 returns. Recall that as $|V_1| = \ldots = |V_k|$ and Lemma 7.11 partitions a graph into subsets of equal size, then all the sets V_i are partitioned into the same number l of subsets. Note also that by Lemma 7.11 and the definition of T in (7.2) we have $l < T$. Observe, that T is in fact an upper bound for the *total* number of partition classes $V_{i,j}$).

By Lemma 7.11 (recall that by Remark 7.9 we may assume $\gamma_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k)) \le \frac{1}{2}\epsilon$), we are guaranteed that out of the lk sets $V_{i,j}$ at most $\frac{\epsilon}{2}\binom{lk}{2}$ pairs are not $\gamma_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k))$-regular. We define a graph G'', which is obtained from G' by removing all the edges connecting pairs $(V_{i,i'}, V_{j,j'})$ that are not $\gamma_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k))$-regular, and all edges connecting pairs $(V_{i,i'}, V_{j,j'})$ for which their edge density in G' is smaller than $\frac{1}{2}\epsilon$.

Proposition 7.24. *There are k sets $V_{1,t_1}, \ldots, V_{k,t_k}$ such that the graphs induced by G and G'' on these k sets differ by less than $\mathcal{H}_{\mathcal{F}}(W)\cdot\frac{n^2}{l^2} - \frac{\epsilon n^2}{2l^2}$ edges.*

Proof: We first claim that G'' is obtained from G' by removing less than $\frac{\epsilon}{2}n^2$ edges. To see this note that the number of edges connecting a pair $(V_{i,i'}, V_{j,j'})$ is at most $(n/kl)^2$. As there are at most $\frac{\epsilon}{2}\binom{lk}{2}$ pairs, which are not $\gamma_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k))$-regular we remove at most $\frac{\epsilon}{4}n^2$ edges due to such pairs. Finally, as due to pairs, whose edge density is at most $\frac{1}{2}\epsilon$, we remove at most $\binom{kl}{2}\frac{\epsilon}{2}(n/kl)^2 \le \frac{\epsilon}{4}n^2$ edges, the total number of edges removed is at most $\frac{\epsilon}{2}n^2$, as needed.

As we assume that G' is obtained from G by removing less than $\mathcal{H}_{\mathcal{F}}(W)\cdot n^2 - \epsilon n^2$ edges, we get from the previous paragraph that G'' is obtained from G be removing less than $\mathcal{H}_{\mathcal{F}}(W)\cdot n^2 - \frac{\epsilon}{2}n^2$ edges. Suppose for every $1 \le i \le k$ we randomly and uniformly pick one of the sets $V_{i,1}, \ldots, V_{i,l}$. The probability that an edge, which belongs to G and not to G'', is spanned by these k sets is l^{-2}. As G and G'' differ by less than $\mathcal{H}_{\mathcal{F}}(W)\cdot n^2 - \frac{\epsilon}{2}n^2$ edges,

we get that the expected number of such edges is less than $\mathcal{H}_{\mathcal{F}}(W) \cdot \frac{n^2}{l^2} - \frac{\epsilon n^2}{2l^2}$ and therefore there must be a choice of k sets, which span less than this number of such edges. $\square$

We are now ready to arrive at a contradiction by showing that if it is possible to remove less than $\mathcal{H}_{\mathcal{F}}(W) \cdot n^2 - \epsilon n^2$ edges from G and thus turn it into an $\mathcal{F}$-free graph G', then we can remove from W a set of edges of total weight less than $\mathcal{H}_{\mathcal{F}}(W) \cdot k^2$ and thus turn it into an $\mathcal{F}$-homomorphism-free graph W'. Let $V_{1,i_1}, \ldots, V_{k,i_k}$ be the k sets satisfying the condition of Proposition 7.24 and obtain from W a graph W' by removing from W edge (i,j) if and only if the density of (V_{i,t_i}, V_{j,t_j}) in G'' is 0.

Proposition 7.25. *W' is $\mathcal{F}$-homomorphism-free.*

Proof: Assume $F' \mapsto W'$ for some $F' \in \mathcal{F}$. As W' is a graph of size k this means (recall Definition 7.23) that there is $F \in \mathcal{F}$ of size at most $\Psi_{\mathcal{F}}(k)$ such that $F \mapsto W'$. Let φ be a homomorphism from F to W'. By definition of φ, for any $(u,v) \in E(F)$ we have $(\varphi(u), \varphi(v))$ is an edge of W'. Recall that by definition of G'' either the density of a pair $(V_{i,i'}, V_{j,j'})$ in G'' is zero, or this density is at least $\frac{1}{2}\epsilon$ and the pair is $\gamma_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k))$-regular. By definition of W', this means that for every $(u,v) \in E(F)$ the pair $(V_{\varphi(u),t_{\varphi(u)}}, V_{\varphi(v),t_{\varphi(v)}})$ has density at least $\frac{\epsilon}{2}$ in G'' and is $\gamma_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k))$-regular. By item 1 of the lemma we have for all $1 \leq i \leq k$ that $|V_i| \geq N_{7.20}(k,\epsilon)$. By our choice in (7.4) and the fact that $l \leq T$, the sets V_{i,t_i} must therefore be of size at least

$$|N_{7.20}(k,\epsilon)|/l \geq |N_{7.20}(k,\epsilon)|/T = N_{7.8}(\frac{1}{2}\epsilon, k, \Psi_{\mathcal{F}}(k)).$$

Hence, the sets $V_{1,t_1}, \ldots, V_{k,t_k}$ satisfy all the necessary requirements needed in order to apply Lemma 7.8 on them in order to deduce that they span a copy of F in G'' (recall, that we have already argued that $|V(F)| \leq \Psi_{\mathcal{F}}(k)$). This, however, is impossible, as we assumed that G' was already $\mathcal{F}$-free and G'' is a subgraph of G'. $\square$

Proposition 7.26. *For any $i < j$ the edge densities of (V_i, V_j) and (V_{i,t_i}, V_{j,t_j}) satisfy in G*

$$|d(V_i, V_j) - d(V_{i,t_i}, V_{j,t_j})| \leq \frac{1}{2}\epsilon.$$

Proof: Recall that $1/l > 1/T$ and by (7.3) we have $1/T > \gamma_{7.20}(k,\epsilon)$. We infer that $|V_{i,t_i}| = |V_i|/l \geq \gamma_{7.20}(k,\epsilon)|V_i|$. By item 2 of the lemma, each pair (V_i, V_j) is $\gamma_{7.20}(k,\epsilon)$-regular in G. Hence, by definition of a regular pair, we must have $|d(V_i,V_j) - d(V_{i,t_i}, V_{j,t_j})| \leq \gamma_{7.20}(k,\epsilon) \leq \frac{1}{2}\epsilon$. $\square$

Proposition 7.27. *W' is obtained from W by removing a set of edges of weight less than $\mathcal{H}_{\mathcal{F}}(W) \cdot k^2$.*

Proof: Let S be the set of edges removed from W and denote by $w(S)$ the total weight of edges in S. Let $|e(V_{i,t_i}, V_{j,t_j})|$ denote the number of edges connecting the pair (V_{i,t_i}, V_{j,t_j}) in G. We claim that the following series of inequalities, which imply that $w(S) < \mathcal{H}_{\mathcal{F}}(W) \cdot k^2$, hold:

$$\begin{aligned}\mathcal{H}_{\mathcal{F}}(W)\cdot\frac{n^2}{l^2}-\frac{\epsilon n^2}{2l^2} &> \sum_{(i,j)\in S}|e(V_{i,t_i},V_{j,t_j})|\\ &\geq \sum_{(i,j)\in S}(w(i,j)-\frac{\epsilon}{2})\frac{n^2}{l^2k^2}\\ &\geq \sum_{(i,j)\in S}w(i,j)\frac{n^2}{l^2k^2}-\frac{\epsilon n^2}{2l^2}\\ &= w(S)\frac{n^2}{l^2k^2}-\frac{\epsilon n^2}{2l^2}\ .\end{aligned}$$

Indeed, recall that by the definition of W', we have $(i,j)\in S$ if and only if the density of the pair $(V_{i,i'},V_{j,j'})$ in G'' is 0, which means that all the edges connecting this pair were removed in G''. As by Proposition 7.24 the total difference between G and G'' is less than $\mathcal{H}_{\mathcal{F}}(W)\cdot\frac{n^2}{l^2}-\frac{\epsilon n^2}{2l^2}$ we infer that the first (strict) inequality is valid. The second inequality follows from Proposition 7.26 together with the fact that by the condition of the lemma we have $d(V_i,V_j)=w(i,j)$. The third inequality is due to the fact that W has k vertices and thus $|S|\leq k^2$. □

The sought after contradiction now follows immediately from Propositions 7.25 and 7.27. This completes the proof of the lemma. □

We continue with the proof of Lemma 7.21.

Proof of Lemma 7.21: We prove the lemma with:

$$\mathcal{E}_{7.21}(r)=\begin{cases}\frac{1}{16}\epsilon^2, & r=0\\ \min(\frac{1}{8}\epsilon r^{-2},\ \frac{1}{8}\epsilon^2,\ \gamma_{7.20}(r,\frac{1}{8}\epsilon)), & r\geq 1\end{cases}\tag{7.5}$$

and

$$N_{7.21}(r)=N_{7.20}(r,\frac{1}{8}\epsilon)\ .$$

We start with showing that $E_{\mathcal{F}}(G)\leq\mathcal{H}_{\mathcal{F}}(W)+\epsilon$. Suppose G is a graph of n vertices, in which case the number of edges connecting V_i and V_j is $w(i,j)\cdot\frac{n^2}{k^2}$. We first remove all the edges within the sets $V_1,\ldots,V_k$. As $k\geq 1/\epsilon$ the total number of edges removed in this step is at most $k\binom{n/k}{2}\leq\epsilon n^2$.

Let S be the set of minimal weight whose removal turns W into an $\mathcal{F}$-homomorphism-free graph W'. We claim that if for every $(i,j)\in S$ we remove all the edges connecting V_i and V_j the resulting graph G' spans no copy of a graph $F\in\mathcal{F}$. Suppose to the contrary that G' spans a copy of $F\in\mathcal{F}$, and consider the mapping $\varphi:V(F)\mapsto\{1,\ldots,k\}$, which maps every vertex of F that belongs to V_j to j. As we have removed all the edges within the sets $V_1,\ldots,V_k$ and all edges between V_i and V_j for any $(i,j)\in S$ we get that φ is a

homomorphism from F to W' contradicting our choice of S. Finally, note that the number of edges removed in the second step is

$$\sum_{(i,j)\in S} w(i,j)\cdot\frac{n^2}{k^2} = n^2\cdot\mathcal{H}_{\mathcal{F}}(W)\ .$$

Combined with the first step the total number of edges removed is at most $n^2\cdot\mathcal{H}_{\mathcal{F}}(W)+\epsilon n^2$, as needed.

For the rest of the proof we focus on proving $\mathcal{H}_{\mathcal{F}}(W)\leq E_{\mathcal{F}}(G)+\epsilon$. Let $\mathcal{A}$ and $\mathcal{B}$ be the two equipartitions from the statement of the lemma. Suppose for every $1\leq i\leq k$ we randomly, uniformly and independently pick a set V_{i,t_i} out of the sets $V_{i,1},\ldots,V_{i,l}$. Let P denote the event that (i) All the pairs $(V_{i,t_i},V_{i',t_{i'}})$ are $\mathcal{E}(k)$-regular. (ii) All but at most $\frac{1}{2}\epsilon\binom{k}{2}$ of the pairs $(V_{i,t_i},V_{i',t_{i'}})$ satisfy $|d(V_{i,t_i},V_{i',t_{i'}})-d(V_i,V_{i'})|\leq\mathcal{E}(0)$. We need the following observations:

Proposition 7.28. *P holds with probability at least $1-\frac{1}{2}\epsilon$.*

Proof: Fix any $i<i'$. By definition of $\mathcal{E}_{7.21}$ we have $\mathcal{E}(k)\leq\frac{1}{8}\epsilon k^{-2}$, thus by item 1 of Definition 7.13, the probability that $(V_{i,t_i},V_{i',t_{i'}})$ is not $\mathcal{E}(k)$-regular is at most $\frac{1}{8}\epsilon k^{-2}$. By the union bound, the probability that one of the pairs is not $\mathcal{E}(k)$-regular is at most $\binom{k}{2}\frac{1}{8}\epsilon k^{-2}\leq\frac{1}{4}\epsilon$.

Item 2 of Definition 7.13 can be rephrased as stating that there are at most $\mathcal{E}(0)\binom{k}{2}=\frac{1}{16}\epsilon^2\binom{k}{2}$ choices of $i<i'$ for which the probability that $|d(V_{i,t_i},V_{i',t_{i'}})-d(V_i,V_{i'})|>\mathcal{E}(0)=\frac{1}{16}\epsilon^2$ is larger than $\mathcal{E}(0)=\frac{1}{16}\epsilon^2$. Thus, the expected number of $i<i'$ for which $|d(V_{i,t_i},V_{i',t_{i'}})-d(V_i,V_{i'})|>\mathcal{E}(0)$ is at most $\frac{1}{16}\epsilon^2\binom{k}{2}\cdot 1+\binom{k}{2}\cdot\frac{1}{16}\epsilon^2\leq\frac{1}{8}\epsilon^2\binom{k}{2}$. By Markov's inequality, the probability that more than $\frac{1}{2}\epsilon\binom{k}{2}$ of $i<i'$ violate the above inequality is at most $\frac{\epsilon}{4}$.

As properties (i) and (ii) of event P each hold with probability at least $1-\frac{1}{4}\epsilon$, we get that P holds with probability at least $1-\frac{1}{2}\epsilon$. □

Proposition 7.29. *Assume event P holds and denote by G' the subgraph of G, which is spanned by the sets $V_{1,t_1},\ldots,V_{k,t_k}$. Then, $E_{\mathcal{F}}(G')\geq\mathcal{H}_{\mathcal{F}}(W)-\frac{1}{2}\epsilon$.*

Proof: Let W' be a weighted complete graph on k vertices satisfying $w(i,i')=d(V_{i,t_i},V_{i',t_{i'}})$. Event P assumes that all the pairs $(V_{i,t_i},V_{i',t_{i'}})$ are $\mathcal{E}(k)$-regular. As $\mathcal{E}(k)\leq\gamma_{7.20}(k,\frac{1}{8}\epsilon)$ and the lemma assumes that $|V_{i,j}|\geq N_{7.21}(k)=N_{7.20}(k,\frac{1}{8}\epsilon)$ we may deduce from Lemma 7.20 that

$$E_{\mathcal{F}}(G')\geq\mathcal{H}_{\mathcal{F}}(W')-\frac{\epsilon}{8}. \tag{7.6}$$

Now, event P also assumes that all but at most $\frac{\epsilon}{2}\binom{k}{2}$ of the pairs $i<i'$ are such that $|d(V_i,V_{i'})-d(V_{i,t_i},V_{i',t_{i'}})|\leq\mathcal{E}(0)<\frac{\epsilon}{8}$. This means that the sum of edge weights of W' differs from the sum of edge weights of W by at most $\frac{\epsilon}{2}\binom{k}{2}$ due to pairs that violate the above inequality and by at most $\binom{k}{2}\frac{\epsilon}{8}$ due to the other pairs. This means that the sum of edge weights of W' differs from that of W by at most $\frac{\epsilon}{4}k^2+\frac{\epsilon}{16}k^2\leq\frac{3\epsilon}{8}k^2$. This clearly

implies that

$$\mathcal{H}_{\mathcal{F}}(W') \geq \mathcal{H}_{\mathcal{F}}(W) - \frac{3\epsilon}{8} \; . \tag{7.7}$$

The proof now follows by combining (7.6) and (7.7). □

Let R be an arbitrary set of edges whose removal from G turns it into an $\mathcal{F}$-free graph. Randomly and uniformly select a set V_{i,t_i} from each of the sets $V_{i,1}, \ldots, V_{i,l}$, and let R' denote the set of edges of R, which are spanned by these k sets. We claim that the following upper and lower bound on the expected size of R' hold:

$$\begin{aligned}
\frac{1}{l^2} \cdot |R| &= \mathbb{E}[|R'|] \\
&\geq \mathbb{E}[|R'| \mid P] \cdot Prob[P] \\
&\geq (1 - \frac{\epsilon}{2}) \cdot \mathbb{E}[|R'| \mid P] \\
&\geq (1 - \frac{\epsilon}{2}) \cdot (\mathcal{H}_{\mathcal{F}}(W) - \frac{\epsilon}{2}) \cdot k^2 \frac{n^2}{(kl)^2} \\
&\geq (\mathcal{H}_{\mathcal{F}}(W) - \epsilon) \cdot \frac{n^2}{l^2} \; .
\end{aligned}$$

Indeed, the equality is due to the fact than an edge of R has probability precisely $1/l^2$ to be in R'. The second inequality is due to Proposition 7.28, the third is due to Proposition 7.29 and the last is valid because $\mathcal{H}_{\mathcal{F}}(W) \leq 1$. As we thus infer that $|R| \geq \mathcal{H}_{\mathcal{F}}(W) \cdot n^2 - \epsilon n^2$ for *arbitrary* R, we get that $E_{\mathcal{F}}(G) \geq \mathcal{H}_{\mathcal{F}}(W) - \epsilon$, thus completing the proof. □

7.5 Proofs of Algorithmic Results

The technical lemmas proved in the previous sections enabled us to infer that certain $\mathcal{E}$-regular partitions may be very useful for approximating $E_{\mathcal{P}}$. In this section we apply Proposition 7.16 in order to efficiently obtain these partitions. We first prove Theorem 7.2, while overlooking some subtle issues. We then discuss them is detail.

Proof of Theorem 7.2: Fix any $\epsilon > 0$ and monotone graph property $\mathcal{P}$. Let $\mathcal{F} = \mathcal{F}_{\mathcal{P}}$ be the family of forbidden subgraphs of $\mathcal{P}$ as in Definition 7.17. As satisfying $\mathcal{P}$ is equivalent to being $\mathcal{F}$-free, we focus on approximating $E_{\mathcal{F}}(G)$. Let $\mathcal{E}_{7.21}(r)$ and $N_{7.21}(r)$ be the appropriate function with respect to $\mathcal{F}$ and ϵ. Put $S(\epsilon) = S_{7.14}(1/\epsilon, \mathcal{E}_{7.21})$ and recall that by Proposition 7.15 the integer S can indeed be upper bounded by a function of ϵ.

If an input graph has less than $S(\epsilon) \cdot N_{7.21}(S(\epsilon))$ vertices we use exhaustive search in order to precisely compute $E_{\mathcal{F}}(G)$. Assume then that G has more than $S(\epsilon) \cdot N_{7.21}(S(\epsilon))$ vertices, and use Proposition 7.16 with $m = 1/\epsilon$ and $\mathcal{E}_{7.21}(r)$ as above in order to compute the equipartition $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$ and its refinement $\mathcal{B} = \{V_{i,j} \mid 1 \leq i \leq k, \; 1 \leq j \leq l\}$ satisfying the conditions of Lemma 7.14. As m is bounded by a function of ϵ we get

from Proposition 7.16 that this step takes time $O(n^2)$. Also, by Lemma 7.14 we have $kl \leq S$, therefore, as G has at least $S(\epsilon) \cdot N_{7.21}(S(\epsilon))$ vertices each of the sets $V_{i,j}$ is of size at least $N_{7.21}(S(\epsilon)) \geq N_{7.21}(k)$. Let W be a weighted complete graph of size k where $w(i,j) = d(V_i, V_j)$. Using exhaustive search, we can now precisely compute the value of $\mathcal{H}_{\mathcal{F}}(W)$. By Lemma 7.21 we may infer that $|E_{\mathcal{F}}(G) - \mathcal{H}_{\mathcal{F}}(W)| \leq \epsilon$. $\square$

As we have mentioned in the introduction, one should specify how the property $\mathcal{P}$ is given to the algorithm. For example, $\mathcal{P}$ may be an undecidable property, in which case we cannot do anything. We thus focus on decidable graph properties. However, even in this case we may face some unexpected problems. Note, that for a general infinite family of graphs $\mathcal{F}$ it is not clear how to compute $\mathcal{H}_{\mathcal{F}}$ in finite time. Also, returning to the overview of the proof of Lemma 7.14 given in Section 7.2, note that we have implicitly assumed that one can compute the function $\mathcal{E}$, as this is needed in order to compute the parameters with which one applies Lemma 7.15. A close inspection of the proofs of Lemmas 7.20 and 7.21 reveals that computing $\mathcal{E}$ involves computing the function $\Psi_{\mathcal{F}}$ (see (7.2), (7.3) and (7.5)). One of the main results Chapter 3 asserts that somewhat surprisingly, there is a family of graph properties $\mathcal{F}$, for which the property of being $\mathcal{F}$-free is decidable (in fact, in $coNP$) but at the same time $\Psi_{\mathcal{F}}$ is not computable. Therefore, even if we confine ourselves to decidable graph properties we still run into trouble.

Suppose first that ϵ is not part of the input to the algorithm. As we have discussed in Section 7.2, in this case all the applications of $\mathcal{E}_{7.21}$ are on inputs of size depending on ϵ only, thus the algorithm may "keep" the answers to these (finitely many) applications of $\mathcal{E}_{7.21}$ as part of its description. Similarly, in this case we may need to compute $\mathcal{H}_{\mathcal{F}}$ on graphs of size depending on ϵ only[6], thus the algorithm may "keep" the answers to these (finitely many) applications of $\mathcal{H}_{\mathcal{F}}$ as part of its description. Observe, that we don't need to keep the answer of $\mathcal{H}_{\mathcal{F}}$ for all the (infinite) range of edge weights. Rather, as we only need to approximate $E_{\mathcal{F}}$ within an additive error of ϵ, it is enough to consider edge weights $\{0, \epsilon, 2\epsilon, 3\epsilon, \ldots, 1\}$.

If we want the algorithm to be able to accept ϵ as part of the input, then we must confine ourselves to properties for which $\Psi_{\mathcal{F}}$ is computable. However, as for any reasonable graph property this function is computable, this is not a real constraint. For example, as we have mentioned in Section 7.4, if $\mathcal{P}$ is the property of being bipartite, then $\Psi_{\mathcal{F}}(k)$ is either k or $k-1$. Another natural family of properties for which $\Psi_{\mathcal{F}}(k)$ is computable is that of being H-free for a fixed graph H, as in this case $\Psi_{\mathcal{F}}(k) \leq |V(H)|$. By the definition of the function $\mathcal{E}_{7.21}$ we get that if $\Psi_{\mathcal{F}}$ is computable then so is $\mathcal{E}_{7.21}$. It is also not difficult to see that if $\Psi_{\mathcal{F}}$ is computable then so is $\mathcal{H}_{\mathcal{F}}$. Therefore, in case $\Psi_{\mathcal{F}}$ is computable, there is no problem with accepting ϵ as part of the input.

We now turn to prove Theorem 7.3. We note that the above difficulties are also relevant for Corollary 7.4, which applies Theorem 7.3, but we refrain from discussing them again.

Proof of Theorem 7.3: (sketch) As in the previous proof, we focus on the property

[6]Recall that the size of the graph on which we compute $\mathcal{H}_{\mathcal{F}}$ is the number of partition classes of the $\mathcal{E}$-regular partition, and this number is at most $S_{7.14}(m, \mathcal{E})$, which is bounded by a function of ϵ.

of being $\mathcal{F}$-free, where $\mathcal{F}$ is the family of forbidden subgraphs of $\mathcal{P}$. Suppose, as in the previous proof, that G is a large enough graph (in terms of ϵ) as otherwise we can take D to be the entire vertex set of G. Assume, we *implicitly* apply Lemma 7.14 on G and let $\mathcal{A} = \{V_i \mid 1 \leq i \leq k\}$, $\mathcal{B} = \{V_{i,j} \mid 1 \leq i \leq k,\ 1 \leq j \leq l\}$ be the equipartitions returned by the lemma. Let W be a weighted complete graph on k vertices, where $w(i,j) = d(V_i, V_j)$. By Lemma 7.21 we have

$$|E_{\mathcal{F}}(G) - \mathcal{H}_{\mathcal{F}}(W)| \leq \epsilon\ . \tag{7.8}$$

Let D be a random set of vertices and for $1 \leq i \leq k$ let U_i denote the vertices of D that belong to V_i, and for $1 \leq i \leq k, 1 \leq j \leq l$ let $U_{i,j}$ denote the vertices of D that belong to $V_{i,j}$. Recall that k and l are bounded by functions of ϵ. Using standard Chernoff Bounds (see, e.g., [20]), it is easy to see that if we use a large enough sample of vertices D (but only large enough in terms of ϵ), then with high probability (whp) we will have $|d(V_i, V_{i'}) - d(U_i, U_{i'})| \leq \epsilon$ for any $i < i'$ and $|d(V_{i,j}, V_{i',j'}) - d(U_{i,j}, U_{i',j'})| \leq \epsilon$ for any $i < i'$ and $j \neq j'$. Therefore, if W' is a weighted complete graph on k vertices, where $w(i,j) = d(U_i, U_j)$ then

$$|\mathcal{H}_{\mathcal{F}}(W) - \mathcal{H}_{\mathcal{F}}(W')| \leq \epsilon\ . \tag{7.9}$$

Furthermore, using Chernoff bounds again, one can show[7] that with high probability all the pairs $(U_i, U_{i'})$ and $(U_{i,j}, U_{i',j'})$ are as regular as $(V_i, V_{i'})$ and $(V_{i,j}, V_{i',j'})$ (up to ϵ). Therefore, the graph induced by D, denoted G', will have equipartitions $\mathcal{A}', \mathcal{B}'$ satisfying the requirements of Lemma 7.14. This means that

$$|E_{\mathcal{F}}(G') - \mathcal{H}_{\mathcal{F}}(W')| \leq \epsilon\ . \tag{7.10}$$

As (7.8), (7.9) and (7.10) all hold with high probability for any $\epsilon > 0$, we can thus make sure that with probability at least $1 - \epsilon$, we will have $|E_{\mathcal{F}}(G') - E_{\mathcal{F}}(G)| \leq \epsilon$. This completes the proof. □

7.6 Overview of the Proof of Hardness Result

For the proof of Theorem 7.5 it will be more convenient to denote by $E'_{\mathcal{P}}(G)$ the number of edge removals needed to make G satisfy $\mathcal{P}$, in other words $E'_{\mathcal{P}}(G) = n^2 \cdot E_{\mathcal{P}}(G)$. In particular, $E'_H(G)$ denotes the number of edge removals needed to turn G into an H-free graph. We will also denote by $E'_r(G)$ the number of edge removals needed to turn G into an r-partite graph (or equivalently r-colorable graph). Note, that approximating $E'_{\mathcal{P}}(G)$ within $n^{2-\delta}$ is equivalent to approximating $E_{\mathcal{P}}(G)$ within $n^{-\delta}$.

The main technical result we need in order to obtain Theorem 7.5 is an extension of some classical results in Extremal Graph Theory. Recall, that Turán's Theorem (see [113]) states that the largest K_{r+1}-free graph on n vertices (K_{r+1} = complete graph on $r+1$ vertices) is precisely the largest r-partite graph on n vertices. Another classical result

[7] In fact, showing this fact is not that trivial. This fact is proved in detail in Lemma 2.22

is the Erdős-Stone-Simonovits Theorem (see [113]), which states that for any graph H of chromatic number $r+1$, the largest H-free graph on n vertices has at most $o(n^2)$ more edges than the largest r-partite graph on n vertices. As any r-partite graph does not contain a copy of a graph of chromatic number $r+1$, the above results can thus be restated as saying that when $H = K_{r+1}$ we have $E'_H(K_n) = E'_r(K_n)$ and that for any H of chromatic number $r+1$ we have $E'_r(K_n) - o(n^2) \leq E'_H(K_n) \leq E'_r(K_n)$.

The main extremal graph-theoretic tool, which we use in order to obtain Theorem 7.5, is the following result, which greatly extends one of the main results of [36]. Note, that this result also extends Turán's Theorem and the Erdős-Stone-Simonovits Theorem as it states that $E'_H(G)$ and $E'_r(G)$ are very close not only when G is K_n but already when G has a sufficiently large minimal degree.

Theorem 7.30. *Let H be a graph of chromatic number $r+1 \geq 3$.*

(i) If there is an edge of H whose removal reduces its chromatic number, then there is constant $\mu = \mu(H) > 0$ such that if $G = (V, E)$ is a graph on n vertices of minimum degree at least $(1-\mu)n$, then $E'_H(G) = E'_r(G)$.

(ii) Otherwise, there are constants $\gamma = \gamma(H) > 0$ and $\mu = \mu(H) > 0$ such that if $G = (V, E)$ is a graph on n vertices of minimum degree at least $(1-\mu)n$, then

$$E'_r(G) - O(n^{2-\gamma}) \leq E'_H(G) \leq E'_r(G).$$

The assertion of this theorem for the special case of H being a triangle is proved in [36] and in a stronger form in [27]. We note that the $n^{2-\gamma}$ term in the second item of the theorem cannot be avoided. Note, that the error term we obtain in the second part of the theorem is better than the error term of the classical Erdős-Stone-Simonovits Theorem. Such improvement of the error term was previously known (see, e.g., [53] and [111]) but only for the case of G being K_n and not for G of sufficiently high minimal degree. The proof of Theorem 7.30 appears in Section 7.7.

Our second tool in the proof of Theorem 7.5 is certain pseudo-random graphs. An *(n, d, λ)-graph* is a d-regular graph on n vertices all of whose eigenvalues, except the first one, are at most λ in their absolute values. This notation was introduced by Alon in the 80s, motivated by the fact that if λ is much smaller than d, then such graphs have strong pseudo-random properties. In particular, (see, e.g., [20], Chapter 9), in this case the number of edges between any two sets of vertices U and W of G is roughly its expected value, which is $|U||W|d/n$, (see Section 7.8 for the precise statement). There are many known explicit constructions of (n, d, λ)-graphs that suffice for our purpose here. Specifically, we can use, for example, the graph constructed by Delsarte and Goethals and by Turyn (see [92]). In this graph the vertex set $V(G)$ consist of all elements of the two dimensional vector space over $GF(q)$ (q is any prime power), so G has $n = q^2$ vertices. To define the edges of G we fix a set L of k lines through the origin. Two vertices x and y of the graph G are adjacent if $x - y$ is parallel to a line in L. It is easy to check that this graph is $d = k(q-1)$-regular. Moreover, because it is a strongly regular graph, one can compute its eigenvalues precisely

and show that besides the first one they all are either $-k$ or $q-k$. Therefore, by choosing $k = (1-\mu)\frac{q^2}{q-1}$ we obtain an (n, d, λ)-graph with $d = (1-\mu)n$ and $\lambda \leq \sqrt{n}$ (μ will be chosen as the constant from Theorem 7.30).

Given a graph F let F_b denote the b-blowup of F, that is, the graph obtained from F by replacing every vertex $v \in V(F)$ with an independent set I_v, of size b, and by replacing every edge $(u, v) \in E(F)$, with a complete bipartite graph, whose partition classes are the independent sets I_u and I_v. It is not difficult to show (see Claim 7.39) that for any integer r, we have $E'_r(F_b) = b^2 E'_r(F)$. The final piece of notation we need is the Boolean Or, denoted by $G_1 \cup G_2$ of two graphs G_1 and G_2 on the same set of vertices V. Its set of vertices is V, and its set of edges contains all edges of G_1 and all edges of G_2.

Armed with these preparations, we can now outline the proof of Theorem 7.5. Its first part is an easy application of Turán's Theorem for bipartite graphs. The proof of the second part is more interesting. Suppose all bipartite graphs satisfy $\mathcal{P}$, and let $r+1$ (≥ 3) be the minimum chromatic number of a graph that does not satisfy this property. Fix a graph H of chromatic number $r+1$ that does not satisfy $\mathcal{P}$ and let μ be the constant of Theorem 7.30. Consider, first, the case $r \geq 3$. In this case we show that any efficient algorithm that approximates $E'_{\mathcal{P}}(G)$ up to $n^{2-\delta}$ will enable us to decide efficiently if a given input graph $F = (V(F), E(F))$ is r-colorable. Indeed, given such an F on m vertices, let $b = m^c$ where c is large constant, to be chosen appropriately. Let F_b be the b-blowup of F, and let F' be the vertex disjoint union of r copies of F_b. Let G' be the (n, d, λ)-graph with $d = (1-\mu)n$ and $\lambda \leq \sqrt{n}$, whose number of vertices n, is at least the number of vertices of F', and not more than four times of that, and identify the vertices of F' with some of those of G'. Let $G = G' \cup F'$ be the Boolean Or of these two graphs. If F is r-colorable, then so is its blowup F_b, and hence in this case F' has a proper r-coloring in which all color classes have the same size. This can be extended to a partition of the vertices of G to r nearly equal color classes, providing an r-colorable subgraph of G (which satisfies $\mathcal{P}$ by our choice of r) that contains all edges of F', and some edges of G' that do not belong to F'. The pseudo-random properties of G' enable us to approximate this number well.

On the other hand, if F is not r-colorable, then any r-colorable subgraph of G misses at least $b^2 r$ edges of F', and, by the pseudo-random properties of G' cannot contain too many edges of this graph that do not belong to F'. With the right choice of c, this will ensure that if we can approximate the number of edges in a maximum r-colorable subgraph of G up to an $n^{2-\delta}$-additive error, this will enable us to know for sure whether F is r-colorable or not. However, by Theorem 7.30, and as the minimum degree of our graph is at least $(1-\mu)n$, the maximum size of an H-free subgraph of G is very close to the maximum size of an r-colorable subgraph of it, which is therefore also very close to the maximum number of edges in a subgraph of G satisfying $\mathcal{P}$. This implies that approximating well this last quantity is NP-hard. The case $r = 2$ is similar, but here we have to use that the MAX-CUT problem is NP-hard. The full details appear in Section 7.8.

7.7 Proof of Theorem 7.30

Throughout this section we will assume that the number of vertices n in our graph is sufficiently large. We first prove the first part of Theorem 7.30, which is an extension of Turán's theorem. To this end, we need a result proved for K_{r+1}-free graphs by Andrásfai, Erdős and Sós [21] and in a more general form by Erdős and Simonovits [56].

Theorem 7.31. ([21],[56]) *Let H be a fixed graph with chromatic number $r+1 \geq 3$ which contains an edge e such that $\chi(H-e) = r$. If G is an H-free graph of order n with minimal degree $\delta(G) > \frac{3r-4}{3r-1}n$ then G is r-colorable.*

We will also need the following simple lemma.

Lemma 7.32. *Let $r \geq 2$ be an integer and suppose G' is an r-partite subgraph of a graph G (which may be empty) such that there are m edges incident to the vertices in $V(G)\backslash V(G')$. Then G has an r-partite subgraph of size at least $e(G') + \frac{r-1}{r}m$.*

Proof: Let $(A'_1, \ldots, A'_r)$ be the partition of G'. Consider an r-partite subgraph Γ of G with parts $(A_1, \ldots, A_r)$ such that $A'_i \subset A_i$ for every i, where we place each vertex $v \in V(G)\backslash V(G')$ in A_i randomly and independently with probability $1/r$. All edges of G' are edges of Γ, and each edge incident to a vertex in $V(G)\backslash V(G')$ appears in Γ with probability $\frac{r-1}{r}$. By linearity of expectation $\mathbb{E}\big[e(\Gamma)\big] = e(G') + \frac{r-1}{r}m$, so some r-partite subgraph of G has at least this many edges. $\square$

In particular, by taking G' to be the empty graph we obtain that every G contains an r-partite subgraph of size at least $\frac{r-1}{r}e(G)$.

Proof of Theorem 7.30 part (i): We prove that $E'_H(G) = E'_r(G)$ for all graphs G on n vertices with minimum degree

$$\delta(G) \geq \left(1 - \frac{3}{4(r-1)(3r-1)}\right)n + 1.$$

Let Γ be the largest (in terms of number of edges) r-partite subgraph of G and let F be the largest H-free subgraph of G. To prove the first part of the theorem one needs to show that $e(F) = e(\Gamma)$. As H is not r-colorable we trivially have $e(F) \geq e(\Gamma)$. In the rest of the proof we establish that $e(\Gamma) \geq e(F)$. First, note that by Lemma 7.32 we have

$$e(\Gamma) \geq \frac{r-1}{r}e(G) = \frac{r-1}{r}\left(\left(1 - \frac{3}{4(r-1)(3r-1)}\right)n + 1\right)n/2 = \frac{12r^2-16r+1}{8r(3r-1)}n^2 + \frac{r-1}{2r}n.$$

If F has a vertex of degree at most $\frac{3r-4}{3r-1}n$ we delete it and continue. We construct a sequence of graphs $F = F_n, F_{n-1}, ...$, where if F_k has a vertex of degree $\leq \frac{3r-4}{3r-1}k$ we delete that vertex to obtain F_{k-1}. Let F' be the final graph of this sequence which has s vertices and minimal degree greater than $\frac{3r-4}{3r-1}s$. Since F' is H-free, by Theorem 7.31, it is r-partite. Therefore

we have that

$$\begin{aligned}\frac{r-1}{2r}s^2 &\geq e(F') \geq e(F) - \frac{3r-4}{3r-1}\left(\binom{n+1}{2} - \binom{s+1}{2}\right)\\ &\geq e(\Gamma) - \frac{3r-4}{2(3r-1)}(n^2-s^2) - \frac{3r-4}{2(3r-1)}n\\ &\geq \frac{12r^2-16r+1}{8r(3r-1)}n^2 - \frac{3r-4}{2(3r-1)}(n^2-s^2).\end{aligned}$$

This implies that $\frac{s^2}{2r(3r-1)} \geq \frac{n^2}{8r(3r-1)}$ and so $s \geq n/2$.

Let X be the set of $n-s$ vertices which we deleted, i.e., $X = V(G) - V(F')$. By the minimal degree assumption there are at least

$$m \geq \delta(G)|X| - \binom{|X|}{2} \geq \frac{12r^2-16r+1}{4(r-1)(3r-1)}n(n-s) + (n-s) - \frac{(n-s)^2}{2}$$

edges incident with vertices in X. Thus, by Lemma 7.32, the size of the largest r-partite subgraph of G is at least

$$\begin{aligned}e(\Gamma) &\geq e(F') + \frac{r-1}{r}m \geq e(F) - \frac{3r-4}{3r-1}\left(\binom{n+1}{2} - \binom{s+1}{2}\right) + \frac{r-1}{r}m\\ &= e(F) - \frac{3r-4}{2(3r-1)}(n^2-s^2) - \frac{3r-4}{2(3r-1)}(n-s) + \frac{r-1}{r}m\\ &\geq e(F) - \frac{3r-4}{2(3r-1)}(n^2-s^2) + \frac{r-1}{r}\left(\frac{12r^2-16r+1}{4(r-1)(3r-1)}n(n-s) - \frac{(n-s)^2}{2}\right)\\ &= e(F) + \frac{(n-s)(2s-n)}{4r(3r-1)} \geq e(F).\end{aligned}$$

This implies that $e(\Gamma) \geq e(F)$ and completes the proof. □

We turn to prove Theorem 7.30 part (ii). To this end, we first prove the main technical result of this section, Theorem 7.33 below, which is a version of Theorem 7.31 that applies to arbitrary graphs H. We then apply this theorem in order to prove Theorem 7.30 part (ii). The reader may want to note that this application of Theorem 7.33 is similar to the way we applied Theorem 7.31 in order to prove the first part of Theorem 7.30.

Theorem 7.33. *Let H be a fixed graph on h vertices with chromatic number $r+1 \geq 3$ and let G be an H-free graph of order n with minimum degree $\delta(G) \geq \left(\frac{r-1}{r} - \frac{1}{3hr^2}\right)n$. Then one can delete at most $O\left(n^{2-(r+1)/h}\right)$ edges to make G r-colorable.*

Proof: First we need the following weaker bound on $E'_r(G)$.

Claim 7.34. *G can be made r-partite by deleting $o(n^2)$ edges.*

Proof: We use the Regularity Lemma given in Lemma 7.11. For every constant $0 < \eta < \frac{1}{12hr^2}$ let $\gamma = \gamma_{7.8}(\eta, r+1, h) < \eta^2$ be sufficiently small to guarantee that the assertion of

Lemma 7.8 holds[8]. Consider a γ-regular partition $(U_1, U_2, \ldots U_k)$ of G. Let G' be a new graph on the vertices $1 \leq i \leq k$ in which (i,j) is an edge iff (U_i, U_j) is a γ-regular pair with density at least η. Since G is an H-free graph and H is homomorphic to K_{r+1} (as $\chi(H) = r+1$), by Lemma 7.8, G' contains no clique of size $r+1$. Call a vertex of G' *good* if there are at most ηk other vertices j such that the pair (U_i, U_j) is not γ-regular, otherwise call it *bad*. Since the number of non-regular pairs is at most $\gamma\binom{k}{2} \leq \eta^2 k^2/2$ we have that all but at most ηk vertices are good. By definition, the degree of each good vertex in G' is at least $\left(\frac{r-1}{r} - \frac{1}{3hr^2}\right)k - 2\eta k - 1$, since deletion of the edges from non-regular pairs and sparse pairs can decrease the degree by at most ηk each and the deletion of edges inside the sets U_i can decrease it by 1. By deleting all bad vertices we obtain a K_{r+1}-free graph on at most k vertices with minimal degree at least

$$\left(\frac{r-1}{r} - \frac{1}{3hr^2}\right)k - 3\eta k - 1 \geq \left(\frac{r-1}{r} - \frac{2}{3hr^2}\right)k \geq \left(\frac{r-1}{r} - \frac{1}{3r^2}\right)k > \frac{3r-4}{3r-1}k.$$

Therefore, by Theorem 7.31, this graph is r-partite. This implies that to make G r-partite we can delete at most $\gamma n^2 + \eta n^2 + (\eta n) \cdot n + k \cdot (n/k)^2 \leq 3\eta n^2 + n^2/k = o(n^2)$ edges. □

Consider a partition $(V_1, \ldots, V_r)$ of the vertices of G into r parts which maximizes the number of crossing edges between the parts. Then for every $x \in V_i$ and $j \neq i$ the number of neighbors of x in V_i is at most the number of its neighbors in V_j, as otherwise by shifting x to V_j we increase the number of crossing edges. By Claim 7.34, we have that this partition satisfies that $\sum_i e(V_i) = o(n^2)$. Call a vertex x of G *typical* if $x \in V_i$ and has at most $n/(10hr^2)$ neighbors in V_i. Note that there are at most $o(n)$ non-typical vertices in G and, in particular, every part V_i contains a typical vertex. By definition, the degree of this vertex outside V_i is at least $\left(\frac{r-1}{r} - \frac{1}{3hr^2}\right)n - \frac{n}{10hr^2} > \left(\frac{r-1}{r} - \frac{1}{2hr^2}\right)n$ and at most $n - |V_i|$. Therefore $|V_i| \leq (\frac{1}{r} + \frac{1}{2hr^2})n$. Also note that the number of neighbors in V_i of every typical vertex $x \in V_j, j \neq i$ is at least

$$\begin{aligned} d_{V_i}(x) &\geq d(x) - d_{V_j}(x) - (r-2)\max_k |V_k| \\ &\geq \left(\frac{r-1}{r} - \frac{1}{3hr^2}\right)n - \frac{n}{10hr^2} - (r-2)\left(\frac{1}{r} + \frac{1}{2hr^2}\right)n \\ &> \left(\frac{1}{r} - \frac{r-1}{2hr^2}\right)n. \end{aligned} \tag{7.11}$$

The next claim is an immediate corollary of the above observation.

Claim 7.35. *Let U be a subset of V_j of size at least $(\frac{1}{2r} - \frac{1}{4hr})n$ and let $y_1, \ldots, y_k$ be an arbitrary set of $k \leq r-1$ typical vertices outside V_j. Then, there are at least $\frac{n}{2r(r+1)}$ vertices in U, which are adjacent to all vertices y_i.*

Proof: By definition, there are at most $|V_j| - d_{V_j}(y_i)$ non-neighbors of y_i in V_j and thus there are at most that many vertices in U not adjacent to y_i. Delete from U any vertex,

[8]Recall that by Remark 7.9 we may assume that $\gamma_{7.8}(\eta, r+1, h) < \eta^2$.

which is not a neighbor of either $y_1, y_2, \ldots, y_k$. The remaining set is adjacent to every vertex y_i and has size at least

$$|U| - \sum_i \big(|V_j| - d_{V_j}(y_i)\big).$$

Since by (7.11) the degree in V_j of every typical vertex $y_i \notin V_j$ is at least $d_{V_j}(y_i) \geq (\frac{1}{r} - \frac{r-1}{2hr^2})n$, we obtain that the number of common neighbors of $y_1, \ldots, y_k$ in U is at least

$$\begin{aligned}
|U| - \sum_i \big(|V_j| - d_{V_j}(y_i)\big) &\geq k\left(\frac{1}{r} - \frac{r-1}{2hr^2}\right)n - k|V_j| + |U| \\
&\geq k\left(\frac{1}{r} - \frac{r-1}{2hr^2}\right)n - k\left(\frac{1}{r} + \frac{1}{2hr^2}\right)n + |U| \\
&\geq |U| - \frac{k}{2hr}n \geq \left(\frac{1}{2r} - \frac{1}{4hr}\right)n - \frac{k}{2hr}n \\
&\geq \left(\frac{1}{2r} - \frac{k+1}{2hr}\right)n \geq \frac{n}{2r} - \frac{n}{2h} \geq \frac{n}{2r(r+1)}.
\end{aligned}$$

Here we used that $k+1 \leq r$ and $h \geq r+1$. □

Claim 7.36. *For every non-typical vertex $x \in V_i$ there are at least $\frac{n^r}{5h(3r^2)^r}$ r-cliques $y_1, \ldots, y_r$ such that $y_j \in V_j$ for all $1 \leq j \leq r$ and all vertices y_j are adjacent to x.*

Proof: Without loss of generality let $i = 1$ and let $x \in V_1$ be a non-typical vertex. Since for every $j \neq 1$ the number of neighbors of x in V_j is at least as large as the number of its neighbors in V_1 we have that

$$\begin{aligned}
d_{V_j}(x) &\geq \frac{d_{V_j}(x) + d_{V_1}(x)}{2} \geq \frac{1}{2}\left(\Big(\frac{r-1}{r} - \frac{1}{3hr^2}\Big)n - (r-2)\max_i |V_i|\right) \\
&\geq \frac{1}{2}\left(\Big(\frac{r-1}{r} - \frac{1}{3hr^2}\Big)n - (r-2)\Big(\frac{1}{r} + \frac{1}{2hr^2}\Big)n\right) \\
&\geq \left(\frac{1}{2r} - \frac{1}{4hr}\right)n. \qquad (7.12)
\end{aligned}$$

To construct the r-cliques satisfying the assertion of the claim, first observe, that since x is non-typical it has at least $n/(10hr^2)$ neighbors in V_1 and at least $n/(10hr^2) - o(n) > n/(15hr^2)$ of these neighbors are typical. Choose y_1 to be an arbitrary typical neighbor of x in V_1 and continue. Suppose at step $1 \leq k \leq r-1$ we already have a k-clique $y_1, \ldots, y_k$ such that $y_i \in V_i$ for all i and all vertices y_i are adjacent to x. Let U_{k+1} be the set of neighbors of x in V_{k+1}. Then, by (7.12) we have that $|U_{k+1}| = d_{V_{k+1}}(x) \geq (\frac{1}{2r} - \frac{1}{4hr})n$ and therefore by Claim 7.35 there are at least $\frac{n}{2r(r+1)}$ common neighbors of the vertices y_i in U_{k+1}. Moreover, at least $\frac{n}{2r(r+1)} - o(n) > \frac{n}{3r^2}$ of them are typical and we can choose y_{k+1} to be any of them. Therefore at the end of the process we indeed obtained at least $\frac{n}{15hr^2}(\frac{n}{3r^2})^{r-1} = \frac{n^r}{5h(3r^2)^r}$ r-cliques with the desired property. □

Claim 7.37. *Each V_i contains at most $O(1)$ non-typical vertices.*

Proof: Suppose that the number of non-typical vertices in V_i is at least $5h^2(3r^2)^r$. Consider an auxiliary bipartite graph F with parts W_1, W_2, where W_1 is the set of some $t = 5h^2(3r^2)^r$ non-typical vertices in V_i, W_2 is the family of all n^r r-element multi-sets of $V(G)$ such that $x \in W_1$ is adjacent to multi-set Y from W_2 iff Y is an r-clique in G with exactly one vertex in every V_j and all vertices of Y are adjacent to x. By the previous claim, F has at least $e(F) \geq t\frac{n^r}{5h(3r^2)^r} = hn^r$ edges and therefore the average degree of a vertex in W_2 is at least $d_{av} = e(F)/|W_2| = e(F)/n^r \geq h$. By the convexity of the function $f(z) = \binom{z}{h}$, we can find h vertices $x_1, \ldots, x_h$ in W_1 such that the number of their common neighbors in W_2 is at least

$$m \geq \frac{\sum_{Y \in W_2} \binom{d(Y)}{h}}{\binom{t}{h}} \geq n^r \frac{\binom{d_{av}}{h}}{t^h} = \Omega\big(n^r\big).$$

Thus we proved that G contains h vertices $X = \{x_1, \ldots, x_h\}$ and a family of r-cliques $\mathcal{C}$ of size $m = \Omega\big(n^r\big)$ such that every clique in $\mathcal{C}$ is adjacent to all vertices in X. Next we need the following well-known lemma which appears first implicitly in Erdős [52] (see also, e.g., [71]). It states that if an r-uniform hypergraph on n vertices has $m = \Omega\big(n^r\big)$ edges, then it contains a complete r-partite r-uniform hypergraph with parts of size h. By applying this statement to $\mathcal{C}$, we conclude that there are r disjoint set of vertices $A_1, \ldots, A_r$ each of size h such that every r-tuple $a_1, \ldots, a_r$ with $a_i \in A_i$ forms a clique which is adjacent to all vertices in X. The restriction of G to $X, A_1, \ldots, A_r$ forms a complete $(r+1)$-partite graph with parts of size h each, which clearly contains H. This contradiction shows that there are less than $5h^2(3r^2)^r = O(1)$ non-typical vertices in V_i and completes the proof of the claim. □

Having finished all the necessary preparations, we are now ready to complete the proof of Theorem 7.33. Let $h_1 \leq h_2 \leq \ldots \leq h_{r+1}$ be the sizes of the color-classes in an $r+1$ coloring of H. Clearly $h_1 \leq h/(r+1)$. Without loss of generality, suppose that V_1 spans at least $2hn^{2-(r+1)/h}$ edges. By the previous claim, only at most $O(n)$ of these edges are incident to non-typical vertices. Therefore the set of typical vertices in V_1 spans at least $hn^{2-(r+1)/h}$ edges. Then, by the well known result of Kövari, Sós and Turán [91] about Turán numbers of bipartite graphs, V_1 contains a complete bipartite graph $H_1 = K_{h_1,h_2}$ all of whose vertices are typical. If there are at least h_3 typical vertices in V_2 which are adjacent to all vertices of H_1 then we add them to H_1 to form a complete 3-partite graph H_2 with parts of sizes h_1, h_2 and h_3 and continue. We claim that if at step $1 \leq k \leq r-1$ there is a $k+1$-partite graph $H_k \subset \cup_{i=1}^{k} V_i$ with parts of sizes $h_1, \ldots, h_{k+1}$ all of whose vertices are typical, then we can extend it to the complete $k+2$-partite graph H_{k+1} by adding h_{k+2} typical vertices from V_{k+1} which are adjacent to all vertices of H_k. Indeed, recall that by (7.11) the number of neighbors in V_{k+1} of every typical vertex $x \in V_i, i \neq k+1$ is at least $d_{V_{k+1}}(x) \geq (\frac{1}{r} - \frac{r-1}{2hr^2})n$. Let $t \leq h$ be the order of H_k. Then, as in Claim 7.35 the number

of vertices in V_{k+1} which are adjacent to all vertices of H_k is at least

$$\begin{aligned} |V_{k+1}| - t\left(|V_{k+1}| - \left(\frac{1}{r} - \frac{r-1}{2hr^2}\right)n\right) &\geq t\left(\frac{1}{r} - \frac{r-1}{2hr^2}\right)n - (t-1)\left(\frac{1}{r} + \frac{1}{2hr^2}\right)n \\ &= \frac{n}{r} - \frac{t(r-1)+t-1}{2hr^2}n \\ &\geq \frac{n}{r} - \frac{t}{2hr}n \geq \frac{n}{r} - \frac{n}{2r} = \frac{n}{2r} \end{aligned}$$

and thus at least $n/(2r) - O(1) > h_{k+2}$ of these vertices are typical. Continuing the above process $r-1$ steps we obtain a complete $(r+1)$-partite graph with parts of sizes $h_1, \ldots, h_{r+1}$, which clearly contains H. This contradicts our assumption that G is H-free and shows that every V_i spans at most $O\big(n^{2-(r+1)/h}\big)$ edges. Therefore the number of edges we need to delete to make G r-partite is bounded by $\sum_i e(V_i) \leq O\big(n^{2-(r+1)/h}\big)$. This completes the proof. □

Proof of Theorem 7.30 part (ii): Let H be a fixed graph on h vertices with chromatic number $r+1 \geq 3$. We show that the constants $\gamma(H)$ and $\mu(H)$ in the assertion of the theorem can be chosen to be $(r+1)/h$ and $1/(4hr^2)$ respectively. Let G be an H-free graph of order n with minimal degree $\delta(G) \geq (1 - \frac{1}{4hr^2})n$ and let Γ be the largest r-partite subgraph of G and F be a largest H-free subgraph of G. To prove the second item of the theorem it is enough to show that $e(\Gamma) \leq e(F) \leq e(\Gamma) + O(n^{2-(r+1)/h})$. As H is not r-colorable we trivially have $e(\Gamma) \leq e(F)$. In the rest of the proof we establish that $e(F) \leq e(\Gamma) + O(n^{2-(r+1)/h})$. By Lemma 7.32 we have that

$$e(\Gamma) \geq \frac{r-1}{r}e(G) = \frac{r-1}{r}\left(1 - \frac{1}{4hr^2}\right)n^2/2 = \left(\frac{r-1}{2r} - \frac{r-1}{8hr^3}\right)n^2.$$

If F has a vertex of degree at most $(\frac{r-1}{r} - \frac{1}{3hr^2})n$ we delete it and continue. We construct a sequence of graphs $F = F_n, F_{n-1}, ...$, where if F_k has a vertex of degree $\leq (\frac{r-1}{r} - \frac{1}{3hr^2})k$ we delete that vertex to obtain F_{k-1}. Let F' be the final graph of this sequence which has s vertices and minimal degree greater than $(\frac{r-1}{r} - \frac{1}{3hr^2})s$ and let Γ' be the largest r-partite subgraph of F'. Since F' is H-free, Theorem 7.33 implies $e(F') \leq e(\Gamma') + O(n^{2-(r+1)/h})$. Therefore we have that

$$\begin{aligned} \frac{r-1}{2r}s^2 + o(n^2) &\geq e(F') \geq e(F) - \left(\frac{r-1}{r} - \frac{1}{3hr^2}\right)\left(\binom{n+1}{2} - \binom{s+1}{2}\right) \\ &\geq e(\Gamma) - \left(\frac{r-1}{2r} - \frac{1}{6hr^2}\right)(n^2 - s^2) - O(n) \\ &\geq \left(\frac{r-1}{2r} - \frac{r-1}{8hr^3}\right)n^2 - \left(\frac{r-1}{2r} - \frac{1}{6hr^2}\right)(n^2 - s^2) - o(n^2). \end{aligned}$$

This implies that

$$\frac{s^2}{6hr^2} \geq \left(\frac{1}{6hr^2} - \frac{r-1}{8hr^3}\right) n^2 - o(n^2) > \left(\frac{1}{6hr^2} - \frac{1}{8hr^2}\right) n^2 = \frac{n^2}{24hr^2}$$

and so $s \geq n/2$.

Let X be the set of $n-s$ vertices which we deleted, i.e., $X = V(G) - V(F')$. By the minimal degree assumption there are at least

$$m \geq \delta(G)|X| - \binom{|X|}{2} \geq \left(1 - \frac{1}{4hr^2}\right) n(n-s) - \frac{(n-s)^2}{2} = (n-s)\left(\left(\frac{1}{2} - \frac{1}{4hr^2}\right)n + \frac{s}{2}\right)$$

edges incident with vertices in X. Thus, by Lemma 7.32, the size of the largest r-partite subgraph of G is at least

$$\begin{aligned}
e(\Gamma) &\geq e(\Gamma') + \frac{r-1}{r}m \geq e(F') - O\big(n^{2-(r+1)/h}\big) + \frac{r-1}{r}m \\
&\geq e(F) - \left(\frac{r-1}{r} - \frac{1}{3hr^2}\right)\left(\binom{n+1}{2} - \binom{s+1}{2}\right) + \frac{r-1}{r}m - O\big(n^{2-(r+1)/h}\big) \\
&\geq e(F) - \left(\frac{r-1}{2r} - \frac{1}{6hr^2}\right)(n^2 - s^2) + \frac{r-1}{r}m - O\big(n^{2-(r+1)/h}\big) \\
&\geq e(F) - \left(\frac{r-1}{2r} - \frac{1}{6hr^2}\right)(n^2 - s^2) + (n-s)\left(\left(\frac{r-1}{2r} - \frac{r-1}{4hr^3}\right)n + \frac{(r-1)s}{2r}\right) - O\big(n^{2-\frac{r+1}{h}}\big) \\
&= e(F) + \frac{(n-s)(2s - \frac{r-3}{r}n)}{12hr^2} - O\big(n^{2-(r+1)/h}\big) \geq e(F) - O\big(n^{2-(r+1)/h}\big). \qquad \square
\end{aligned}$$

7.8 Proof of Hardness Result

We start with the proof of the first part of Theorem 7.5. If there is a bipartite graph H that does not satisfy $\mathcal{P}$, then, by the known results about the Turán numbers of bipartite graphs proved in [91], there exists a positive $\delta > 0$ such that for any large n, any graph with n vertices and at least $n^{2-\delta}$ edges contains a copy of H. Thus, given a graph G on n vertices, one must delete all its edges besides, possibly, $n^{2-\delta}$ of them, to obtain a subgraph satisfying $\mathcal{P}$. As certainly the edgeless graph satisfies $\mathcal{P}$, this provides the required approximation in this case.

The proof of the second part is more complicated, and requires all the preparations obtained in the previous section. Suppose all bipartite graphs satisfy $\mathcal{P}$, and let $r+1 \geq 3$ be the minimum chromatic number of a graph that does not satisfy this property. Fix a graph H of chromatic number $r+1$ that does not satisfy $\mathcal{P}$. We will show that any efficient algorithm that approximates $E'_{\mathcal{P}}(G)$ up to $n^{2-\delta}$ will enable us to decide efficiently how many edges we need to delete from a given input graph $F = (V(F), E(F))$ to make it r-partite. For $r \geq 3$ this problem contains the r-colorability problem, and for $r = 2$ it is the MAX-CUT problem and therefore it is NP-hard for every $r \geq 2$.

Given a graph F on m vertices such that we need to delete ℓ edges to make it r-partite, let $b = m^c$ where c is a large constant, to be chosen later. Let F_b be the b-blowup of F, and let F' be the vertex disjoint union of r copies of F_b. Let $\mu = \mu(H)$ be the constant from Theorem 7.30 and let G' be the (n, d, λ)-graph with $d = (1-\mu)n$ and $\lambda \leq \sqrt{n}$, described in Section 7.6. As the integer q in the construction discussed in Section 7.6 can be a prime power, we can always choose the number of vertices of G', which is q^2, to be at least the number of vertices of F', and not more than 4 times of that. In particular, we have $n = \Theta(rmb) = \Theta\big(m^{c+1}\big)$. Identify the vertices of F' with some of those of G'. Let $G = G' \cup F'$ be the Boolean Or of these two graphs.

Suppose, that instead of adding to F' a pseudo-random graph G', we would put any non-edge of F' in G with probability $1-\mu$. It is easy to see that in this case the expected number of edges, which would be spanned by a set of a vertices that span t edges in F', would be $(1-\mu)\binom{a}{2} + \mu t$. The following claim establishes that this is approximately what we find when we add to F' a pseudo-random graph. We then use this claim to show that we can also estimate $E'_r(G)$ as a function of $\ell = E'_r(F)$.

Claim 7.38. *Let A be a subset of the vertices of G of size a which contains precisely t edges of F'. Then the number of edges of G in A satisfies*

$$(1-\mu)\frac{a^2}{2} + \mu t - O\big(m^2 n^{3/2}\big) \leq e_G(A) \leq (1-\mu)\frac{a^2}{2} + \mu t + O\big(m^2 n^{3/2}\big).$$

Proof: By construction, the edges of the subgraph of F' induced on the set A form an edge disjoint union of complete bipartite graphs we denote by $\Gamma_i = (U_i, W_i), 1 \leq i \leq k$. Thus $\sum_i |U_i||W_i| = t$ and the fact that F' is a blowup of r disjoint copies of F, which altogether have rm vertices and at most $r\binom{m}{2}$ edges, implies that $k \leq r\binom{m}{2} < rm^2$. The number of edges of G spanned on A is the number of edges of G' inside A, minus the number of edges of G' spanned by the pairs (U_i, W_i), plus the number of edges of F' inside A. To estimate this quantity, we need the well-known fact (see, e.g, Chapter 9 of [20]), that the number of edges between two subsets X, Y of an (n, d, λ)-graph G' satisfies

$$\left| e(X,Y) - \frac{|X||Y|d}{n} \right| \leq \lambda\sqrt{|X||Y|}$$

and the fact that in such a graph $\left|e(X) - \frac{d|X|^2}{2n}\right| \leq \lambda|X|$. Therefore we obtain that

$$\begin{aligned}
e_G(A) &= e_{G'}(A) - \sum_{i=1}^{k} e_{G'}(U_i, W_i) + t \\
&= e_{G'}(A) + \sum_{i=1}^{k} \Big(|U_i||W_i| - e_{G'}(U_i, W_i) \Big) \\
&\geq \frac{d|A|^2}{2n} - \lambda|A| + \sum_{i=1}^{k} \left(|U_i||W_i| - \frac{d}{n}|U_i||W_i| - \lambda\sqrt{|U_i||W_i|} \right) \\
&\geq \frac{d|A|^2}{2n} - \lambda n + \sum_{i=1}^{k} \big(\mu|U_i||W_i| - \lambda n \big) \\
&= (1-\mu)\frac{a^2}{2} + \mu \sum_{i=1}^{k} |U_i||W_i| - (k+1)\lambda n \\
&= (1-\mu)\frac{a^2}{2} + \mu t - O\big(m^2 n^{3/2}\big).
\end{aligned}$$

The upper bound $e_G(A) \leq (1-\mu)\frac{a^2}{2} + \mu t + O\big(m^2 n^{3/2}\big)$ can be obtained similarly. □

Recall that the b-blowup F_b of a graph F, defined in Section 7.6, is the graph obtained from F by replacing every vertex $v \in V(F)$ with an independent set I_v, of size b, and by replacing every edge $(u, v) \in E(F)$, with a complete bipartite graph, whose partition classes are the independent sets I_u and I_v.

Claim 7.39. *For any graph F and any integer b, we have $E'_r(F_b) = b^2 E'_r(F)$.*

Proof: We start by showing that $E'_r(F_b) \leq b^2 E'_r(F)$. Suppose S is a set of $E'_r(F)$ edges whose removal turns F into an r-colorable graph F'. Suppose we remove from F_b all the edges connecting I_u and I_v for any $(u, v) \in S$. Note, that we thus remove $b^2 E'_r(F)$ edges from F_b. We claim that the resulting graph F'_b is r-colorable. Indeed, let $c : V(F) \mapsto \{1, \ldots, r\}$ be a r-coloring of F' and note that by definition of F'_b, if we color all the vertices of I_v with the color $c(v)$, we get a legal r-coloring of F'. Therefore $E'_r(F_b) \leq b^2 E'_r(F)$.

To see that $E'_r(F_b) \geq b^2 E'_r(F)$, let S be a set of edges whose removal turns F_b into an r-colorable graph, and suppose for every $v \in V(F)$ we randomly pick a single vertex from each of the sets I_v. For every edge of S, the probability that we picked both of its endpoints is b^{-2}, therefore the expected number of edges spanned by these vertices is $|S|/b^2$. As the removal of the edges of S makes F_b r-colorable, this in particular applies to all of its subgraphs. Note, that for any choice of a single vertex from each of the independent sets I_v, the graph they span is isomorphic to F. Thus, any such choice spans at least $E'_r(F)$ of the edges of S. It thus must be the case that $|S|/b^2 \geq E'_r(F_b)$, and the proof is complete. □

Claim 7.40. *The graph G satisfies*

$$\left| E'_r(G) - \left((1-\mu)\frac{n^2}{2r} + \mu r \ell b^2 \right) \right| \leq O(m^2 n^3). \tag{7.13}$$

Proof: Fix a partition of F into r parts which misses exactly ℓ edges and consider r disjoint copies of F. By taking appropriately different parts in every copy of F we can partition this new graph into r equal parts such that exactly $r\ell$ edges are non-crossing. Since F' is a b-blowup of r disjoint copies of F, this gives a partition of F' into equal parts which misses $r\ell b^2$ edges. We can extend this to a partition of G into r nearly equal sets $V(G) = V_1 \cup \ldots \cup V_r$ which misses exactly $r\ell b^2$ edges of F'. Let t_i be the number of edges of F' inside V_i, then $\sum_i t_i = r\ell b^2$. This, together with Claim 7.38, implies that it is enough to delete at most

$$\begin{aligned}
\sum_{i=1}^{r} e_G(V_i) &\leq \sum_{i=1}^{r} \left((1-\mu)\frac{|V_i|^2}{2} + \mu t_i + O\big(m^2 n^{3/2}\big) \right) \\
&\leq (1-\mu) r \frac{(n/r+1)^2}{2} + \mu \sum_{i=1}^{r} t_i + O\big(m^2 n^{3/2}\big) \\
&= (1-\mu)\frac{n^2}{2r} + \mu r \ell b^2 + O\big(m^2 n^{3/2}\big).
\end{aligned}$$

edges to make G r-partite and hence to satisfy property $\mathcal{P}$.

On the other hand, by Claim 7.39, any partition of F', which is b-blowup of r disjoint copies of F, into r parts misses at least $r\ell b^2$ edges. Therefore for every partition of the vertices of G into r sets there are at least $r\ell b^2$ edges of F' which are non-crossing. Let $V_1 \cup \ldots \cup V_r$ be a partition of $V(G)$ that maximizes the number of crossing edges and let again t_i be the number of edges of F' inside V_i (note that in this case the sets V_i are not necessarily of the same size). Using Claim 7.38, together with the fact that $\sum_i t_i \geq r\ell b^2$ and the Cauchy-Schwartz inequality, we conclude that

$$\begin{aligned}
\sum_{i=1}^{r} e_G(V_i) &\geq \sum_{i=1}^{r} \left((1-\mu)\frac{|V_i|^2}{2} + \mu t_i - O\big(m^2 n^{3/2}\big) \right) \\
&\geq \frac{1-\mu}{2} r \left(\frac{\sum_i |V_i|}{r} \right)^2 + \mu r \ell b^2 - O\big(m^2 n^{3/2}\big) \\
&= (1-\mu)\frac{n^2}{2r} + \mu r \ell b^2 - O\big(m^2 n^{3/2}\big).
\end{aligned}$$

This completes the proof of the claim. □

We are now ready to complete the proof of Theorem 7.5. Choose the constant c to be sufficiently large so that $2/(c+1) < \min(\delta, \gamma, 1/4)$. Recall, that as we chose $b = m^c$ and

$n = \Theta(m^{c+1})$, we have

$$n^{2-\delta} = o(b^2), \qquad n^{2-\gamma} = o(b^2), \qquad m^2 n^{3/2} = o(b^2). \tag{7.14}$$

Also, as G has minimum degree $(1-\mu)n$ we get from Theorem 7.30, that

$$E'_H(G) \geq E'_r(G) - O(n^{2-\gamma}). \tag{7.15}$$

As H does not satisfy $\mathcal{P}$ we clearly have $E'_{\mathcal{P}}(G) \geq E'_H(G)$. Combining this with (7.13), (7.14) and (7.15) we get

$$\begin{aligned} E'_{\mathcal{P}}(G) \geq E'_H(G) &\geq E'_r(G) - O(n^{2-\gamma}) \\ &\geq (1-\mu)\frac{n^2}{2r} + \mu r \ell b^2 - O\big(m^2 n^{3/2}\big) - O\big(n^{2-\gamma}\big) \\ &\geq (1-\mu)\frac{n^2}{2r} + \mu r \ell b^2 - o\big(b^2\big)\ . \end{aligned}$$

Furthermore, by our choice of r, we get that any r-colorable graph satisfies $\mathcal{P}$, hence we infer from (7.13) and (7.14) that

$$\begin{aligned} E'_{\mathcal{P}}(G) \leq E'_r(G) &\leq (1-\mu)\frac{n^2}{2r} + \mu r \ell b^2 + O\big(m^2 n^{3/2}\big) \\ &\leq (1-\mu)\frac{n^2}{2r} + \mu r \ell b^2 + o\big(b^2\big)\ . \end{aligned}$$

We thus conclude that $|E'_{\mathcal{P}}(G) - ((1-\mu)\frac{n^2}{2r} + \mu r \ell b^2)| \leq o(b^2)$. Therefore, if one can approximate $E'_{\mathcal{P}}(G)$ in time polynomial in n (and hence also in m) within an additive error of $n^{2-\delta} = o(b^2)$ then one thus efficiently computes an integer L, which is within an additive error of $o(b^2)$ from $(1-\mu)\frac{n^2}{2r} + \mu r \ell b^2$. But as in this case ℓ is precisely the nearest integer to $(L - (1-\mu)\frac{n^2}{2r})/\mu r b^2$, this implies that we can *precisely* compute the number of edge removals, needed in order to turn the input graph F into an r-partite graph. This implies that the problem of approximating $E'_{\mathcal{P}}(G)$ within $n^{2-\delta}$ is NP-hard, and completes the proof of Theorem 7.5.

7.9 Concluding Remarks and Open Problems

- We have shown that for any monotone graph property $\mathcal{P}$ and any $\epsilon > 0$ one can approximate efficiently the minimum number of edges that have to be deleted from an n-vertex input graph to get a graph that satisfies $\mathcal{P}$, up to an additive error of ϵn^2. Moreover, for any *dense* monotone property, that is, a property for which there are graphs on n vertices with $\Omega(n^2)$ edges that satisfy it, it is NP-hard to approximate

this minimum up to an additive error of $n^{2-\delta}$. It will be interesting to obtain similar sharp results for the case of sparse monotone properties. In some of these cases (like the property of containing no cycle, or the property of containing no vertex of degree at least 2) the above minimum can be computed precisely in polynomial time, and in some other cases, a few of which are treated in [24], [25], [115], a precise computation is known to be hard. Obtaining sharp estimates for the best approximation achievable efficiently seems difficult.

- As we have mentioned in Section 7.1, a special case of Theorem 7.5 implies that for any non-bipartite H, computing the smallest number of edge removals that are needed to make a graph H-free is NP-hard. This is clearly not the case for some bipartite graphs such as a single edge or any star. It will be interesting to classify the bipartite graphs for which this problem is NP-hard.

- It is natural to ask if the main results of this chapter can be extended to the larger family of hereditary properties, namely, properties closed under removal of vertices, but not necessarily under removal of edges. Many natural properties such as being Perfect, Chordal and Interval are hereditary non-monotone properties. By combining the ideas we used in order to prove Theorem 7.2 along with the main ideas of Chapter 1 it can probably be shown that Theorem 7.2 (as well as Theorem 7.3 and Corollary 7.4) also hold for any hereditary graph property. It seems interesting to decide if one can obtain a result analogous to Theorem 7.5 for the family of hereditary properties.

- A weaker version of Theorem 7.2 can be derived by combining the results of Chapter 1 and [64]. However, this only enables one to approximate $E_{\mathcal{P}}(G)$ within an additive error ϵ in time $n^{f(\epsilon)}$, while the running time of our algorithm is of type $f(\epsilon)n^2$.

- Recall that $E'_{\mathcal{F}}(G)$ denotes the smallest number of edge deletions that are needed in order to make G $\mathcal{F}$-free. For a family of graphs $\mathcal{F}$, let $\nu_{\mathcal{F}}(G)$ denote the $\mathcal{F}$-packing number of G, which is the size of the largest family of edge-disjoint copies of members of $\mathcal{F}$, which is spanned by G. Let $\nu^*_{\mathcal{F}}(G)$ denote the natural Linear Programming relaxation of $\nu_{\mathcal{F}}(G)$. Haxell and Rödl [82] and Yuster [116] have shown that $\nu_{\mathcal{F}}(G) \leq \nu^*_{\mathcal{F}}(G) \leq \nu_{\mathcal{F}}(G) + \epsilon n^2$ for any $\mathcal{F}$ and any $\epsilon > 0$, implying that for any finite $\mathcal{F}$, $\nu_{\mathcal{F}}(G)$ can be approximated within any additive error of ϵn^2 by solving the Linear Program for computing $\nu^*_{\mathcal{F}}(G)$. One may wonder whether it is possible to obtain Theorem 7.2 by solving the natural Linear Programming relaxation of $E'_{\mathcal{F}}(G)$, which we denote by $E^*_{\mathcal{F}}(G)$. Regretfully, this is not the case. Linear Programming duality implies that $E^*_{\mathcal{F}}(G) = \nu^*_{\mathcal{F}}(G)$ and by the results of [82] and [116] we thus have

$$\nu_{\mathcal{F}}(G) \leq E^*_{\mathcal{F}}(G) \leq \nu_{\mathcal{F}}(G) + \epsilon n^2 \ . \tag{7.16}$$

  Consider now any $\mathcal{F}$, which does not contain the single edge graph and note that we trivially have $\nu_{\mathcal{F}}(K_n) \leq \frac{1}{2}\binom{n}{2} \leq \frac{1}{4}n^2$ (we denote by K_n the n-vertex complete graph). If $\mathcal{F}$ contains a bipartite graph then by the theorem of Kövari, Sós and Turán (see

Section 7.6) we have $E'_{\mathcal{F}}(K_n) > \binom{n}{2} - n^{2-\delta} \geq (\frac{1}{2} - o(1))n^2$. If on the other hand all the graphs in $\mathcal{F}$ are of chromatic number $r \geq 3$ then clearly they all must contain at least $\binom{r}{2}$ edges, and therefore we must have $\nu_{\mathcal{F}}(K_n) \leq \binom{n}{2}/\binom{r}{2} \leq \frac{n^2}{r(r-1)}$. On the other hand, by the theorem of Erdős-Stone-Simonovits (see Section 7.6) $E'_{\mathcal{F}}(K_n) > \frac{n^2}{2(r-1)} - o(n^2)$. In any case, we have that $\nu_{\mathcal{F}}(K_n) + \delta n^2 \leq E'_{\mathcal{F}}(K_n)$ for some fixed $\delta = \delta(\mathcal{F}) > 0$. Combined with (7.16) we get that for any $\mathcal{F}$ not containing the single edge graph $E^*_{\mathcal{F}}(K_n) + \delta n^2 < E'_{\mathcal{F}}(K_n)$. Thus, the (trivial) case in which $\mathcal{F}$ contains a single edge is the only one for which computing $E^*_{\mathcal{F}}(G)$ is guaranteed to approximate $E'_{\mathcal{F}}(G)$ within ϵn^2 for any $\epsilon > 0$. In fact, in this degenerate case we actually have $E^*_{\mathcal{F}}(G) = E'_{\mathcal{F}}(G)$.

Bibliography

[1] N. Alon, Testing subgraphs in large graphs, Proc. 42^{nd} IEEE FOCS, IEEE (2001), 434-441. Also: Random Structures and Algorithms 21 (2002), 359-370.

[2] N. Alon, Ranking tournaments, SIAM J. Discrete Math. 20 (2006), 137-142.

[3] N. Alon, S. Dar, M. Parnas and D. Ron, Testing of clustering, Proc. 41 IEEE FOCS, IEEE (2000), 240-250.

[4] N. Alon, R. A. Duke, H. Lefmann, V. Rödl and R. Yuster, The algorithmic aspects of the Regularity Lemma, Proc. 33^{rd} IEEE FOCS, Pittsburgh, IEEE (1992), 473-481. Also: J. of Algorithms 16 (1994), 80-109.

[5] N. Alon, W. Fernandez de la Vega, R. Kannan and M. Karpinski, Random Sampling and Approximation of MAX-CSP Problems, Proc. of 34^{th} ACM STOC, ACM Press (2002), 232-239.

[6] N. Alon, E. Fischer, M. Krivelevich and M. Szegedy, Efficient testing of large graphs, Proc. of 40^{th} FOCS, New York, NY, IEEE (1999), 656–666. Also: Combinatorica 20 (2000), 451-476.

[7] N. Alon and M. Krivelevich, Testing k-colorability, SIAM J. Discrete Math., 15 (2002), 211-227.

[8] N. Alon, M. Krivelevich, I. Newman and M. Szegedy, Regular languages are testable with a constant number of queries, Proc. 40^{th} FOCS, New York, NY, IEEE (1999), 645–655. Also: SIAM J. on Computing 30 (2001), 1842-1862.

[9] N. Alon, M. Krivelevich and B. Sudakov, Turán numbers of bipartite graphs and related Ramsey-type questions, Combinatorics, Probability and Computing 12 (2003), 477-494.

[10] N. Alon and A. Shapira, Testing satisfiability, Proc. 13^{th} Annual ACM-SIAM SODA, ACM Press (2002), 645-654. Also: Journal of Algorithms, 47 (2003), 87-103.

[11] N. Alon and A. Shapira, Testing subgraphs in directed graphs, Proc. of the 35^{th} Annual Symp. on Theory of Computing (STOC), San Diego, California, 2003, 700–709. Also: JCSS 69 (2004), 354-382.

[12] N. Alon and A. Shapira, A characterization of easily testable induced subgraphs, Proc. of the 15^{th} Annual ACM-SIAM SODA, ACM Press (2004), 935-944. Also: Combinatorics, Probability and Computing, 15 (2006), 791-805.

[13] N. Alon and A. Shapira, Linear equation, arithmetic progressions and hypergraph property testing, Proc. of the 16^{th} Annual ACM-SIAM SODA, ACM Press (2005), 708-717. Also, Theory of Computing, Vol. 1 (2005), 177-216.

[14] N. Alon and A. Shapira, Every monotone graph property is testable, Proc. of the 37^{th} Annual Symp. on Theory of Computing (STOC), Baltimore, Maryland, 2005, 128-137.

[15] N. Alon and A. Shapira, A separation theorem in property-testing, manuscript, 2005.

[16] N. Alon and A. Shapira, A characterization of the (natural) graph properties testable with one-sided error, Proc. of 46^{th} Annual IEEE Symposium on Foundations of Computer Science (FOCS), 2005, Pittsburgh, Pennsylvania, 429-438.

[17] N. Alon, A. Shapira and B. Sudakov, Additive approximation for edge-deletion problems, Proc. of 46^{th} Annual IEEE Symposium on Foundations of Computer Science (FOCS), 2005, Pittsburgh, Pennsylvania, 419-428.

[18] N. Alon, E, Fischer, I. Newman and A. Shapira, A combinatorial characterization of the testable graph properties: it's all about regularity, Proc. of the 38^{th} Annual Symp. on Theory of Computing (STOC), 2006, 251-260.

[19] N. Alon and A. Shapira, On an extremal hypergraph problems of Brown, Erdős and Sós, Combinatorica, to appear.

[20] N. Alon and J. H. Spencer, **The Probabilistic Method**, Second Edition, Wiley, New York, 2000.

[21] B. Andrásfai, P. Erdős and V. Sós, On the connection between chromatic number, maximal clique and minimal degree of a graph, Discrete Math. 8 (1974), 205-218.

[22] S. Arora, A. Frieze and H. Kaplan, A new rounding procedure for the assignment problem with applications to dense graph arrangement problems, Proc. of 36^{th} FOCS (1996), 21-30. Also, Mathematical Programming 92:1 (2002), 1-36.

[23] S. Arora, D. Karger and M. Karpinski, Polynomial time approximation schemes for dense instances of graph problems, Proc. of 28^{th} STOC (1995). Also, JCSS 58 (1999), 193-210.

[24] T. Asano, An application of duality to edge-deletion problems, SIAM J. on Computing, 16 (1987), 312-331.

[25] T. Asano and T. Hirata, Edge-deletion and edge-contraction problems, Proc. of STOC (1982), 245-254.

[26] J. Balogh, B. Bollobás and D. Weinreich, Measures on monotone properties of graphs, Discrete Applied Mathematics, to appear.

[27] J. Balogh, P. Keevash and B. Sudakov On the minimal degree implying equality of the largest triangle-free and bipartite subgraphs, submitted.

[28] J. Bang-Jensen and P. Hell, The effect of two cycles on the complexity of colorings by directed graphs, Discrete Applied Math. 26 (1990), 1-23.

[29] F. A. Behrend, On sets of integers which contain no three terms in arithmetic progression, *Proc. National Academy of Sciences USA* 32 (1946), 331–332.

[30] M. Ben-Or, D. Coppersmith, M. Luby and R. Rubinfeld, Non-abelian homomorphism testing, and distributions close to their self-convolutions, Proc. of APPROX-RANDOM (2004), 273-285.

[31] E. Ben-Sasson, P. Harsha and S. Raskhodnikova, Some 3-CNF properties are hard to test, Proc. of STOC 2003, 345-354.

[32] M. Blum, M. Luby and R. Rubinfeld, Self-testing/correcting with applications to numerical problems, JCSS 47 (1993), 549-595.

[33] A. Bogdanov, K. Obata and L. Trevisan, A Lower Bound for Testing 3-Colorability in Bounded-degree Graphs, Proc. 43^{rd} IEEE FOCS, IEEE (2002), 93-102.

[34] B. Bollobás, **Extremal Graph Theory**, Academic Press, New York (1978).

[35] B. Bollobás, P. Erdös, M. Simonovits and E. Szemerédi, Extremal graphs without large forbidden subgraphs, *Annals of Discrete Mathematics* 3 (1978), 29–41.

[36] J. Bondy, J. Shen, S. Thomassé and C. Thomassen, Density conditions for triangles in multipartite graphs, *Combinatorica*, to appear.

[37] C. Borgs, J. Chayes, L. Lovász, V.T. Sós, B. Szegedy and K. Vesztergombi, Graph limits and parameter testing, Proc. of STOC 2006, 261-270.

[38] W. G. Brown, P. Erdős and V.T. Sós, Some extremal problems on r-graphs, New Directions in the Theory of Graphs, Proc. 3^{rd} Ann Arbor Conference on Graph Theorey, Academic Press, New York, 1973, 55-63.

[39] W. G. Brown, P. Erdős and V.T. Sós, On the existence of triangulated spheres in 3-graphs and related problems, Periodica Mathematica Hungaria, 3 (1973), 221-228.

[40] L. Cai, Fixed-parameter tractability of graph modification problems for hereditary properties, Information Processing Letters, 58 (1996), 171-176.

[41] T. M. Chan, Polynomial-time approximation schemes for packing and piercing fat objects, Journal of Algorithms 46 (2003), 178-189.

[42] K. Cirino, S. Muthukrishnan, N. Narayanaswamy and H. Ramesh, graph editing to bipartite interval graphs: exact and asymptotic bounds, Proc. of 17^{th} FSTTCS (1997), 37-53.

[43] D. G. Corneil, Y. Perl and L. K. Stewart, A linear recognition algorithm for cographs, SIAM J. Comput. 14 (1985), 926–934.

[44] D. G. Corneil, H. Lerchs, L. Stewart Burlingham, Complement Reducible Graphs, Discrete Applied Mathematics 3 (1981), 163–174.

[45] A. Czumaj and C. Sohler, Testing hypergraph coloring, Proc. of ICALP 2001, 493-505.

[46] A. Czumaj and C. Sohler, Property testing in computational geometry, Proceedings of the 8th Annual European Symposium on Algorithms (2000), 155–166.

[47] A. Czumaj and C. Sohler, Abstract combinatorial programs and efficient property testers, SIAM Journal on Computing 34 (2005), 580-615.

[48] Reinhard Diestel, **Graph Theory**, Second Edition, Springer-Verlag, New York, 2000.

[49] D. Eichhorn and D. Mubayi, Edge-coloring cliques with many colors on subcliques, Combinatorica 20 (2000), 441-444.

[50] E. S. El-Mallah and C. J. Colbourn, The complexity of some edge-deletion problems, IEEE transactions on circuits and systems, 35 (1988), 354-362.

[51] P. Erdős, Graph theory and probability, Canad. J. Mathematics, (11) 1959, 34-38.

[52] P. Erdős, On extremal problems of graphs and generalized graphs. Israel J. Math. 2 1964 183-190.

[53] P. Erdős, On some new inequalities concerning extremal properties of graphs, Theory of Graphs (Proc. Colloq., Tihany, 1966), Academic Press, New York, 1968, 77–81.

[54] P. Erdős, P. Frankl and V. Rödl, The asymptotic number of graphs not containing a fixed subgraph and a problem for hypergraphs having no exponent, Graphs Combin. 2 (1986) 113-121.

[55] P. Erdős and A. Gyárfás, A variant of the classical Ramsey problem, Combinatorica 17 (1997), 459-467.

[56] P. Erdős and M. Simonovits, On a valence problem in extremal graph theory, Discrete Math. 5 (1973), 323-334.

[57] T. Feder, P. Hell, S. Klein, and R. Motwani, Complexity of list partitions, Proc. of STOC 1999, 464-472. Also, SIAM J. Comput., in press.

[58] W. Fernandez de la Vega, Max-Cut has a randomized approximation scheme in dense graphs, Random Structures and Algorithms, 8(3) 1996, 187-198.

[59] E. Fischer, The art of uninformed decisions: A primer to property testing, The Computational Complexity Column of The Bulletin of the European Association for Theoretical Computer Science 75 (2001), 97-126.

[60] E. Fischer, Testing graphs for colorability properties, Proc. of the 12^{th} SODA (2001), 873-882.

[61] E. Fischer, The difficulty of testing for isomorphism against a graph that is given in advance, SIAM Journal on Computing, 34, 1147-1158.

[62] E. Fischer, G. Kindler, D. Ron, S. Safra, and A. Samorodnitsky, Testing juntas, Proc. of The 43^{rd} FOCS (2002), 103-112.

[63] E. Fischer, E. Lehman, I. Newman, S. Raskhodnikova, R. Rubinfeld and A. Samorodnitsky, Monotonicity testing over general poset domains, Proc. of The 34^{th} STOC (2002), 474-483.

[64] E. Fischer and I. Newman, Testing versus estimation of graph properties, Proc. of the 37^{th} Annual Symp. on Theory of Computing (STOC), Baltimore, Maryland, 2005, 138-146.

[65] E. Fischer, I. Newman and J. Sgall, Functions that have read-twice constant width branching programs are not necessarily testable, Random Structures and Algorithms, in press.

[66] P. Frankl and Z. Füredi, Colored packings of sets in combinatorial design theory, Annals of Discrete Math. 34 (1987), 165-178.

[67] E. Friedgut and G. Kalai, Every monotone graph property has a sharp threshold. Proc. Amer. Math. Soc. 124 (1996), 2993-3002.

[68] K. Friedl, G. Ivanyos and M. Santha, Efficient testing of groups, Proc. of STOC (2005), 157-166.

[69] A. Frieze and R. Kannan, The regularity lemma and approximation schemes for dense problems, Proc. of 37^{th} FOCS, 1996, 12-20.

[70] A. Frieze and R. Kannan, Quick approximation to matrices and applications, Combinatorica, 19(2), 1999, 175-220.

[71] Z. Füredi, Turán type problems, in: *Surveys in combinatorics*, London Math. Soc. Lecture Note Ser. 166, Cambridge Univ. Press, Cambridge, 1991, 253–300

[72] M.R. Garey and D.S. Johnson, Computers and Intratability: A guide to the Theory of NP-Completeness, W.H. Freeman and Co., San Francisco, 1979.

[73] P. W. Goldberg, M. C. Golumbic, H. Kaplan and R. Shamir, Four strikes against physical mapping of DNA, Journal of Computational Biology 2 (1995), 139–152.

[74] O. Goldreich, Combinatorial property testing - a survey, In: Randomization Methods in Algorithm Design (P. Pardalos, S. Rajasekaran and J. Rolim eds.), AMS-DIMACS (1998), 45-60.

[75] O. Goldreich, S. Goldwasser and D. Ron, Property testing and its connection to learning and approximation, Proc. of 37^{th} Annual IEEE FOCS, (1996), 339–348. Also: JACM 45(4): 653-750 (1998).

[76] O. Goldreich and D. Ron, Property Testing in Bounded-Degree Graphs, Proc. of STOC 1997, 406-415.

[77] O. Goldreich and L. Trevisan, Three theorems regarding testing graph properties, Proc. 42^{nd} IEEE FOCS, IEEE (2001), 460-469. Also, Random Structures and Algorithms, 23(1):23-57, 2003.

[78] M.C. Golumbic, **Algorithmic Graph Theory and Perfect Graphs**, Academic Press, 1980.

[79] M. C. Golumbic, H. Kaplan and R. Shamir, On the complexity of DNA physical mapping, Advances in Applied Mathematics, 15 (1994), 251-261.

[80] R. L. Graham, B. L. Rothschild and J. H. Spencer, *Ramsey Theory*, Second Edition, Wiley, New York, 1990.

[81] W. T. Gowers, Hypergraph regularity and the multidimensional Szemerédi theorem, manuscript.

[82] P. E. Haxell and V. Rödl, Integer and fractional packings in dense graphs, Combinatorica 21 (2001), 13-38.

[83] P. Hell and J. Nešetřil, **Graphs and Homomorphisms**, Oxford University Press, 2004.

[84] P. Hell and J. Nešetřil, The core of a graph, Discrete Math 109 (1992), 117-126.

[85] P. Hell, J. Nešetřil, and X. Zhu, Duality of graph homomorphisms, in : Combinatorics, Paul Erdős is Eighty, (D. Miklós et. al, eds.), Bolyai Society Mathematical Studies, Vol.2, 1996, pp. 271-282.

[86] D. Karger, R. Motwani and M. Sudan, Approximate graph coloring by semidefinite programming, JACM 45(2), 1998, 246-265.

[87] S. Khot and V. Raman, Parameterized complexity of finding subgraphs with hereditary properties, COCOON 2000, 137-147.

[88] Y. Kohayakawa, B. Nagle and V. Rödl, Efficient testing of hypergraphs, Proc. of 29^{th} ICALP, (2002), 1017–1028.

[89] Y. Kohayakawa, V. Rödl and L. Thoma, An optimal algorithm for checking regularity, SIAM J. on Computing 32 (2003), no. 5, 1210-1235.

[90] J. Komlós and M. Simonovits, Szemerédi's Regularity Lemma and its applications in graph theory. In: *Combinatorics, Paul Erdös is Eighty*, Vol II (D. Miklós, V. T. Sós, T. Szönyi eds.), János Bolyai Math. Soc., Budapest (1996), 295–352.

[91] T. Kövari, V.T. Sós and P. Turán, On a problem of K. Zarankiewicz,*Colloquium Math.* 3 (1954), 50-57.

[92] M. Krivelevich and B. Sudakov, Pseudo-random graphs, More Sets, Graphs and Numbers, E. Gyori, G. O. H. Katona and L. Lovasz, Eds., Bolyai Society Mathematical Studies Vol. 15 (2006), 199-262.

[93] J. Lewis and M. Yannakakis, The node deletion problem for hereditary properties is NP-complete, JCSS 20 (1980), 219-230.

[94] L. Lovász, On the shannon capacity of a graph, IEEE Transactions on Information Theory 25(1), 1979, 1-7.

[95] L. Lovász and B. Szegedy, Graph limits and testing hereditary graph properties, manuscript, 2005.

[96] A. Lubotzky, R. Phillips and P. Sarnak, Ramanujan graphs, Combinatorica, 8 (1988), 261-277.

[97] T. A. McKee and F.R. McMorris, **Topics in Intersection Graph Theory**, SIAM, Philadelphia, PA, 1999.

[98] B. Nagle, V. Rödl and M. Schacht, The counting lemma for regular k-uniform hypegraphs, manuscript.

[99] A. Natanzon, R. Shamir and R. Sharan, Complexity classification of some edge modification problems, Discrete Applied Mathematics 113 (2001), 109–128.

[100] I. Newman, Testing of functions that have small width branching programs, Proc. of 41^{th} FOCS (2000), 251-258.

[101] M. Parnas and D. Ron, Testing the diameter of graphs, Random structures and algorithms, 20 (2002), 165-183.

[102] C. Papadimitriou, **Computational Complexity**, Addison Wesley, 1994.

[103] M. Parnas, D. Ron and R. Rubinfeld, Tolerant property testing and distance approximation, manuscrip, 2004.

[104] J. L. Ramírez-Alfonsín, B. A. Reed (Editors), **Perfect Graphs**, Wiley, 2001.

[105] V. Rödl and R. Duke, On graphs with small subgraphs of large chromatic number, *Graphs and Combinatorics* 1 (1985), 91–96.

[106] V. Rödl and J. Skokan, Regularity lemma for k-uniform hypergraphs, Random Structures and Algorithms, 25 (2004), 1-42.

[107] D. Ron, Property testing, in: P. M. Pardalos, S. Rajasekaran, J. Reif and J. D. P. Rolim, editors, *Handbook of Randomized Computing*, Vol. II, Kluwer Academic Publishers, 2001, 597–649.

[108] J. D. Rose, A graph-theoretic study of the numerical solution of sparse positive-definite systems of linear equations, Graph Theory and Computing, R.C. Reed, ed., Academic Press, N.Y., 1972, 183-217.

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

[110] I. Ruzsa and E. Szemerédi, Triple systems with no six points carrying three triangles, in Combinatorics (Kenszthely, 1976), Coll. Math. Soc. J. Bolyai 18, Volume II, 939-945.

[111] M. Simonovits, A method for solving extremal problems in graph theory, stability problems, Theory of Graphs (Proc. Colloq., Tihany, 1966), Academic Press, New York, 1968, 279–319.

[112] E. Szemerédi, Regular partitions of graphs, In: *Proc. Colloque Inter. CNRS* (J. C. Bermond, J. C. Fournier, M. Las Vergnas and D. Sotteau, eds.), 1978, 399–401.

[113] D. B. West, **Introduction to Graph Theory**, Prentice Hall, 2001.

[114] J. Xue, Edge-maximal triangulated subgraphs and heuristics for the maximum clique problem. Networks 24 (1994), 109-120

[115] M. Yannakakis, Edge-deletion problems, SIAM J. Comput. 10 (1981), 297-309.

[116] R. Yuster, Integer and fractional packing of families of graphs, Random Structures and Algorithms 26 (2005), 110-118.