Logical Methods for Property Testing

in the Bounded Degree Model

Noleen Köhler

Submitted in accordance with the requirements
for the degree of Doctor of Philosophy

The University of Leeds
School of Computing
April 2021

The candidate confirms that the work submitted is his/her own, except where work which has formed part of a jointly authored publication has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Some parts of the work presented in Chapters 6, 7, 8 and 9 have been published in the following articles:

– **I. Adler, N. Köhler and P. Peng,** On Testability of First-Order Properties in Bounded-Degree Graphs, In: Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA). Society for Industrial and Applied Mathematics, 2021. S. 1578-1597

– **I. Adler and N. Köhler,** An explicit construction of graphs of bounded degree that are far from being Hamiltonian, under review

– **I. Adler, N. Köhler and P. Peng,** GSF-Locality Is Not Sufficient For Proximity-Oblivious Testing, In: Proceeding of the 36th Computational Complexity Conference (CCC 2021). volume 200 of Leibniz International Proceedings in Informatics (LIPIcs), pages 34:1– 34:27, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik

All three papers are primarily the work of the second author (N. Köhler). Most technical parts of all three papers were written by the second author. The first author (I. Adler) and the third author (P. Peng) contributed mostly but not exclusively towards the presentation and motivation of the results. Some proof ideas and research directions were results of discussions between all authors on the respective paper. The first author also had the role of primary supervisor of the candidate.

Acknowledgements

First and foremost, I would like to thank my supervisor Isolde Adler for her positivity, her continued support and the many interesting discussion we had. Isolde's enthusiasm has inspired me to tackle hard problem with her. I would also like to thank my collaborator Pan Peng for many fruitful discussions and for sharing his expertise in property testing. I am confident that Pan's influence on my research has been positive.

I would like to thank my examiners Anuj Dawar and Haiko Müller for their great effort in reading my thesis, for the many helpful comments and for the nice discussion we had during my viva.

I would like to express my gratitude to Jake, Judith, Polly and Sam for proof reading my thesis.

The School of Computing, at Leeds, has provided me with an enjoyable and productive research environment. I would like to thank the School of Computing for funding my PhD and travel whilst studying. During my time at Leeds I have encountered some amazing people who I have spent many enjoyable times with. I would especially like to thanks Bernard, my coffee and brownie partner, who I spent many hours talking to. To Luke, Judith and Jake for being my office mates with lots of interesting, and often random, distractions from work.

I would especially like to mention Polly, my fellow property testing PhD student in the School of Computing, for many interesting discussions about research. Special thanks also goes to Sam, who has been my moral support throughout my PhD, who has cheered me up whenever I needed cheering up and listened to me whenever I became a flibbertigibbet.

I would like to thank "the pub group", including Luke, Sam, Jake, Judith, Leroy, Bjarki, Petar, Mark and Guilia for many hours spent in various public houses - discovering British beers.

Last but not least, I would like to thank my family, especially by parents and my brothers. My parents continued care and support has been appreciated and has enabled me to get to this stage in my studies. My brothers endless concern for me has always inspired me to be a better person.

Abstract

Property testers are randomised sublinear time algorithms which infer global structure from a local view of an input. In the context of bounded degree graphs, testability of a property is tightly linked to the question of whether an approximate distribution of r-neighbourhood types of a graph is sufficient to capture whether the graph has the property or is "far" from having the property. Our understanding of when this is the case is limited. The central open question in the field of property testing is to characterise testable properties of bounded degree graphs.

Towards a characterisation of testable properties in the bounded degree model, we study property testing of properties definable in first-order logic (FO) in the bounded degree model. By Gaifman's locality theorem it is known that FO can only express local properties. On the other hand, testers can explore only local neighbourhoods and hence locality is necessary for testability. We prove however, that the notion of locality imposed by being definable in FO is not sufficient for property testing. More precisely, we show that there is a non-testable property defined by an FO-sentence whose quantifier prefix contains only one quantifier alternation of the form "$\forall\exists$". We complement this by proving that every FO-sentence not containing a quantifier alternation of the form "$\forall\exists$" can be tested with a constant number of queries. We further identify some classes of FO-sentences which yield testable properties and contain quantifier alternations of the form "$\forall\exists$". These sentences express that the distribution of r-neighbourhood types is of a particular, simple form.

We explore the connection between the notion of locality imposed by FO and the notion of locality necessary for testability further. We establish links between FO definability and a generalised notion of subgraph freeness. This notion was introduced in the context of characterising one-sided error proximity oblivious testers, a particularly simple type of testers. Using a variation of our non-testable FO definable property we show that generalised subgraph freeness does equally not capture the locality needed for testability, which answers an open question regarding the characterisation of testable properties in the bounded degree model.

Going beyond FO definability, we explore hardness of property testing problems in connection with classical complexity theory. We obtain a deterministic construction of hard instances for property testing Hamiltonicity, which is a known non-testable property. This construction is interesting from the perspective of exploring links between structural properties and non-testability. We further utilise the lower bound technique developed in the context of our non-testable FO definable property to prove non-testability of having treewidth logarithmic in the number of vertices.

Contents

List of Figures

List of Notation

$\mathbb{N}$	the set of natural numbers including 0
$[n]$	the set $\{0, \ldots, n-1\}$
$\mathcal{S}_n$	symmetric group on $[n]$
$\sqcup$	disjoint union
$\triangle$	symmetric difference
$\mathbb{P}[A]$	probability of A
$\mathbb{P}[A \mid B]$	probability of A under condition B
$\mathbb{E}[X]$	expected value of random variable X
NP	non-deterministic polynomial time
PSPACE	polynomial space
$\mathcal{O}(\cdot)$	asymtotic upper-bound
$o(\cdot)$	aymtotic tight upper-bound
$\Omega(\cdot)$	asymtotic lower-bound
$\omega(\cdot)$	asymtotic strict lower-bound
$\Theta(\cdot)$	asymtotic strict upper- and lower-bound
ⓩ	zig-zag product
K_n	complete graph on n vertices
$h(G)$	expansion ratio of G
$\langle S, T \rangle_G$	edges crossing S and T in G
σ	signature
$\mathcal{C}_d$	class of graphs of bounded degree d
C_d	class of σ-structures of bounded degree d
P_φ	property defined by formula φ
FO	first-order logic
CNF	conjunctive normal form
DNF	disjunctive normal form
GNF	Gaifman normal form
HNF	Hanf normal form

Σ_i, Π_i, Δ_i	prefix classes with $i-1$ quantifier alterations
$\models$	is a model of
$\equiv$	equivalence of FO-formulas
$\equiv_d$	equivalence of FO-formulas on structures of bounded degree d
$\neg$, $\wedge$, $\vee$, $\to$, $\leftrightarrow$	logical negation, conjunction, disjunction, implication and biimplication
$\exists$, $\forall$	existential and universal quantifier
POT	proximity oblivious tester
GSF	generalised subgraph freeness
$\mathrm{ans}_{\mathcal{A}}(q)$	answer to query q to structure $\mathcal{A}$

Chapter 1

Introduction

Computational challenges arising in the context of big data are omnipresent in modern day computer science. As technology advances, data sets get increasingly large. Due to the nature of distributed systems we are faced with the challenge of big data sets for which classical algorithmic approaches are impractical. In order to still do a qualitative analysis of the data, we have to come up with new algorithmic approaches. In cases where the data set is so large that storing the entire data set in working memory is infeasible, we can consider streaming algorithms. These algorithms see a random stream of the data but can only store a certain amount of it at any point in time (see e. g. [105, 108]). Now assume that the data set is so large it would even be unfeasible to process it even once. In this case we can consider using a sampling based algorithm which explores only a small (randomly picked) part of the object. Property testing is a framework for studying sampling based algorithms that solve a relaxation of a decision problem. Given a property P, a property testing algorithm (short: tester) for P is given query access to an input object $\mathcal{A}$ (i. e. some large data set) and has to decide whether $\mathcal{A}$ has property P or is far from having property P, where the notion of farness is model dependent. Property testing was first proposed in 1996 by Rubinfeld and Sudan in the context of program checking [117]. The idea of property testing was extended to graphs shortly after. In 1996 Goldreich, Goldwasser and Ron introduced the dense model [71]. Property testing on dense graphs is well understood due to a characterisation of which properties are testable on dense graphs by Alon, Fischer, Newman and Shapira [7] based on Szemerédi's celebrated regularity Lemma [121].

Shortly after the introduction of the dense model, Goldreich and Ron introduced the bounded degree model [73]. In the bounded degree model we consider graphs of degree at most d, where $d \in \mathbb{N}$ is a constant. A tester accesses an input graph via neighbour queries, i. e. for any vertex v a tester can obtain the i-th neighbour of v where $i \in \{1, \dots, d\}$. A graph G on n vertices is ϵ-far from a property P if more than ϵdn edge modifications are necessary to make G have the property P. A tester has to distinguish whether an input graphs G has property P or is ϵ-far from having property P with probability $\frac{2}{3}$ correctly, where the number of queries the tester makes might depend on ϵ and d, but is independent of the size of G. Since the introduction of the bounded degree model, extensive research has been done into studying property testing in this model (see Chapter 4 for an overview). However, in the bounded degree model no characterisation of which properties are testable is known. This is a long standing open problem in the area of property testing. In this thesis we approach the question from the direction of first-order logic.

Studying property testing of first-order logic (FO) definable properties of graphs, we are aiming for algorithmic meta-theorems. Recall that FO for graphs is recursively defined from the atomic formulas expressing equality and adjacency of vertices using boolean connectives (negation, conjunction, disjunction, implication and biimplication) and existential and universal quantification over vertices. FO can express a variety of properties such as subgraph freeness and subgraph containment, which are constant query testable in the bounded degree model [73]. There are also properties which are constant query testable but can not be expressed in FO, e. g. connectivity and cycle freeness [73]. Furthermore, the corresponding decision problem i. e. deciding a property defined by an FO-sentence on the class of bounded degree graphs, takes linear time by Seese's theorem [119].

On bounded degree graphs Hanf's theorem [83] implies a normal form for FO, so called Hanf normal form (HNF). A sentence in HNF is a boolean combination of Hanf sentences, which are sentences of the form $\exists^{\geq m} x \phi_\tau(x)$ expressing that there are at least m vertices whose r-neighbourhoods have isomorphism type τ. Every sentence is equivalent to a sentence in HNF on any class of bounded degree graphs. This implies that satisfying a given formula depends on which neighbourhood types appear in a graph. On the other hand, property testers for bounded degree graphs essentially sample a constant set of r-neighbourhoods and compute an answer only depending on which neighbourhood types they observe (see [78] and [34]). It is known that a constant query tester can estimate the relative frequencies of neighbourhood isomorphism types appearing in a bounded degree graph well [112]. This hinges on the following modification problem. Assume φ is a sentence in HNF. For any graph G for which the relative frequencies of neighbourhood types appearing in G almost satisfy the requirements for satisfying φ, is there a set of at most ϵdn edge modification to adjust the frequencies of neighbourhoods appearing in G according to φ. Taking into account this close connection between testability and FO definability, the question of whether all FO definable properties are testable seems reasonable and was in fact asked in [2].

We study testability of FO definable properties by prefix classes motivated by a similar study in the dense model by Alon, Fischer, Krivelevich, and Szegedy [6]. Here an FO-sentence is in the prefix class Σ_2 if it is equivalent to a sentence in prenex normal form with quantifier prefix of the form $\exists x_1 \ldots \exists x_k \forall y_1 \ldots \forall y_\ell$ where $k, \ell \in \mathbb{N}$. Similarly, a sentence is in Π_2 if it is equivalent to a sentence in prenex normal form with quantifier prefix of the form $\forall x_1 \ldots \forall x_k \exists y_1 \ldots \exists y_\ell$ where $k, \ell \in \mathbb{N}$. We obtain the following result for testing FO definable properties in the bounded degree model (proved in Chapter 6).

Theorem. *Every FO-sentence $\varphi \in \Sigma_2$ defines a testable property in the bounded degree model. On the other hand, there is a property in Π_2 which is not testable in the bounded degree model.*

For the testability of any sentences φ in Σ_2 we show that φ has a certain structure which allows us to reduce testability of satisfying φ to the case of testing subgraph freeness using certain closure properties of property testing. On the other hand, for the non-testability result we define an FO-sentence $\varphi_{Ⓩ}$ of relational structures, which we use for modelling purposes. The sentence $\varphi_{Ⓩ}$ essentially defines a property of structures whose underlying graphs are edge expanders and is equivalent to a sentence in Π_2 on structures of bounded degree d. These structures are constructed exploiting a recursively defined expander construction based on the zig-zag product introduced by Reingold, Vadhan and Wigderson [115]. Replacing relations by suitable graph gadgets we obtain an FO definable class of bounded degree expanders. Beyond the negative algorithmic ramifications, this shows how surprisingly expressive FO is despite its locality (see Gaifman's locality theorem [63] and Hanf normal form [83]). Besides our construction we are not aware of any other (infinite) class of expanders which is definable in FO.

We explore testability of FO definable properties further in Chapter 7. We call the property of all graphs where neighbourhood isomorphism type τ does not appear, τ-neighbourhood freeness. We show that this generalisation of subgraph freeness, which can be expressed by a negated Hanf sentence, is testable under some mild assumptions on the degree. We further consider the property where the neighbourhood of every vertex has isomorphism type τ, which we call τ-neighbourhood regularity. We identify a special class of radius 1 neighbourhood isomorphism types τ, for which τ-neighbourhood freeness is testable. Furthermore, τ-neighbourhood freeness and τ-neighbourhood regularity are in general not expressible by a sentence in Σ_2 and hence testability does not follow from our previous testability result. This further implies that prefix classes do not yield a characterisation of which FO definable properties are testable.

Modification problem similar to the one stated above form the core of the question of a characterisation of which properties are testable in the bounded degree model. In their seminal work on a special class of particularly simple property testers, i. e. proximity oblivious testers (POTs) with one-sided error, Goldreich and Ron gave a characterisation of POTs with one-sided error using the notion of generalised subgraph freeness [76]. Generalised subgraph freeness intuitively expresses that some induced subgraphs can not appear with a specific interface to the rest of the graph. In [76] the following question is asked, "Is every generalised subgraph freeness property non-propagating?", where intuitively a generalised subgraph freeness property is non-propagating if the removal of a small set of appearances of such generalised subgraphs can be removed without causing a chain reaction of necessary modification. A recent work of Ito, Khoury and Newman, in which one-sided error testability of both monotone properties and hereditary properties is characterised, picks up this open question [89]. Indeed, [89] provides evidence that a positive answer to the question asked in [76] would lead to a classification of one-sided error testability in the bounded degree model. ‘We answer the question asked in [76] negatively by showing that a minor variation of the non-testable FO-property defined by $\varphi_{Ⓩ}$ can be expressed as a forbidden subgraph freeness property (see Chapter 8). For this we identify a condition of FO-sentences that implies being a generalised subgraph freeness property and argue that a slight variation of the sentence $\varphi_{Ⓩ}$ satisfies this condition. Even though this is essentially a negative result, it provides us with some insights into the problem of a characterisation of which properties are testable in the bounded degree model.

We further explore connections between property testing and classical complexity theory. Reducibility amongst property testing problems requires a notion of reduction that allows simulating a tester for a problem by another tester. Since query access is local this implies that the reduction has to be local in the sense that presence of a certain edge in the reduced graph can only depend on a constant size set of neighbourhoods in the original graph. These local reductions do not have to be restricted in time, because they are only computed locally in the simulation. In some cases (particularly but not necessarily when the polynomial time reduction is linear) polynomial time reductions are essentially local reductions (see e. g. [70,129]). We give an example of such a case in Section 9.1 for dominating set. The existence of a polynomial time reduction amongst decision problems restricted to bounded degree graphs does not in general yield a local reduction, i. e. there are NP-hard problems that are testable in the bounded degree model (for details see Chapter 9). A problem where the polynomial time reduction known is not local is treewidth ([14, 66, 68] or [14, 107] for bounded degree planar graphs). We obtain non-testability for treewidth using a combination of a result from Grohe and Marx showing that the treewidth of expanders is linear in their size and a lower-bound technique we developed for proving non-testability of FO definable properties. This lower-bound technique combines a result by Alon [102, Proposition 19.10] and a theorem by Adler and Harwath [2, Theorem 19], but only provides non-testability with $\mathcal{O}(1)$ queries.

With the aim of understanding structural reasons for hardness we provide a deterministic construction of hard instances for testing whether a graph is Hamiltonian. Hamiltonicity can not be tested with a sublinear amount of queries [70, 129], which is due to a local reduction from 3-SAT. The graphs we construct are both far from being Hamilonian (we need at least an ϵ-fraction of edge modifications to make the graph Hamiltonian) while they locally look Hamiltonian (the neighbourhood of any δ-fraction of vertices appears in a Hamiltonian graph, for some fixed $\delta \in [0, 1]$). The construction uses a base expander and encodes a property into certain graph gadgets, which can be satisfied at a δ-fraction of the vertices but can not be satisfied for any larger amount of vertices. We hope that this construction will give us further insights into connections between non-testability and graph structure.

Outline of chapters In Chapter 2 we recall basic concepts of graph theory, relational structures and first-order logic and introduce the notation used throughout this thesis. In Chapter 3 we introduce property testing in general, the models relevant for this thesis and local reductions. In Chapter 4 we survey the history and recent developments in related areas such as model checking and property testing. In Chapter 5 we provide generalisations from bounded degree graphs to bounded degree relational structure of two results ([102, Proposition 19.10] and the canonical tester result [34]). In Chapter 6 we provide the proof of the characterisation by prefix classes of FO definable properties. In Chapter 7 we prove testability results for neighbourhood regularity and neighbouhood freeness under certain restrictions. We prove that there is a non-testable generalised subgraph freeness property in Chapter 8. Results related to testing NP-hard problems are contained in Chapter 9. We provide concluding remarks in Chapter 10.

Chapter 2

Preliminaries

In this section we give a short overview of the basic concepts and notation used. Note that a detailed introduction to property testing is postponed to Section 3. Furthermore, some concepts only needed in specific chapters of this thesis are given in the respective chapter. To keep this chapter as concise as possible we also omit introducing notions of complexity theory like big-O notation and refer e. g. to [28].

Besides introducing some general notation in Section 2.1, we introduce basic concepts of graph theory in Section 2.2. We further introduce relational structures in Section 2.3 including some notions for bounded degree relational structures. We give an introduction to first-order logic in Section 6.3 including an introduction to all normal forms for first-order logic required in this thesis.

2.1 Set notation

In the following $\mathbb{N} = \{0, 1, 2, 3, \dots\}$ denotes the set of all natural numbers. We denote the set $\mathbb{N} \setminus \{0\}$ of positive integers by $\mathbb{N}_{>0}$. For any natural number $n \in \mathbb{N}$ we denote the set of the n smallest natural numbers by $[n] := \{m \in \mathbb{N} : m < n\}$.

For a function $f : A \to B$ we denote by $f(S)$ for some $S \subseteq A$ the set $\{f(a) \mid a \in S\}$. We further denote the restriction of f to $S \subseteq A$ by $f|_S$.

More convention on the notation used in this thesis can be found in the list of notation given in the preamble.

2.2 Graphs

In this section we give a short introduction to the graph theory concepts we need. A more detailed introduction to graph theory can be found for example in the book [42]. We further like to point out that some further notions for graphs are defined in Section 2.3.

A *(simple, undirected) graph* G is a tuple $G = (V, E)$, where V is a finite set and $E \subseteq \{e \subseteq V \mid |e| = 2\}$. We call the elements of V *vertices* of G and the elements of E *edges* of G. For a graph $G = (V, E)$ we denote the set of vertices V by $V(G)$ and the set of edges E by $E(G)$. We say that a vertex $v \in V(G)$ is *incident* to an edge $e \in E(G)$, if $v \in e$. We call two vertices $v, w \in V(G)$ *adjacent*, if v, w are incident to the same edge. If two vertices $v, w \in V(G)$ are adjacent, we say that w is a *neighbour* of v (note that the neighbourhood of a vertex is defined later in Section 2.3). The *size* of a graph G is defined to be $|V(G)| + |E(G)|$. The *order* of a graph G is the number of vertices $|V(G)|$. A *(graph) isomorphism* from a graph G to a graph H is a bijective map $f : V(G) \to V(H)$ which preserves adjacency, i. e. for any $v, w \in V(G)$

$$\{v, w\} \in E(G) \qquad \Longleftrightarrow \qquad \{f(v), f(w)\} \in E(H).$$

We say that G and H are *isomorphic*, denoted $G \cong H$, if there is an isomorphism from G to H. A graph H is a *subgraph* of a graph G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For any set $S \subseteq V(H)$ we say that the graph (S, E_S), where $E_S := \{e \in E(G) \mid e \subseteq S\}$, is the subgraph induced by S. We say that H is an *induced subgraph* of G if H is the subgraph induced by some set $S \subseteq V(G)$.

Let G be a graph and $v \in V(G)$. The *degree* of v, denoted $\deg_G(v)$, is the number of edges v is incident to, i. e.

$$\deg_G(v) := |\{e \in E(G) \mid v \in e\}|.$$

The *degree* of G, denoted $\deg(G)$, is the maximum degree of the vertices of G, i. e.

$$\deg(G) := \max_{v \in V(G)} \{\deg_G(v)\}.$$

We say that G is *d-regular* for some $d \in \mathbb{N}$ if every vertex in G has degree d. We say that G is *degree-regular* if there is $d \in \mathbb{N}$ such that G is d-regular.

Let G be a graph, $v, w \in V(G)$ and $\ell \in \mathbb{N}$.

- A *walk* from v to w of length ℓ is a tuple $(p_0, \dots, p_\ell) \in V(G)^{\ell+1}$ such that $v = p_0$, $w = p_\ell$ and $\{p_{i-1}, p_i\} \in E(G)$ for every $i \in \{1, \dots, \ell\}$.

- A *path* from v to w of length ℓ is a walk $(p_0, \dots, p_\ell)$ from v to w such that $\{p_{i-1}, p_i\} \neq \{p_{j-1}, p_j\}$ for every $i, j \in \{1, \dots, \ell\}$ with $i \neq j$.

- A path $(s_0, \dots, s_k)$ is a *subpath* of a path $(p_0, \dots, p_\ell)$ if there is an index $0 \leq i \leq \ell - k$ such that $s_j = p_{i+j}$ for every $1 \leq j \leq k$.

- A *simple path* in G is a path in which no vertex appears twice.

- A *cycle* of length ℓ is a path $(c_0, \dots, c_\ell)$ such that $c_0 = c_\ell$ and $(c_0, \dots, c_{\ell-1})$ is a simple path.

Note that for every vertex $v \in V(G)$ the tuple (v) is a path of length 0 from v to v. We define the graph theoretic *distance* $\operatorname{dist}_G(v, w)$ between v and w in G to be

$$\operatorname{dist}_G(v, w) := \min\left(\left\{\ell \,\middle|\, \begin{array}{l}\text{there is a path of}\\ \text{length } \ell \text{ from } v \text{ to } w\end{array}\right\} \cup \{\infty\}\right).$$

We further define an equivalence relation $\sim_c$ on $V(G)$ as follows. For every two vertices $v, w \in V(G)$

$$v \sim_c w \qquad \Longleftrightarrow \qquad \operatorname{dist}_G(v, w) < \infty.$$

We call a subgraph of G induced by an equivalence class of $\sim_c$ a *connected component* of G. We say that a graph G is *connected* if G has only one connected component, that is, if $\sim_c$ has only one equivalence class.

2.2.1 Graph representation

In property testing graph representation plays a fundamental role as it influences which information about a graph is accessible in constant time.

Let G be a graph and $V(G) = [n]$. We call the symmetric matrix $M \in \mathbb{N}^{n \times n}$ defined by

$$M_{i,j} := \begin{cases} 1 & \text{if } \{i, j\} \in E(G), \\ 0 & \text{otherwise.} \end{cases}$$

an *adjacency matrix* of G. Note that the adjacency matrix depends on the choice of an order on $V(G)$.

We call a tuple $(L_1, \ldots, L_n)$ an *adjacency list* of G where $L_i \in V(G)^{\deg_G(i)}$ is a tuple such that

$$(L_i)_j := k, \text{ where } k \text{ is the } j\text{-th neighbour of } i \text{ with respect to } \leq_i,$$

where $\leq_i$ is a total order on the set of neighbours $\{k \in V(G) \mid \{i, k\} \in E(G)\}$ for every $1 \leq i \leq n$.
Note that L_G depends on the choice of an order on $V(G)$ and the orders of neighbours for every vertex.

2.2.2 Directed graphs and multigraphs

While we mostly are interested in simple undirected graphs in some chapters we use multigraphs or directed graphs. Most concepts for simple undirected graphs can be extended to multigraphs and directed graphs in a straight forward way. Hence we will (besides introducing both types of graphs) just introduce the notions that are different from simple, undirected graphs.

A *multigraph (or an undirected graph with parallel edges and self-loops)* G is a triple $G = (V(G), E(G), f_G)$, where $V(G)$ and $E(G)$ are finite sets and

$$f_G : E(G) \to \{x \subseteq V(G) \mid 1 \leq |x| \leq 2\}.$$

Here we call f_G the *adjacency map* of G. Let G and H be two multigraphs. We call a pair of bijective maps (h_V, h_E), where $h_V : V(G) \to V(H)$ and $h_E : E(G) \to E(H)$, an *isomorphism* from G to H, if

$$h_V(f_G(e)) = f_H(h_E(e))$$

for any $e \in E(G)$. We say that G and H are *isomorphic*, denoted $G \cong H$, if there is an isomorphism from G to H. Let us remark here that every simple graph is a multigraph.

A *directed graph* G is a tuple $G = (V(G), E(G))$, where $V(G)$ is a finite set and $E(G) \subseteq V(G)^2$. While directed graphs are relational structures, which will be introduced in Section 2.3 in detail, the following concepts are particular for directed graphs. Let G be a directed graph and $v \in V(G)$. We define the set of incoming edges of v to be the set $E_G^-(v) := \{e \in E(G) \mid e = (w, v)\}$ and the set of outgoing edges of v to be $E_G^+(v) := \{e \in E(G) \mid e = (v, w)\}$.

2.2.3 Expansion and hyperfiniteness

In this section we introduce the concept of expanders and hyperfinite classes of graphs. These two concepts play a central role in property testing. We define expansion for the general case of multigraphs. We need hyperfiniteness only for simple graphs.

Definition 2.2.1 (Class of expanders)**.** Let $G = (V, E, f)$ be a multigraph.

- For any subsets $S, T \subseteq V$, $S \cap T = \emptyset$ let $\langle S, T\rangle_G := \{e \in E \mid f(e) \cap S \neq \emptyset, f(e) \cap T \neq \emptyset\}$ be the set of edges *crossing* S and T.
- For any set $S \subseteq V$, we let $h(S) := \frac{|\langle S, \overline{S}\rangle_G|}{|S|}$ be the *expansion* of S.
- We let $h(G)$ be the *expansion ratio* of G defined by

$$h(G) := \min_{\{S \subseteq V \mid |S| \leq |V|/2\}} h(S).$$

For any constant $\epsilon > 0$ we call a sequence $(G_N)_{N\in\mathbb{N}}$ of graphs a *family of ϵ-expanders*, if $|V(G_{N+1})| > |V(G_N)|$ and $h(G_N) \geq \epsilon$ for all $N \in \mathbb{N}$.

We call a class $\mathcal{C}$ a *class of expanders* if $\mathcal{C}$ contains some sequence of expanders.

Definition 2.2.2 (Hyperfinite class)**.** Let $\delta \in [0, 1]$ and $k \in \mathbb{N}_{>0}$. A graph G is called (δ, k)-hyper-finite if we can remove $\delta \cdot |V(G)|$ edges and obtain a graph whose connected components have size at most k.

A class $\mathcal{C}$ of graphs is hyper-finite if for every $\delta \in [0, 1]$ there is a $k \in \mathbb{N}_{>0}$ such that every graph in $\mathcal{C}$ is (δ, k)-hyper-finite.

Examples of hyperfinite classes of graphs are the class of all planar graphs or the class of graphs of bounded treewidth.

2.3 Relational structures

In this section we define all concepts we use for relational structures. Here we dedicate a subsection to bounded degree relational structures.

A *(relational) signature* is a finite set $\sigma = \{R_1, \dots, R_\ell\}$ of relation symbols R_i. Every relation symbol R_i has an arity $\mathrm{ar}(R_i) \in \mathbb{N}_{>0}$. For a relational signature σ we denote the maximum arity of its relation symbols by $\mathrm{ar}(\sigma)$. A *(relational) σ-structure* is a tuple $\mathcal{A} = (A, R_1^{\mathcal{A}}, \dots, R_\ell^{\mathcal{A}})$, where A is a finite set called the universe of $\mathcal{A}$ (typically we let $A = [n]$) and $R_i^{\mathcal{A}} \subseteq A^{\mathrm{ar}(R_i)}$ is an $\mathrm{ar}(R_i)$-ary relation on A for every $i \in \{1, \dots, \ell\}$. For σ-structures $\mathcal{A}, \mathcal{B}, \mathcal{C}, \dots$ we denote their universes by $A, B, C \dots$ and their respective relations by $R^{\mathcal{A}}, R^{\mathcal{B}}, R^{\mathcal{C}} \dots$ for every $R \in \sigma$. For a relational signature σ and a σ-structure $\mathcal{A}$ we call $|A|$ the size of $\mathcal{A}$. Note that we define structures to be finite. Let $\sigma = \{R_1, \dots, R_\ell\}$ be a signature, $\mathcal{A} = (A, R_1^{\mathcal{A}}, \dots, R_\ell^{\mathcal{A}})$ and $\mathcal{B} = (B, R_1^{\mathcal{B}}, \dots, R_\ell^{\mathcal{B}})$ σ-structures. An *isomorphism* from $\mathcal{A}$ to $\mathcal{B}$ is a bijective map $f : A \to B$ which preserves relations, i. e.

$$\Big(a_1, \dots, a_{\mathrm{ar}(R_i)}\Big) \in R_i^{\mathcal{A}} \iff \Big(f(a_1), \dots, f(a_{\mathrm{ar}(R_i)})\Big) \in R_i^{\mathcal{B}}$$

for every $1 \le i \le \ell$ and elements $a_1, \dots, a_{\mathrm{ar}(R_i)}$. We call $\mathcal{A}$ and $\mathcal{B}$ *isomorphic*, denoted $\mathcal{A} \cong \mathcal{B}$, if there is an isomorphism from $\mathcal{A}$ to $\mathcal{B}$. The *union* of $\mathcal{A}$ and $\mathcal{B}$, denoted $\mathcal{A} \cup \mathcal{B}$, is the σ-structure

$$\mathcal{A} \cup \mathcal{B} := \Big(A \cup B, R_1^{\mathcal{A}} \cup R_1^{\mathcal{B}}, \dots, R_\ell^{\mathcal{A}} \cup R_\ell^{\mathcal{B}}\Big).$$

If $A \cap B = \emptyset$ we often denote the union of $\mathcal{A}$ and $\mathcal{B}$ by $\mathcal{A} \sqcup \mathcal{B}$. $\mathcal{B}$ is a *substructure* of $\mathcal{A}$ if $B \subseteq A$ and $R_i^{\mathcal{B}} \subseteq R_i^{\mathcal{A}}$ for every $i \in \{1, \dots, \ell\}$. For a subset $M \subseteq A$, we let

$$\mathcal{A}[M] := \big(M, R_1^{\mathcal{A}} \cap M^{\mathrm{ar}(R_1)}, \dots, R_\ell^{\mathcal{A}} \cap M^{\mathrm{ar}(R_\ell)}\big)$$

be the substructure of $\mathcal{A}$ that is *induced* by M. We call a substructure $\mathcal{B}$ an *induced substructure* of $\mathcal{A}$, if $\mathcal{B}$ is induced by some subset $M \subseteq A$.

There is a natural way of assigning a graph to a relational structure. Let σ be a relational signature and $\mathcal{A}$ be a σ-structure. The *Gaifman graph* of $\mathcal{A}$ is the graph $G(\mathcal{A}) = (A, E)$, where $\{x, y\} \in E$, if there is a tuple $(b_1, \dots, b_{\mathrm{ar}(R_i)}) \in R_i^{\mathcal{A}}$ for some $1 \le i \le \ell$, such that $x = b_j$ and $y = b_k$ for some $1 \le k, j \le \mathrm{ar}(R_i)$ and $j \neq k$ (see for instance [101]). While $G(\mathcal{A})$ does not capture all the structural information contained in $\mathcal{A}$, we use $G(\mathcal{A})$ to apply graph theoretic notions to relational structures. For two elements $a, b \in A$ we define the *distance* between a

and b in $\mathcal{A}$, denoted by $\mathrm{dist}_{\mathcal{A}}(a,b)$, to be the graph theoretic distance of a and b in $G(\mathcal{A})$, i. e.

$$\mathrm{dist}_{\mathcal{A}}(a,b) := \mathrm{dist}_{G(\mathcal{A})}(a,b).$$

This allows us to define neighbourhoods and neighbourhood types. Let $r \in \mathbb{N}$ and $a \in A$. The *r-neighbourhood* of a, denoted $N_r^{\mathcal{A}}(a)$, is the set of all elements in A with distance at most r to a, i. e.

$$N_r^{\mathcal{A}}(a) := \{b \in A \mid \mathrm{dist}_{\mathcal{A}}(a,b) \leq r\}.$$

For a subset of elements $S = \{a_1, \dots, a_k\} \subseteq A$ we define the r-neighbourhood of S to be $N_r^{\mathcal{A}}(S) := \bigcup_{i=1}^{k} N_r^{\mathcal{A}}(a_i)$. We denote the structure induced by the r-neighbourhood of a by

$$\mathcal{N}_r^{\mathcal{A}}(a) := \mathcal{A}[N_r^{\mathcal{A}}(a)]$$

and equivalently we let $\mathcal{N}_r^{\mathcal{A}}(S) := \mathcal{A}[N_r^{\mathcal{A}}(S)]$ for any $S \subseteq A$. An *r-ball* is a tuple $(\mathcal{B}, b)$ where $\mathcal{B}$ is a σ-structure and $b \in B$ such that $N_r^{\mathcal{B}}(b) = B$. For an r-ball $(\mathcal{B}, b)$ we call b the *centre* of $(\mathcal{B}, b)$. Two r-balls $(\mathcal{B}, b), (\mathcal{B}', b')$ are *isomorphic* if there is an isomorphism f from $\mathcal{B}$ to $\mathcal{B}'$ such that $f(b) = b'$. We call the isomorphism classes of r-balls *r-types*. Note that by definition we have that $(\mathcal{N}_r^{\mathcal{A}}(a), a)$ is an r-ball for any σ-structure $\mathcal{A}$ and any $a \in A$. For an r-type τ we say that a has (neighbourhood) type τ if $(\mathcal{N}_r^{\mathcal{A}}(a), a) \in \tau$.

Let σ be a signature. A *class of σ-structures* is a set C of σ-structures, which is closed under isomorphism, i. e. if $\mathcal{A} \in C$ and $\mathcal{B}$ is isomorphic to $\mathcal{A}$ then $\mathcal{B} \in C$. For a class C of relational structures we denote by $C|n := \{\mathcal{A} \in C \mid |A| = n\}$ the subset of all structures on n elements in C. A *property* on C is a class $P \subseteq C$ of structures. We say that a structure $\mathcal{A} \in C$ has property P if $\mathcal{A} \in P$.

Remark 2.3.1. Let $\sigma_{\mathrm{Graph}} := \{E\}$, where E is a relation symbol of arity 2. We can define a directed graph $\mathcal{G} = (A, E^{\mathcal{G}})$ (or di-graphs) to be a σ_{Graph}-structure, where the universe A is the set of vertices and the tuples in $E^{\mathcal{G}}$ define the edges.

We can regard undirected graphs as defined in Section 2.2 as a subclass of directed graphs, where a directed graph is an undirected graph if the edge relation is symmetric and irreflexive. Therefore we get that the notion of neighbourhoods, r-balls, r-types, properties and classes directly translate to graphs.

2.3.1 Bounded-degree structures and neighbourhood distributions

In this section we introduce classes of bounded degree relational structures/graphs which are of particular interest to us. We further point out some particularities for bounded degree which are the basis for our work.

Let σ be a relational signature, $\mathcal{A}$ be a σ-structure and C be a class of σ-structures. The *degree* of an element $a \in A$ denoted by $\deg_{\mathcal{A}}(a)$ is defined to be the number of tuples of $\mathcal{A}$ in which a occurs, i. e.

$$\deg_{\mathcal{A}}(a) := \sum_{R \in \sigma} \Big| \Big\{ (a_1, \ldots, a_{\mathrm{ar}(R)}) \in R^{\mathcal{A}} \ \Big| \ a \in \{a_1, \ldots, a_{\mathrm{ar}(R)}\} \Big\} \Big|.$$

We define the *degree* of $\mathcal{A}$ denoted by $\deg(\mathcal{A})$ to be the maximum degree of its elements. For an r-type τ we let the degree of τ be $\deg(\mathcal{B})$ for $(\mathcal{B}, b) \in \tau$. A structure $\mathcal{A}$ is called *degree* regular if for every element $a \in A$ we have $\deg_{\mathcal{A}}(a) = \deg(\mathcal{A})$, i. e. if every element in $\mathcal{A}$ has the same degree. Let $d \in \mathbb{N}$. We say that C has bounded degree d if $\deg(\mathcal{A}) \leq d$ for all $\mathcal{A} \in C$. For $d \in \mathbb{N}$ we denote the class of all σ-structures of bounded degree d by C_d. Note that for simplicity we omit σ from the notation. For $d \in \mathbb{N}$ we denote the class of graphs of bounded degree d by $\mathcal{C}_d$.

Remark 2.3.2. Note that the degree for undirected graphs defined in Section 2.2 is by a factor two smaller than the degree defined for the corresponding relational structures due to the symmetry of the edge relation.

In the following we will argue that for bounded degree structures/graphs there is a finite number of r-types which allows us to define vectors capturing precisely how often a certain neighbourhood type appears in a structure/graph. These vectors play a central role both for first-order logic and property testing and will reoccur throughout the following chapters.

Lemma 2.3.3. *Let σ be a relational signature, $d \in \mathbb{N}$ and $\mathcal{A}$ be a σ-structure of bounded degree d. For all $r \in \mathbb{N}$ and $a \in A$*

$$|N_r^{\mathcal{A}}(a)| \leq (2 \cdot d \cdot \mathrm{ar}(\sigma))^r.$$

Proof. We can write $N_r^{\mathcal{A}}(a)$ as a disjoint union

$$N_r^{\mathcal{A}}(a) = \bigsqcup_{i=0}^{r} \{b \in A \mid \mathrm{dist}_{\mathcal{A}}(a, b) = i\}.$$

Since every element $b \in A$ can be in no more then d tuples, each of which contains fewer than $\mathrm{ar}(\sigma)$ elements besides b, we get that

$$\big|\big\{b \in A \mid \mathrm{dist}_{\mathcal{A}}(a, b) = i\big\}\big| \leq d \cdot \mathrm{ar}(\sigma) \cdot \big|\big\{b \in A \mid \mathrm{dist}_{\mathcal{A}}(a, b) = i - 1\big\}\big|$$

for $1 \leq i \leq r$. Hence

$$\begin{aligned} |N_r^{\mathcal{A}}(a)| &\leq 1 + d \cdot \mathrm{ar}(\sigma) + \big(d \cdot \mathrm{ar}(\sigma)\big)^2 + \cdots + \big(d \cdot \mathrm{ar}(\sigma)\big)^r \\ &\leq (r + 1) \cdot \big(d \cdot \mathrm{ar}(\sigma)\big)^r \\ &\leq \big(2 \cdot d \cdot \mathrm{ar}(\sigma)\big)^r, \end{aligned}$$

where we use $r + 1 \leq 2^r$ in the last inequality. $\square$

Lemma 2.3.4. *Let σ be a relational signature and $d \in \mathbb{N}$. Let $r \in \mathbb{N}$. The number of different r-types of bounded degree d is finite and depends only on d, r and σ.*

Proof. For any number $n \in \mathbb{N}$, the number of non-isomorphic σ-structures with at most n elements is finite. This is the case as for every isomorphism class of σ-structures there is a representative $\mathcal{A}$ such that $A = \{1, \dots, k\}$, $k \leq n$. For two non-isomorphic σ-structure $\mathcal{A}, \mathcal{B}$ with $A = \{1, \dots, k\}$ and $B = \{1, \dots, k\}$ there must be $R \in \sigma$ such that $R^{\mathcal{A}} \neq R^{\mathcal{B}}$. But for every $R \in \sigma$ there is a finite number of ways to pick a set of tuples from $\{1, \dots, t\}^{\mathrm{ar}(R)}$ and since σ is finite the total number of non-isomorphic σ-structure with at most n elements is finite and depends only on n and σ.

For any r-type τ and $(\mathcal{B}, b) \in \tau$ Lemma 2.3.3 implies that $\mathcal{B}$ has at most $(2 \cdot d \cdot \mathrm{ar}(\sigma))^r$ elements. Furthermore, for any σ-structure $\mathcal{B}$ on at most $\big(2 \cdot d \cdot \mathrm{ar}(\sigma)\big)^r$ elements and any $b \in B$ there can be only one r-type τ with $(\mathcal{B}, b) \in \tau$. Since there are at most $\big(2 \cdot d \cdot \mathrm{ar}(\sigma)\big)^r$ choices for b the number of r-types of bounded degree d is finite and depends only on d, r and σ. $\square$

We can now define the following notions each capturing the appearance of types in a structure. Note that each of the notions defined in the following depends on some fixed ordering of the r-types.

Definition 2.3.5 (Histogram vector)**.** Let σ be a signature and $d \in \mathbb{N}$. For $\mathcal{A} \in C_d$ and $r \in \mathbb{N}$ we define the *histogram vector*, denoted $\mathrm{hist}_r(\mathcal{A})$, of $\mathcal{A}$ by

$$(\mathrm{hist}_r(\mathcal{A}))_i := |\{a \in A \mid (\mathcal{N}_r^{\mathcal{A}}(a), a) \in \tau_i\}|,$$

where $\tau_1, \dots, \tau_t$ is a list of all r-types of bounded degree d.

Definition 2.3.6 (Frequency vector)**.** Let σ be a signature and $d \in \mathbb{N}$. For $\mathcal{A} \in C_d$ and $r \in \mathbb{N}$ we define the *frequency vector*, denoted $\mathrm{freq}_r(\mathcal{A})$, of $\mathcal{A}$ by

$$(\mathrm{freq}_r(\mathcal{A}))_i := \frac{|\{a \in A \mid (\mathcal{N}_r^{\mathcal{A}}(a), a) \in \tau_i\}|}{|A|},$$

where $\tau_1, \dots, \tau_t$ is a list of all r-types of bounded degree d.

Definition 2.3.7 (Neighbourhood distribution)**.** Let σ be a signature and $d \in \mathbb{N}$. For $\mathcal{A} \in C_d$ and $r \in \mathbb{N}$ we define the *r-neighbourhood distribution* of $\mathcal{A}$ to be the function $\rho_{\mathcal{A},r} : \{X \subseteq \{\tau_1, \dots, \tau_t\}\} \to [0, 1]$, where $\tau_1, \dots, \tau_t$ is a list of all r-types of bounded degree d, defined by

$$\rho_{\mathcal{A},r}(X) := \frac{\sum_{\tau \in X} |\{a \in A \mid (\mathcal{N}_r^{\mathcal{A}}(a), a) \in \tau\}|}{|A|}$$

for any $X \subseteq \{\tau_1, \dots, \tau_t\}$.

Note that we omit σ and d from the notation as they will be clear from the context. Furthermore, if $\sigma = \sigma_{\mathrm{graph}}$ then we assume that the list of r-types $\tau_1, \dots, \tau_t$ does not contain types τ for which $\mathcal{B}$ does not have a symmetric and irreflexive edge relation for any $(\mathcal{B}, b) \in \tau$. We essentially omit all σ_{graph}-types which do not represent graphs.

While the histogram vector, the frequency vector and the neighbourhood distribution all contain essentially the same information, the respective notions are later used in different context where they are the appropriate notion to look at.

2.4 First-order logic

In this section we will introduce first-order logic (FO). A more detailed introduction to relational structures and FO can be found for example in the books [51, 52, 101].

Let σ be a signature. Let $\textsc{Var}$ be a infinite, countable set of variables. Typically we denote variables by $x, y, z, x_1, x_2, \ldots, y_1, y_2, \ldots$ or $z_1, z_2, \ldots$. The *alphabet of first-order logic* A_σ over σ is the set

$$A_\sigma := \textsc{Var} \cup \sigma \cup \{=\} \cup \{\forall, \exists\} \cup \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\} \cup \{(,)\} \cup \{,\},$$

where $\exists$ is the existential quantifier, $\forall$ is the universal quantifier, $\neg$ is the logical negation, $\wedge$ is the conjunction, $\vee$ is the disjunction, $\rightarrow$ is the logical implication and $\leftrightarrow$ is the logical biimplication. The *formulas of first-order logic* $\mathrm{FO}[\sigma]$ is the following recursively defined subset of $(A_\sigma)^*$

- $x_1 = x_2$ is a formula in $\mathrm{FO}[\sigma]$ for all $x_1, x_2 \in \textsc{Var}$.
- If $R \in \sigma$ is a relation symbol and $x_1, \ldots, x_{\mathrm{ar}(R)} \in \textsc{Var}$, then $R(x_1, \ldots, x_{\mathrm{ar}(R)}) \in \mathrm{FO}[\sigma]$ is a formula.
- If $\varphi \in \mathrm{FO}[\sigma]$ is a formula, then $\neg\varphi \in \mathrm{FO}[\sigma]$ is a formula.
- If $\varphi, \psi \in \mathrm{FO}[\sigma]$ are formulas, then $(\varphi * \psi) \in \mathrm{FO}[\sigma]$ is a formula for any logical connective $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.
- If $\varphi \in \mathrm{FO}[\sigma]$ is a formula, then $\forall x\varphi \in \mathrm{FO}[\sigma]$ and $\exists x\varphi \in \mathrm{FO}[\sigma]$ are formulas for any variable $x \in \textsc{Var}$.

Formulas of the form $x_1 = x_2$, where $x_1, x_2 \in \textsc{Var}$, or $R(x_1, \ldots, x_{ar(R)})$, where $R \in \sigma$ is a relation symbol and $x_1, \ldots, x_{ar(R)} \in \textsc{Var}$, are called *atomic formulas.* The *length* of a formula $\varphi \in \mathrm{FO}[\sigma]$, denoted by $\|\varphi\|$, is the length of φ as a string over the alphabet A_σ.

In a formula of the form $Qx\varphi$, where $Q \in \{\exists, \forall\}$, $x \in \textsc{Var}$ is a variable and $\varphi \in \mathrm{FO}[\sigma]$ is a formula, we say that every occurrence of x in φ is in the scope of Q. For any formula $\varphi \in \mathrm{FO}[\sigma]$ we call the variables in φ, that do not occur in the scope of any quantifier, free variables and denote the set of free variables in a formula φ by $\mathrm{free}(\varphi)$. We write $\varphi(x_1, \ldots, x_k)$ to specify that $\mathrm{free}(\varphi) \subseteq \{x_1, \ldots, x_k\}$. We call a formula $\varphi \in \mathrm{FO}[\sigma]$ an *FO-sentence* if $\mathrm{free}(\varphi) = \emptyset$.

We define the *semantics* of FO recursively as follows. For an atomic formula of the form $R(x_1, \dots, x_{\mathrm{ar}(R)})$, $R \in \sigma$ and $a_1, \dots, a_{\mathrm{ar}(R)} \in A$ we say that $\mathcal{A}$ *satisfies* $R(x_1, \dots, x_{\mathrm{ar}(R)})$ under the assignment of variables $x_i \mapsto a_i$ if $(A_1, \dots, a_{\mathrm{ar}(R)}) \in R^{\mathcal{A}}$. For an atomic formula of the form $x_1 = x_2$ and $a_1, a_2 \in A$ we say that $\mathcal{A}$ *satisfies* $x_1 = x_2$ under the assignment of variables $x_i \mapsto a_i$ if $a_1 = a_2$. This allows us to define recursively when a σ-structure $\mathcal{A}$ satisfies a formula $\varphi(x_1, \dots, x_k) \in \mathrm{FO}[\sigma]$ under variable assignment $x_i \mapsto a_i$ for $a_1, \dots, a_k \in A$ where the logical connectives as well as existential and universal quantification have the usual meaning. For a FO-formula $\varphi(x_1, \dots, x_k) \in FO[\sigma]$, $a_1, \dots, a_k \in A$ we write $\mathcal{A} \models \varphi[a_1, \dots, a_k]$ if φ is satisfied for $\mathcal{A}$ under the variable assignment $x_i \mapsto a_i$. For an FO-sentence φ we say that $\mathcal{A}$ is a *model of* φ if $\mathcal{A} \models \varphi$. We can now define equivalence of FO-formulas. Two formulas $\varphi(x_1, \dots, x_k)$ and $\psi(x_1, \dots, x_k)$ are called *equivalent*, denoted $\varphi \equiv \psi$, if for all σ-structures and elements $a_1, \dots, a_k \in A$ the following holds

$$\mathcal{A} \models \varphi[a_1, \dots, a_k] \iff \mathcal{A} \models \psi[a_1, \dots, a_k].$$

The following lemma can be proved by induction over the construction of FO (see e. g. [52]).

Lemma 2.4.1 (Isomorphism Lemma [52])**.** *Let σ be a relational signature and $\mathcal{A}$ and $\mathcal{B}$ isomorphic σ-structures. Then for every* FO$[\sigma]$*-sentence the following holds*

$$\mathcal{A} \models \varphi \iff \mathcal{B} \models \varphi.$$

The isomorphism lemma (lemma 2.4.1) implies that being a model of an FO-sentence is closed under isomorphism. This allows us to define properties by FO-sentences.

Definition 2.4.2 (Properties defined by FO-sentences)**.** Let σ be a relational structure and C a class of σ-structures. Then every FO-sentence $\varphi \in \mathrm{FO}[\sigma]$ defines a property $P_\varphi \subseteq C$ given by

$$P_\varphi := \{\mathcal{A} \in C \mid \mathcal{A} \models \varphi\}.$$

We use the following abbreviations.

- We use $x \neq y$ instead of $\neg x = y$.
- For any formula $\varphi(x)$ with free variable x we let $\exists^{\geq m} x\, \varphi(x)$ be short for
$$\exists x_1 \dots \exists x_m \big(\bigwedge_{1 \leq i < j \leq m} x_i \neq x_j \wedge \bigwedge_{1 \leq i \leq m} \varphi(x_i) \big).$$
- For any formula $\varphi(x)$ with free variable x we let $\exists^{\leq m} x\, \varphi(x)$ be short for $\neg \exists^{\geq m+1} x \varphi(x)$.
- For any formula $\varphi(x)$ with free variable x we let $\exists^{= m} x\, \varphi(x)$ be short for $\exists^{\geq m} x \varphi(x) \wedge \exists^{\leq m} x \varphi(x)$.

2.4.1 Normal forms of first-order logic

Normal forms for FO play a central role in the analysis of FO-formulas as well as our understanding of the expressive power of FO.

Disjunctive normal form

In this section we introduce disjunctive normal form for boolean combinations.

Let σ be a signature and $\varphi_1, \dots, \varphi_m$ be any set of formulas in $\mathrm{FO}[\sigma]$. The *boolean combinations* of $\varphi_1, \dots, \varphi_m$ is the following recursively defined subset of $\mathrm{FO}[\sigma]$.

– φ_i is a boolean combination of $\varphi_1, \dots, \varphi_m$ for every $1 \leq i \leq m$.

– If ψ is a boolean combination of $\varphi_1, \dots, \varphi_m$ then so is $\neg\psi$.

– If ψ and ψ' are boolean combinations of $\varphi_1, \dots, \varphi_m$ then so is $\psi * \psi'$ for $* \in \{\wedge, \vee\}$.

We say that a boolean combination φ of $\varphi_1, \dots, \varphi_m$ is in *dijunctive normal form (DNF)* if

$$\varphi = \bigvee_{i=1}^{n} \Big(\bigwedge_{j=1}^{\ell_i} \psi_{i,j} \wedge \bigwedge_{j=1}^{k_i} \neg\chi_{i,j} \Big),$$

for some $n \in \mathbb{N}$, $\ell_i, k_i \in \mathbb{N}$ for every $1 \leq i \leq n$ and $\psi_{i,j} \in \{\varphi_1, \dots, \varphi_m\}$ for every $i \in \{1, \dots, n\}$, $j \in \{1, \dots, \ell_i\}$ and $\chi_{i,j} \in \{\varphi_1, \dots, \varphi_m\}$ for every $i \in \{1, \dots, n\}$, $j \in \{1, \dots, k_i\}$. We call the formulas $\bigwedge_{j=1}^{\ell_i} \psi_{i,j} \wedge \bigwedge_{j=1}^{k_i} \neg\chi_{i,j}$ for $i \in \{1, \dots, n\}$ *clauses* of φ. We further refer to the formulas $\psi_{i,j}$ for every $i \in \{1, \dots, n\}$, $j \in \{1, \dots, \ell_i\}$ and $\neg\chi_{i,j}$ for every $i \in \{1, \dots, n\}$, $j \in \{1, \dots, k_i\}$ as *literals* of φ.

The following lemma can be shown with a standard, straight forward argument.

Lemma 2.4.3 (Disjunctive normal form [52])**.** *Let σ be a signature and $\varphi_1, \dots, \varphi_m \in \mathrm{FO}[\sigma]$. Then any boolean combination φ of $\varphi_1, \dots, \varphi_m$ is equivalent to a boolean combination of $\varphi_1, \dots, \varphi_m$ in DNF.*

Note that for any signature σ and any two formulas $\psi, \psi' \in \mathrm{FO}[\sigma]$ we have that $\psi \leftrightarrow \psi' \equiv (\psi \to \psi') \wedge (\psi' \to \psi)$ and $\psi \to \psi' \equiv \neg\psi \vee \psi'$. Therefore any quantifier-free formula $\varphi \in \mathrm{FO}[\sigma]$ is equivalent to a boolean combination of atomic formulas. This allows us to define DNF for quantifier-free formulas as follows. Let $\varphi \in \mathrm{FO}[\sigma]$ be a quantifier-free formula. We say that φ is in DNF if φ is a boolean combination of atomic formulas which is in DNF.

Prenex normal form and prefix classes

In this section we introduce prenex normal form which gives us a way to classify FO-formulas and properties defined by FO-sentences.

Let σ be a relational signature. An $\mathrm{FO}[\sigma]$-formula φ is in *prenex normal form* if φ is of the form $Q_1x_1Q_2x_2 \dots Q_kx_k\psi(x_1, x_2, \dots, x_k, y_1, \dots, y_\ell)$, where $Q_i \in \{\exists, \forall\}$, $x_i, y_j \in \textsc{Var}$ for all $i \in \{1, \dots, k\}$, $j \in \{1, \dots, \ell\}$ and ψ is an FO-formula not containing any quantifiers.

Theorem 2.4.4 (Prenex normal form [52])**.** *Let σ be a relational signature. For every formula $\varphi(x_1, \ldots, x_k) \in \mathrm{FO}[\sigma]$ there is a formula $\varphi'(x_1, \ldots, x_k) \in \mathrm{FO}[\sigma]$ in prenex normal form, which is equivalent to φ.*

First-order formulas can be classified by counting the number of alternations between existential and universal quantifiers in a prenex normal form of the formula.

We define $\Sigma_0 = \Pi_0$ to be the set of all quantifier-free FO-formulas. We then define recursively for all $i \in \mathbb{N}_{>0}$.

- $\Sigma_i := \left\{ \varphi(y_1, \ldots, y_\ell) \in \mathrm{FO}[\sigma] \;\middle|\; \begin{array}{l} \text{there is } k \in \mathbb{N} \text{ and } \psi(x_1, \ldots, x_k, y_1, \ldots, y_\ell) \in \Pi_{i-1} \\ \text{s.t. } \varphi \equiv \exists x_1 \ldots \exists x_k \psi(x_1, \ldots, x_k, y_1, \ldots, y_\ell) \end{array} \right\}.$
- $\Pi_i := \left\{ \varphi(y_1, \ldots, y_\ell) \in \mathrm{FO}[\sigma] \;\middle|\; \begin{array}{l} \text{there is } \ell \in \mathbb{N} \text{ and } \psi(x_1, \ldots, x_k, y_1, \ldots, y_\ell) \in \Sigma_{i-1} \\ \text{s.t. } \varphi \equiv \forall x_1 \ldots \forall x_k \psi(x_1, \ldots, x_k, y_1, \ldots, y_\ell) \end{array} \right\}.$
- $\Delta_i := \Sigma_i \cap \Pi_i$.

For a prefix class $\Gamma \in \{\Pi_i, \Sigma_i, \Delta_i\}$ we say that a property P is a Γ*-property* if $P = P_\varphi$ for a formula $\varphi \in \Gamma$.

Combining the above definition with Theorem 2.4.4 we get

$$\mathrm{FO}[\sigma] = \bigcup_{i \in \mathbb{N}} (\Sigma_i \cup \Pi_i).$$

Additionally the following chain of inclusions holds.

$$
\begin{tikzcd}[column sep=small]
 & \Sigma_0 \arrow[dr, phantom, sloped, "\subseteq"] & & \Sigma_1 \arrow[dr, phantom, sloped, "\subseteq"] & & \iddots & & & \Sigma_{2i-1} \arrow[dr, phantom, sloped, "\subseteq"] & & \Sigma_{2i} \arrow[dr, phantom, sloped, "\subseteq"] & & \iddots \\
\emptyset \arrow[ur, phantom, sloped, "\subseteq"] \arrow[dr, phantom, sloped, "\subseteq"] & & \Delta_1 \arrow[ur, phantom, sloped, "\subseteq"] \arrow[dr, phantom, sloped, "\subseteq"] & & \Delta_2 & & \ddots & \Delta_{2i-1} \arrow[ur, phantom, sloped, "\subseteq"] \arrow[dr, phantom, sloped, "\subseteq"] & & \Delta_{2i} \arrow[ur, phantom, sloped, "\subseteq"] \arrow[dr, phantom, sloped, "\subseteq"] & & \Delta_{2i+1} & \\
 & \Pi_0 \arrow[ur, phantom, sloped, "\subseteq"] & & \Pi_1 \arrow[ur, phantom, sloped, "\subseteq"] & & \ddots & \iddots & & \Pi_{2i-1} \arrow[ur, phantom, sloped, "\subseteq"] & & \Pi_{2i} \arrow[ur, phantom, sloped, "\subseteq"] & & \ddots
\end{tikzcd}
$$

Gaifman normal form

In 1981 Gaifman established that an FO-formula can only define local properties by proving that every FO-formula is equivalent to a formula in Gaifman normal form [63]. To introduce Gaifman normal form we need the following concepts of locality of formulas.

For any $r \in \mathbb{N}$ let $\mathrm{dist}_{\leq r}(x, y) \in \mathrm{FO}[\sigma]$ denote the formula, such that $\mathcal{A} \models \mathrm{dist}_{\leq r}[a, b]$ if and only if $\mathrm{dist}_{\mathcal{A}}(a, b) \leq r$ for any σ-structure $\mathcal{A}$ and $a, b \in A$. Note that we can express $\mathrm{dist}_{\mathcal{A}}(a, b) \leq r$ in FO, since we can express that there is a path in $G(\mathcal{A})$ from a to b of length exactly k for any $k \in \mathbb{N}$ and hence $\mathrm{dist}_{\leq r}(x, y)$ is a disjunction of formulas of that form. Let $\mathrm{dist}_{>r}(x, y) = \neg\, \mathrm{dist}_{\leq r}(x, y)$.

Let $\overline{x} = (x_1, \dots, x_k)$. We define *relativised quantifiers* $\exists y \in \mathcal{N}_r(\overline{x})\psi(y)$ and $\forall y \in \mathcal{N}_r(\overline{x})\psi(y)$ which are abbreviations for the formulas

$$\exists y\Big(\bigvee_{i=1}^{k} \mathrm{dist}_{\leq r}(y, x_i) \wedge \psi(y)\Big)\text{and}$$
$$\forall y\Big(\bigvee_{i=1}^{k} \mathrm{dist}_{\leq r}(y, x_i) \to \psi(y)\Big).$$

The formula $\varphi(\overline{x})$ is called r-*local* around $\overline{x}$, if all quantification is of the form $\exists y \in \mathcal{N}_r(\overline{x})$ or $\forall y \in \mathcal{N}_r(\overline{x})$. $\varphi(\overline{x})$ is called *local*, if there is an $r \in \mathbb{N}$ such that $\varphi(\overline{x})$ is r-local.

Let $r, \ell \in \mathbb{N}$, $\ell \geq 1$. A sentence $\varphi \in \mathrm{FO}[\sigma]$ is called *basic local* (with parameters r, ℓ) if φ is of the form

$$\varphi = \exists x_1 \dots \exists x_\ell \Big(\Big(\bigwedge_{1 \leq i < j \leq \ell} \mathrm{dist}_{>2r}(x_i, x_j)\Big) \wedge \Big(\bigwedge_{i=0}^{\ell} \psi(x_i)\Big)\Big),$$

where $\psi(x) \in \mathrm{FO}[\sigma]$ is r-local around x.

- A $\mathrm{FO}[\sigma]$-sentence φ is in *Gaifman normal form* (GNF) if φ is a boolean combination of basic local sentences.
- A $\mathrm{FO}[\sigma]$-formula $\varphi(x_1, \dots, x_k)$ is in GNF if φ is a boolean combination of basic local sentences and formulas that are local around $x_1, \dots, x_k$.

This allows us to formulate Gaifman's locality Theorem.

Theorem 2.4.5 (Gaifman's Locality Theorem [63])**.** *Let σ be a relational signature. For every formula $\varphi \in \mathrm{FO}[\sigma]$ there is a formula $\varphi' \in \mathrm{FO}[\sigma]$ in Gaifman normal form, that is equivalent to φ and* $\mathrm{free}(\varphi) = \mathrm{free}(\varphi')$.

Let us remark that there is an algorithm converting a given formula into GNF. However computing a GNF for a given sentence is very complex in the sense that there is no $k \in \mathbb{N}$ such that there is an algorithm with k-fold exponential running time for computing a GNF [40]. However, there is an algorithm which computes for a given sentence φ a sentence in GNF which is equivalent to φ on structures of bounded degree d in triple exponential time [85]. As we consider the problem of testing a property defined by a fixed sentence, this has no immediate consequences for our algorithms.

Hanf normal form

Hanf's Theorem [83] was proved in 1965 and later improved by Fagin, Stockmeyer and Vardi [54]. It roughly states that two structures $\mathcal{A}, \mathcal{B} \in C_d$ are equivalent on all sentences of quantifier rank at most k if there are $r, m \in \mathbb{N}$ such that for every r-type τ either τ appears the same number of times in both $\mathcal{A}$ and $\mathcal{B}$ or τ appears at least m times in both $\mathcal{A}$ and $\mathcal{B}$. Here the quantifier rank of a formula is the maximum number of nested quantifiers in the formula. This implies a strong normal form on structures of bounded degree, called Hanf normal form (HNF). We will define HNF in detail in the following.

Lemma 2.4.6. *Let $r \in \mathbb{N}$ and τ be any r-type. There is an FO-formula $\phi_\tau(x)$ such that $\mathcal{A} \models \phi_\tau(a)$ iff a has r-type τ for every σ-structure $\mathcal{A}$ and every element $a \in A$.*

Proof. Let $(\mathcal{B}, b) \in \tau$ be an r-ball with centre b in τ. Let further $B := \{b_0, b_1, \dots, b_\ell, \dots, b_k\}$ be the set of elements of $\mathcal{B}$ where $b_0 := b$ and $\{b_0, \dots, b_\ell\}$ is the set of all elements in $\mathcal{B}$ of distance less than r to b. $\phi_\tau(x)$ has to express that there are $k+1$ pairwise different elements $x =: x_0, \dots, x_k$ containing exactly the tuples in τ. Additionally $\phi_\tau(x)$ should express that there are no additional elements of distance less or equal than r to x. We express this by saying that all elements which are contained in a tuple with any elements from the set $\{x_0, \dots, x_\ell\}$ have to be amongst the elements $x_0, \dots, x_k$. We set

$$\begin{aligned}
&\phi_\tau(x_0) := \exists x_1, \dots, x_k \Big[\bigwedge_{0 \le i < j \le k} (x_i \neq x_j) \\
&\wedge \bigwedge_{R \in \sigma} \Big(\bigwedge_{\left(b_{i_1}, \dots, b_{i_{\mathrm{ar}(R)}}\right) \in R^{\mathcal{B}}} R(x_{i_1}, \dots, x_{i_{\mathrm{ar}(R)}}) \wedge \bigwedge_{\left(b_{i_1}, \dots, b_{i_{\mathrm{ar}(R)}}\right) \in B^{\mathrm{ar}(R)} \setminus R^{\mathcal{B}}} \neg R(x_{i_1}, \dots, x_{i_{\mathrm{ar}(R)}}) \Big) \\
&\wedge \bigwedge_{R \in \sigma} \forall y_1, \dots, y_{\mathrm{ar}(R)} \Big(\big(R(y_1, \dots, y_{\mathrm{ar}(R)}) \wedge \bigvee_{\substack{0 \le i \le \ell, \\ 1 \le j \le \mathrm{ar}(R)}} x_i = y_j \big) \to \bigwedge_{1 \le j \le \mathrm{ar}(R)} \big(\bigvee_{0 \le i \le k} x_i = y_j \big) \Big) \Big].
\end{aligned}$$

A simple argument shows that indeed $\mathcal{A} \models \phi_\tau(a)$ iff a has r-type τ for any σ-structure $\mathcal{A}$ and any element $a \in A$. $\square$

This allows us to define Hanf normal form. A *Hanf-sentence* is a sentence of the form $\exists^{\geq m} x \phi_\tau(x)$, for some $m \in \mathbb{N}_{>0}$, where τ is an r-type and $\phi_\tau(x)$ is the formula defined in Lemma 2.4.6. Here r is the *locality radius* of the Hanf-sentence. An FO sentence is in *Hanf normal form*, if it is a Boolean combination of Hanf sentences.

Two formulas $\phi(\bar{x})$ and $\psi(\bar{x})$ of signature σ are called *d-equivalent*, if they are equivalent on C_d, i. e. for all $\mathcal{A} \in C_d$ and $\bar{a} \in A^{|\bar{x}|}$ we have $\mathcal{A} \models \phi(\bar{a})$ iff $\mathcal{A} \models \psi(\bar{a})$.

The following theorem follows directly form Hanf's Theorem

Theorem 2.4.7 (Hanf normal form [83])**.** *Let $d \in \mathbb{N}$. Every FO-sentence is d-equivalent to a sentence in Hanf normal form.*

We want to remark that there is an algorithm which computes for a given sentence φ a d-equivalent sentence in HNF in triple exponential time [23].

Chapter 3

Background on property testing

In this section we introduce property testing. For a historical background and development of property testing, as well as recent advances in the field we refer to Section 4. We introduce property testing in a general way, focusing on what is in our opinion needed to consider property testing. This generality is needed to consider reductions between property testing problem. While all property testing models considered in this thesis fit the general framework introduced here, we do not claim that this is a full definition of property testing. There are various different settings, which fit the basic description of a probabilistic algorithm of testing an object for a property by looking at a set of samples and hence our definition of property testing might not cover all instances of property testing problems.

We introduce the general setting for property testing in Section 3.1. We introduce the two most important models for this thesis, the bounded-degree model for graphs and the bounded-degree model for relational structures, in Section 3.2. We provide a tester for testing subgraph freeness [69, 73] as an example of a typical property tester and an analysis of correctness in Section 3.3. We introduce proximity oblivious testers, a special type of property testers in Section 3.4. We further introduce some basic tools, i. e. property testing being closed under unions and local reductions, which are used throughout this thesis in Section 3.5 and Section 3.6.

3.1 The general setting

In this thesis we are interested in graph property testing. As property testing was first introduce for functions over finite fields, it is natural to consider graphs as functions representing their adjacency matrix or adjacency list. In the literature, especially when property testing is consider more generally then just for graphs, this is sometimes done. However, as we want to consider structural properties of graphs, we typically want properties to be invariant under graph isomorphisms. In this sense graph property testing deviates from property testing of functions and hence we would like to take a slightly different route of introducing property testing in this section without using the generality needed when relating the complexity of different property testing models via reductions.

Property testing is concerned with solving a relaxed version of the problem of deciding whether a given input has a certain property. In the following we will explain what classes of inputs, which relaxed decision problem and what model of computation we consider. As property testing can be considered in different contexts, we will introduce property testing models.

Input class: The class of inputs of a property tester can be any class of objects where each object of the class has a finite encoding. Therefore, let C be a class of objects and $\cong$ be an equivalence relation on C, which we call *isomorphism*. We call two objects $\mathcal{A}, \mathcal{B} \in C$ *isomorphic* if they are in the same equivalence class of $\cong$.

In the case of graphs we consider $\cong$ to be the graph isomorphism relation and on the class of functions or strings we let $\cong$ be the equivalence relation in which each equivalence class contains only one object.

We require objects to be finite in the sense that there should exist a finite encoding of the objects in C. An *encoding* is an injective function $\mathsf{code} : C \to \Sigma^*$, where Σ is a finite alphabet and Σ^* the Kleene closure of Σ, having the property that $|\mathsf{code}(\mathcal{A})| = |\mathsf{code}(\mathcal{B})|$ for all isomorphic objects $\mathcal{A}, \mathcal{B} \in C$. We define the *size* of an object $\mathcal{A} \in C$, denoted $|\mathcal{A}|$, to be the length of its encoding as a string over the alphabet Σ. We set $C|n := \{\mathcal{A} \in C \mid |\mathcal{A}| = n\}$ to be the subset of C of objects of a certain size $n \in \mathbb{N}$. Note that the existence of an injective encoding $\mathsf{code} : C \to \Sigma^*$, where Σ is a finite alphabet, implies that the class C is a countable set.

Relaxed decision problem: A *property* on C is a subset P of C closed under isomorphism. We say that an objects of C has the property P if and only if it is contained in P. In classical complexity theory we consider the following decision problem.

Decision problem for P on C
Input: An object $\mathcal{A}$ of C. **Aim:** Decide whether $\mathcal{A}$ has the property P.

In property testing we want to solve a relaxation of this classical decision problem. This requires us to have a *distance function* $\text{dist} : C \times C \to \mathbb{R} \cup \{\infty\}$ on the set of objects C, for which the following properties hold for all objects $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ of C.

(D1) $\mathrm{dist}(\mathcal{A},\mathcal{B}) = 0$ if and only if $\mathcal{A}$ and $\mathcal{B}$ are isomorphic. (definiteness)

(D2) $\mathrm{dist}(\mathcal{A},\mathcal{B}) = \mathrm{dist}(\mathcal{B},\mathcal{A})$. (symmetry)

(D3) $\mathrm{dist}(\mathcal{A},\mathcal{C}) \leq \mathrm{dist}(\mathcal{A},\mathcal{B}) + \mathrm{dist}(\mathcal{B},\mathcal{C})$. (triangle inequality)

Note that dist can be considered as being a metric, where in addition we allow objects having distance infinity.

Remark 3.1.1. Note that there always exists such a distance function. Since isomorphism is an equivalence relation, we can always pick the trivial distance function for every class of objects C, which is defined by

$$\mathrm{dist}(\mathcal{A},\mathcal{B}) := \begin{cases} 0, & \text{if } \mathcal{A} \text{ and } \mathcal{B} \text{ are isomorphic} \\ \infty, & \text{otherwise.} \end{cases}$$

The function dist allows us to define how far we consider an object from having a property P. For a property P we define the function $\mathrm{dist}_P : C \to \mathbb{R} \cup \{\infty\}$ by

$$\mathrm{dist}_P(\mathcal{A}) := \begin{cases} \infty, & \text{if } P = \emptyset \\ \min_{\mathcal{B}\in P}\{\mathrm{dist}(\mathcal{A},\mathcal{B})\}, & \text{otherwise} \end{cases}.$$

Let $\epsilon \in (0,1)$. We want to consider an object $\mathcal{A}$ of C as being ϵ-*close* to the property P if the distance of $\mathcal{A}$ to the property P in relation to its size is smaller than ϵ, i. e. if

$$\frac{\mathrm{dist}_P(\mathcal{A})}{|\mathcal{A}|} \leq \epsilon.$$

We say that the object $\mathcal{A}$ is ϵ-far from having the property P, if $\mathcal{A}$ is not ϵ-close to having P. For any property P let

$$\epsilon\text{-far}_C(P) := \{\mathcal{A} \in C \mid \mathcal{A} \text{ is } \epsilon\text{-far from } P\}$$

be the set of all structures, that are ϵ-far from P.

Fixing two parameters $\epsilon \in (0,1)$, called the *proximity parameter*, and $\delta \in (0, \frac{1}{2})$, called the *error probability*, we can formulate the following relaxation of the decision problem defined above.

RELAXATION OF THE DECISION PROBLEM FOR P ON C WITH PARAMETERS ϵ, δ
Input: An object $\mathcal{A}$ of C. **Aim:** Make a conjecture on whether $\mathcal{A}$ has the property P, where the conjecture has to be correct with probability at least $1 - \delta$ if $\mathcal{A}$ is in $P \cup \epsilon\text{-far}_C(P)$.

In property testing we are concerned with solving this relaxed decision problem.

Model of computation: In order to define what a property tester is, we have to consider the model of computation used. A property tester is given *oracle access* to its input object. We can think of *oracle access* to an object $\mathcal{A}$ as a function $\text{ans}_{\mathcal{A}} : Q_{\mathcal{A}} \to A_{\mathcal{A}}$, where the domain $Q_{\mathcal{A}}$ and codomain $A_{\mathcal{A}}$ are finite, dependent on $\mathcal{A}$ and are property testing model specific. We call $Q_{\mathcal{A}}$ the set of *permissible queries* for $\mathcal{A}$, $A_{\mathcal{A}}$ the set of *possible answers*, an element $q \in Q_{\mathcal{A}}$ a *query* and $\text{ans}_{\mathcal{A}}(q)$ the *answer to the query* q. In order for the property tester to determine the set of permissible queries $Q_{\mathcal{A}}$ it is given some *auxiliary information* $\text{aux}_{\mathcal{A}} \in \mathbb{N}^{\ell}$ as input where $\ell \in \mathbb{N}$ depends on the model. Here $\text{aux}_{\mathcal{A}}$ should not encode the object $\mathcal{A}$ and hence typically the size of $\text{aux}_{\mathcal{A}}$ should be in $o(|\mathcal{A}|)$. Specific to the model there has to be a way to determine the set of permissible queries $Q_{\mathcal{A}}$ from the auxiliary information. Furthermore, the answer to a query typically should be considered to take a constant amount of time. Additionally, we would like the answers to all permissible queries to identify the input object $\mathcal{A} \in C$, that is $\{(q, \text{ans}_{\mathcal{A}}(q)) \mid q \in Q_{\mathcal{A}}\} \neq \{(q, \text{ans}_{\mathcal{B}}(q)) \mid q \in Q_{\mathcal{B}}\}$ for $\mathcal{A}, \mathcal{B} \in C$, $\mathcal{A} \neq \mathcal{B}$.

We define a *query access model* of C to be a tuple

$$\text{QA} = \Big(\ell, (\text{aux}_{\mathcal{A}})_{\mathcal{A}\in C}, (Q_{\mathcal{A}})_{\mathcal{A}\in C}, (A_{\mathcal{A}})_{\mathcal{A}\in C}, (\text{ans}_{\mathcal{A}})_{\mathcal{A}\in C}\Big),$$

where $\ell \in \mathbb{N}$, $\text{aux}_{\mathcal{A}} \in \mathbb{N}^{\ell}$ and $\text{ans}_{\mathcal{A}} : Q_{\mathcal{A}} \to A_{\mathcal{A}}$ for every $\mathcal{A} \in C$, where $Q_{\mathcal{A}}, A_{\mathcal{A}}$ are finite sets with the properties described above.

Property testing We will first introduce what we understand under a property testing model before we introduce property testing.

Definition 3.1.2 (Property testing model)**.** A property testing model is a tuple

$$\Big(C, \cong, \Sigma, \mathsf{code}, \text{dist}, \text{QA}\Big),$$

where C is a class of objects, $\cong$ is an equivalence relation on C, $\mathsf{code} : C \to \Sigma^*$ is an encoding of C, $\mathrm{dist} : C \times C \to \mathbb{R} \cup \{\infty\}$ is a distance function with properties (D1), (D2) and (D3) and QA is a query access model of C.

From now on, if the property testing model $\big(C, \cong, \Sigma, \mathsf{code}, \mathrm{dist}, \mathrm{QA}\big)$ is uniquely identified by the class C then we do not explicitly state which property testing model we consider.

Definition 3.1.3 (Property tester)**.** Let P be a property on C. Let $\epsilon \in (0, 1)$ and $\delta \in (0, \frac{1}{2})$. We call a probabilistic algorithm T, which is given auxiliary information $\mathrm{aux}_{\mathcal{A}}$ about the input object $\mathcal{A}$ and has oracle access to $\mathcal{A}$ via queries, an *ϵ-tester* for P on C with error probability δ, if T

– accepts $\mathcal{A}$ with probability $1 - \delta$ if $\mathcal{A} \in P$.

– rejects $\mathcal{A}$ with probability $1 - \delta$ if $\mathcal{A} \in \epsilon\text{-far}_C(P)$.

As is apparent from the definition, a property tester solves the relaxed decision problem defined above. We are also interested in property testers, which always make the correct decision for objects having the property.

Definition 3.1.4 (One-sided error property tester)**.** Let P be a property on C. A property tester T is called a one-sided error tester if T accepts every $\mathcal{A}$ of C which has property P with probability 1. In contrast, we sometimes call property testers *two-sided error testers* to explicitly express that they are not required to be one-sided error testers.

We consider the complexity of a property testers in terms of the amount of queries it makes.

Definition 3.1.5 (Query complexity)**.** Let P be a property on C. Let $\epsilon \in (0, 1)$ and $\delta \in (0, \frac{1}{2})$. Let T be a property tester for P on C with parameters ϵ, δ. The *query complexity* of T is a function $f : \mathbb{N} \to \mathbb{N}$, where $f(n)$ gives the maximum number of queries T makes when testing any structure $\mathcal{A} \in C$ of size n.

Remark 3.1.6. We consider a query to take constant time. Therefore the query complexity of a property tester T only provides a lower bound for the running time of the algorithm but it can also be significantly worse. The query complexity of a tester should therefore be understood as a measure of the portion of the input object we need to look at to be able to make a good conjecture about the object having the property and is therefore an interesting invariant of a property tester.

Next we would like to remark that every given error probability can be improved by repeating the tester and deciding according to the majority of outcomes. We formalise this in the following Lemma.

Lemma 3.1.7. *Let P be a property on C, $\epsilon \in (0, 1)$, $\delta, \delta' \in (0, \frac{1}{2})$, $\delta > \delta'$ and $f : \mathbb{N} \to \mathbb{N}$. For every ϵ-tester T with error probability δ and query complexity f there is $k := k(\delta, \delta') \in \mathbb{N}$ and an ϵ-tester T' with error probability δ' and query complexity $k \cdot f$.*

Proof. Let T' be the tester which repeats T exactly $k := 2 \cdot \lceil \frac{1}{\delta \cdot \delta'} \rceil + 1$ times and decides whether to accept an input if T accepts the input on more than half of the repetitions.

We say that T or T' is successful for input $\mathcal{A} \in P \cup \epsilon\text{-far}_C(P)$ if T or T' respectively accepts $\mathcal{A}$ if $\mathcal{A} \in P$ and rejects $\mathcal{A}$ otherwise. Let $\mathcal{A} \in P \cup \epsilon\text{-far}_C(P)$. We need to argue that T' is successful for input $\mathcal{A}$ with probability at least $1 - \delta'$. Let X_i be the indicator random variable which is 1 in the event of the i-th repetition of T being successful and -1 otherwise. Let $Y_i := X_i - \mathbb{E}[X_i] = X_i - \delta$ and $Y := \sum_{i=1}^{k} Y_i$. Using Chernoff bounds (see Theorem A.1.13 in [13]) we get that the probability of more than half of the repetitions of T being successful is

$$\begin{aligned}
\mathbb{P}[T' \text{ is not successful }] &= \mathbb{P}[Y < -k\delta] \\
&< \exp(-\frac{k^2\delta^2}{2\delta k}) \\
&\leq \exp(-\frac{1}{\delta'}) \\
&\leq \delta'.
\end{aligned}$$

Hence T' is an ϵ-tester with error probability δ' and query complexity kf. □

The dependency of the query complexity on δ and δ' in Lemma 3.1.7 may be improved using techniques from approximation algorithms.

The above lemma shows that the precise error probability is in fact not relevant to the asymptotic growth of the query complexity and hence from now on, unless stated otherwise, the error probability is always $\frac{1}{3}$. An error probability of $\frac{1}{3}$ is an arbitrary but widely used choice.

Definition 3.1.8 (Uniform/non-uniform Testability)**.** Let C be a class of objects, $P \subseteq C$ be a property on C and $f : \mathbb{N} \to \mathbb{N}$ a function.

- P is *uniformly testable* on C with query complexity f, if for every $\epsilon \in (0,1)$ there exist an ϵ-tester for P on C with query complexity f.
- P is *(non-uniformly) testable* on C with query complexity f, if for every $\epsilon \in (0,1)$ and $n \in \mathbb{N}$ there exist an ϵ-tester for $P|n$ on $C|n$ with query complexity $f(n)$.
- We say that P is *uniformly/(non-uniformly) testable* on C if P is uniformly/(non-uniformly) testable on C with query complexity f and f is a constant function.

As indicated in the definition, by testable we mean non-uniformly testable.

Remark 3.1.9. From the definition it is clear that if a property $P \subseteq C$ is uniformly testable on C then P is also testable on C.

For further reading on property testing we refer the reader to the book [69] by Goldreich or one of the surveys [57, 116].

3.2 The bounded degree model

In this section we will introduce the bounded degree model for simple, undirected graphs and for relational structures. In both models the key is to encode graphs/structures in an adjacency list, which keeps the size of the encoding linear in the number of vertices as the degree is bounded by a constant. We further assume that every vertex/element can be stored in constant space.

The bounded degree model for graphs: In this section we introduce the bounded degree model for property testing of simple, undirected graphs from [73].

Let $d \in \mathbb{N}$ and $\mathcal{C}_d$ be the class of graphs of bounded degree d with graph isomorphism as equivalence relation. We consider d to be a constant which is known to any property tester testing a property on $\mathcal{C}_d$.

We will encode a graph into an adjacency list. Note that this encoding depends on the choice of an order on $V(G)$ and an order on the set of all neighbours of any vertex in $V(G)$. We will therefore assume that $V(G) = [n]$ where $n := |V(G)|$. Since a graph is encoded by an adjacency list and every vertex has at most d neighbours we set the size of the encoding of a graph G to be $d \cdot n$.

The following definition of the distance between graphs satisfies the properties (D1), (D2) and (D3), which can be easily verified using the definition of isomorphisms of graphs.

Definition 3.2.1 (Distance between graphs)**.** Let G and G' be two graphs. We allow two types of edge modifications; deleting an edge from $E(G)$ or $E(G')$ and adding an edge to $E(G)$ or $E(G')$. We define the distance between G and G', denoted $\mathrm{dist}(G, G')$, to be the minimum number of edge modifications needed to make G isomorphic to G' or ∞ if we can not make G and G' isomorphic by edge modifications, i. e.

$$\mathrm{dist}(G, G') = \min \Big\{ |E| \;\Big|\; E \subseteq \{e \subseteq V(G) \mid |e| = 2\}, \big(V(G), E(G) \triangle E\big) \cong G' \Big\} \cup \{\infty\}.$$

Note that $\mathrm{dist}(G, G') = \infty$ if and only if $|V(G)| \neq |V(G')|$. Furthermore, if a graph G is ϵ-far from a property $\mathcal{P} \subseteq \mathcal{C}_d$ then by definition it takes more than ϵdn edge modifications to make G have the property $\mathcal{P}$. On the other hand, if G is ϵ-close to $\mathcal{P}$ then there is a set $E \subseteq \{x \subseteq V(G) \mid |x| = 2\}$ of size at most ϵdn such that the graph $(V(G), E(G) \triangle E) \in \mathcal{P}$.

A property tester in this model gets the number of vertices $n := |V(G)|$ as auxiliary information. This enables the tester to determine the set of permissible queries $Q_G := [n] \times [d]$.

Then $A_G := [n]$ and $\text{ans}_G : Q_G \to A_G$ is defined by

$$\text{ans}_G(i,j) := \begin{cases} k & \text{if } k \in [n] \text{ is the } j\text{-th neighbour of } i, \\ \perp & \text{if } i \text{ has less then } j \text{ neighbours} \end{cases}$$

for $i \in [n]$, $j \in [d]$.

The bounded degree model for relational structures: The following property testing model for bounded degree relational structures was defined in [86] and extends the bounded degree model for simple, undirected graphs from [73] and the bidirectional model for directed graphs from [34].

For the rest of this section, let $\sigma = \{R_1, \dots, R_\ell\}$ be a relational signature. Let $d \in \mathbb{N}$ and C_d be a class of σ-structures of bounded degree d with isomorphism of relational structures as equivalence relation. We assume d and σ to be known to any property tester operating on C_d.

We will use an encoding of σ-structures similar to adjacency lists.

Definition 3.2.2 (Encoding of bounded degree relational structures)**.** Let $\mathcal{A}$ be a σ-structure and $n := |A|$. We assume that $A = [n]$. We encode the σ-structure $\mathcal{A}$ into a tuple $L_\mathcal{A} = (L_1, \dots, L_n)$, where $L_i \in \Big(\{\perp\} \cup \bigcup_{i=1}^{\ell} \{R_i\} \times [n]^{\text{ar}(R_i)}\Big)^d$ is defined by

$$(L_i)_j := \begin{cases} \perp & \text{if } \deg_\mathcal{A}(i) < j, \\ (R, \overline{u}) & \text{if } \overline{u} \text{ is the } j\text{-th tuple containing } a_i \text{ and } \overline{u} \in R^\mathcal{A}, \end{cases}$$

for every $i \in [n]$, $j \in [d]$. Note that $L_\mathcal{A}$ depends on the choice of an order on A and an order on the set of all tuples containing a specific element of A.

Note that for a σ-structure $\mathcal{A}$ with a degree bounded by $d \in \mathbb{N}$ the size of the encoding is bound by $nd \cdot \text{ar}(\sigma)$. For simplicity we let $|\mathcal{A}| = nd$ which is not a restriction as $\text{ar}(\sigma)$ is a constant and can be "hidden" in ϵ.

Definition 3.2.3 (Distance between relational structures)**.** Let $\mathcal{A}$ and $\mathcal{B}$ be two σ-structures. We allow two types of tuple modifications; deleting a tuple from $R_i^\mathcal{A}$ or $R_i^\mathcal{B}$ and adding a tuple to $R_i^\mathcal{A}$ or $R_i^\mathcal{B}$ for any $i \in \{1, \dots, \ell\}$. We define the distance between $\mathcal{A}$ and $\mathcal{B}$, denoted $\text{dist}(\mathcal{A}, \mathcal{B})$, to be the minimum number of tuple modifications we need to transform $\mathcal{A}$ into a structure isomorphic to $\mathcal{B}$ or ∞ if we can not make $\mathcal{A}$ and $\mathcal{B}$ isomorphic by tuple modifications.

It can be easily seen that this distance satisfies properties (D1), (D2) and (D3). In addition for any two σ-structures $\mathcal{A}$ and $\mathcal{B}$ we have that $\text{dist}(\mathcal{A}, \mathcal{B}) = \infty$ if and only if $|A| \neq |B|$.

A property tester in this model gets the number of elements $n := |A|$ of the input structure $\mathcal{A}$ as auxiliary information. This enables the tester to determine the set of permissible queries $Q_{\mathcal{A}} := [n] \times [d]$. The set of query answers $A_{\mathcal{A}}$ is the set $\{\perp\} \cup \bigcup_{i=1}^{\ell} \{R_i\} \times [n]^{\mathrm{ar}(R_i)}$ and $\mathrm{ans}_{\mathcal{A}} : Q_{\mathcal{A}} \to A_{\mathcal{A}}$ is defined by $\mathrm{ans}_{\mathcal{A}}(i,j) := (L_i)_j$ for every $i \in [n]$ and $j \in [d]$, where $L_{\mathcal{A}} = (L_1, \dots, L_n)$ is the encoding of $\mathcal{A}$ defined in Definition 3.2.2.

3.3 An example of a property tester: Testing subgraph freeness

The aim of this section is to develop an intuition for the way property testers operate. For this we consider the problem of testing subgraph freeness. This problem is testable and in fact a tester was given in [73]. The tester explained here is similar to the tester for subgraph freeness described in [69].

Let $d \in \mathbb{N}$ be fixed. Consider some graph $F \in \mathcal{C}_d$. We say that a graph G is F-free if G does not contain an induced subgraph isomorphic to F. Let $\mathcal{P}_F \subseteq \mathcal{C}_d$ be the subset of all graphs that are F-free.

Theorem 3.3.1 (Theorem 5.2 from [73])**.** *For every $F \in \mathcal{C}_d$ the property $\mathcal{P}_F$ is testable on $\mathcal{C}_d$.*

Proof. We assume that F contains more than one vertex as testing whether a graph is K_1-free can be trivially done by rejecting every graph with at least one vertex. For simplified analysis we also assume that F is connected. Generalisation to graphs with more than one connected component is straight forward.

Let r be the radius of F. We call a vertex $v \in V(F)$ with $\mathrm{dist}_F(v,u) \leq r$ for every $u \in V(F)$ a *centre* of F. To obtain a property tester for $\mathcal{P}_F$ let us first consider the following algorithm.

Algorithm 1: $\textsc{Find}_F$

Query access: $G \in \mathcal{C}_d$
Input : $n := |V(G)|$

Sample a vertex $v \in V(G)$ uniformly at random;
Do a breadth-first search to depth r from v to obtain some subgraph H of G;
if *H contains F as a subgraph* **then**
    Reject G;
else
    Accept G;
end

Let $\epsilon \in (0,1)$. We now let T be the algorithm that, given query access to a graph G, repeats $\textsc{Find}_F$ $k := \lceil \frac{4}{\epsilon} \rceil$ times, accept G if every repetition of $\textsc{Find}_F$ accepts G and rejects G otherwise.

Let us first observe that if $G \in \mathcal{P}_F$ then the above algorithm will accept G (T is a one-sided error tester). We use the following claim.

Claim 1. If G is ϵ-far from $\mathcal{P}_F$ then it contains at least ϵn vertices which are the centre of a copy of F.

Proof of Claim 1. Assume G contains less than ϵn vertices which are the centre of a copy of F. Isolating every centre of a copy of F takes at most ϵdn edge removals and results in a graph which is F-free. Hence G is ϵ-close to $\mathcal{P}_F$ which proves the claim by contraposition. ■

Now assume G is ϵ-far from $\mathcal{P}_F$. By Claim 1 we know that G contains at least ϵn vertices which are the centre of a copy of F. Hence $\textsc{Find}_F$ rejects G with probability ϵ. Let X_i be the indicator random variable which is 1 in the event of the i-th repetition of $\textsc{Find}_F$ rejects G and -1 otherwise. Let $Y_i := X_i - \mathbb{E}[X_i] = X_i - \epsilon$ and $Y := \sum_{i=1}^{k} Y_i$. Using Chernoff bounds (see Theorem A.1.13 in [13]) we get that the probability of all repetition of $\textsc{Find}_F$ accepting G is

$$\begin{aligned}
\mathbb{P}[\sum_{i=1}^{k} X_i = -k] &= \mathbb{P}[Y < -k] \\
&< \exp(-\frac{k^2}{2\epsilon k}) \\
&\leq \exp(-2) \\
&\leq \frac{1}{3}.
\end{aligned}$$

Hence T accepts G with probability at least $\frac{2}{3}$.

Furthermore, the query complexity of T is clearly constant since the query complexity of $\textsc{Find}_F$ depends only on r and d. This proves that T is an ϵ-tester for $\mathcal{P}_F$. Hence $\mathcal{P}_F$ is testable. □

We would like to remark that what is shown in Claim 1 is sometimes referred to as a "removal lemma". Proving the appropriate "removal lemma" is the key part in the analysis of a property tester.

3.4 Proximity oblivious testing

In proximity oblivious testing we consider particularly simple property testers where some basic test is repeated a number of times. Here the basic test does not depend on the proximity parameter ϵ and the probability of rejecting an input behaves like a monotonically non-decreasing function η of the distance of the input to the property. We call such a basic test a proximity oblivious tester (POT). Repeating a POT some number in $\Theta\left(\frac{1}{\eta(\epsilon)}\right)$ times yields an ϵ-tester. Proximity oblivious testing intuitively is one-sided considering that the rejection probability should be proportionate to the distance of an input to the property. Here we will only consider one-sided error POTs and hence sometimes omit stating that a POT is one-sided error. However, one can also consider two-sided error POTs (see e. g. [69]).

Definition 3.4.1 ((One-sided error) POT)**.** Let C be a class of object and $P = \bigcup_{n\in\mathbb{N}} P|n$ be a property on C. Let $\eta : (0,1] \to (0,1]$ be a monotonically non-decreasing function. A *proximity-oblivious tester (POT)* with detection probability η for $P|n$ is a probabilistic algorithm which, given query access to a structure $\mathcal{A} \in C|n$

- accepts $\mathcal{A}$ with probability 1 if $\mathcal{A} \in P|n$.
- rejects $\mathcal{A}$ with probability at least $\eta\big(\operatorname{dist}_{P|n}(\mathcal{A})\big)$ if $\mathcal{A} \notin \mathcal{P}_n$.

The *query complexity* of a POT is the maximum number of queries the POT makes as a function in the size of the input. We say that a property $\mathcal{P}$ is *proximity oblivious testable* if for every $n \in \mathbb{N}$, there exists a monotonically non-decreasing function $\eta : (0,1] \to (0,1]$ and a POT for $\mathcal{P}|n$ of constant query complexity with detection probability η.

Example 3.4.2. The algorithm $\textsc{test}_F$ from Section 3.3 is a POT with detection probability $\eta(\epsilon) = \epsilon$ as if G has distance ϵ from $\mathcal{P}_F$ then G contains at least $\epsilon|V(G)|$ centres of copies of F and hence $\textsc{Find}_F$ rejects G with probability ϵ. We further demonstrate in the proof of Theorem 3.3.1 how to obtain a property tester from the proximity oblivious tester $\textsc{Find}_F$ for $\mathcal{P}_F$ by repeated application of $\textsc{Find}_F$.

We refer to [69] for a proof of the statement that repeating a POT $\Theta(\frac{1}{\eta(\epsilon)})$ times yields a property tester, which implies the following theorem.

Theorem 3.4.3 (Theorem 1.9 in [69])**.** *Let C be a class of object and P be a property on C. If P has a one-sided error POT with query complexity $f(n)$ and with detection probability η then P has a one-sided error ϵ-tester with query complexity $f' \in \mathcal{O}\Big(\frac{f(n)}{\eta(\epsilon)}\Big)$ for every $\epsilon \in (0,1)$.*

In particular, if the property P is proximity oblivious testable then P is testable.

3.5 Closure properties of testability

In this section we show that property testing is closed under union, but is not closed under intersection nor taking the complement. Union, intersection and complements are of particular interest considering testing properties defined in logic. We obtain that if P_φ and P_ψ are testable properties for two FO-sentences φ and ψ then $P_{\varphi\vee\psi}$ is also testable. Let $\big(C, \cong, \Sigma, \mathsf{code}, \mathrm{dist}, \mathrm{QA}\big)$ be any property testing model.

Lemma 3.5.1. *Let $P, P' \subseteq C$ be properties on C. Let $\epsilon \in [0,1]$ and $f, f' : \mathbb{N} \to \mathbb{N}$. If there is an ϵ-tester for P on C with query complexity f and there is an ϵ-tester for P' on C with query complexity f', then there is an ϵ-tester for the property $P \cup P'$ on C with query complexity $7(f + f')$.*

Proof. Let T be an ϵ-tester for P with query complexity f and T' an ϵ-tester for P' with query complexity f'. We first obtain an ϵ-tester $\tilde{T}$ which on input $\mathcal{A} \in C$ operates as follows. $\tilde{T}$ repeats the tester T seven times with input $\mathcal{A}$ and decides whether to accept $\mathcal{A}$ depending on the majority of outcomes. If $\mathcal{A} \in P \cup$ -far$_C(P)$ the tester $\tilde{T}$ makes the correct decision with probability

$$\binom{7}{4}\left(\frac{2}{3}\right)^4\left(\frac{1}{3}\right)^3 + \binom{7}{5}\left(\frac{2}{3}\right)^5\left(\frac{1}{3}\right)^2 + \binom{7}{6}\left(\frac{2}{3}\right)^6\frac{1}{3} + \left(\frac{2}{3}\right)^7 \geq 0.82.$$

Hence $\tilde{T}$ is an ϵ-tester with error probability 0.82. In the same way we obtain an ϵ-tester $\tilde{T}'$ from T' with error probability 0.82. Now we let $T_\cup$ be an algorithm that, given oracle access to an object $\mathcal{A} \in C$, operates in the following way; $T_\cup$ runs $\tilde{T}$ and $\tilde{T}'$ on $\mathcal{A}$ and accepts $\mathcal{A}$ if either $\tilde{T}$ or $\tilde{T}'$ accept $\mathcal{A}$.

First confirm that the query complexity of $T_\cup$ is indeed $7(f + f')$. To prove that $T_\cup$ is an ϵ-tester let us first assume that $\mathcal{A} \in P \cup P'$. Since the decisions of $\tilde{T}$ and $\tilde{T}'$ are independent, the probability of $T_\cup$ accepting $\mathcal{A}$ satisfies the following inequalities

$$\begin{aligned}
\mathbb{P}[T_\cup \text{ accepts } \mathcal{A}] &\geq \mathbb{P}\begin{bmatrix} \tilde{T} \text{ accepts } \mathcal{A} \text{ and} \\ \tilde{T}' \text{ rejects } \mathcal{A} \end{bmatrix} + \mathbb{P}\begin{bmatrix} \tilde{T} \text{ rejects } \mathcal{A} \text{ and} \\ \tilde{T}' \text{ accepts } \mathcal{A} \end{bmatrix} + \mathbb{P}\begin{bmatrix} \tilde{T} \text{ accepts } \mathcal{A} \text{ and} \\ \tilde{T}' \text{ accepts } \mathcal{A} \end{bmatrix} \\
&\geq \mathbb{P}[\tilde{T} \text{ accepts } \mathcal{A}] + \mathbb{P}[\tilde{T}' \text{ accepts } \mathcal{A}] - \mathbb{P}[\tilde{T} \text{ accepts } \mathcal{A}] \cdot \mathbb{P}[\tilde{T}' \text{ accepts } \mathcal{A}] \\
&\geq \frac{2}{3},
\end{aligned}$$

where the third inequality holds because $\mathbb{P}[\tilde{T} \text{ accepts } \mathcal{A}] \geq \mathbb{P}[\tilde{T} \text{ accepts } \mathcal{A}] \cdot \mathbb{P}[\tilde{T}' \text{ accepts } \mathcal{A}]$, $\mathbb{P}[\tilde{T}' \text{ accepts } \mathcal{A}] \geq \mathbb{P}[\tilde{T} \text{ accepts } \mathcal{A}] \cdot \mathbb{P}[\tilde{T}' \text{ accepts } \mathcal{A}]$ and at least one of the two probabilities $\mathbb{P}[\tilde{T} \text{ accepts } \mathcal{A}]$ and $\mathbb{P}[\tilde{T}' \text{ accepts } \mathcal{A}]$ is at least $\frac{2}{3}$ depending on whether $\mathcal{A} \in P \setminus P'$ or $\mathcal{A} \in P' \setminus P$ or $\mathcal{A} \in P \cap P'$.

Now let $\mathcal{A}$ be ϵ-far from having property $P \cup P'$. Since $\mathcal{A}$ has distance greater than $\epsilon|\mathcal{A}|$ from any structure in P and any structure in P', $\mathcal{A}$ is ϵ-far from P and ϵ-far from P' (see Figure 3.1). Since the decision of $\tilde{T}$ and $\tilde{T}'$ are independent this means for the probability of $T_\cup$ rejecting $\mathcal{A}$

$$\begin{aligned}
\mathbb{P}[T_\cup \text{ rejects } \mathcal{A}] &\geq \mathbb{P}\begin{bmatrix} \tilde{T} \text{ rejects } \mathcal{A} \text{ and} \\ \tilde{T}' \text{ rejects } \mathcal{A} \end{bmatrix} \\
&\geq 0.82^2 \\
&\geq \frac{2}{3}.
\end{aligned}$$

This proves the existence of a tester with the required properties. □

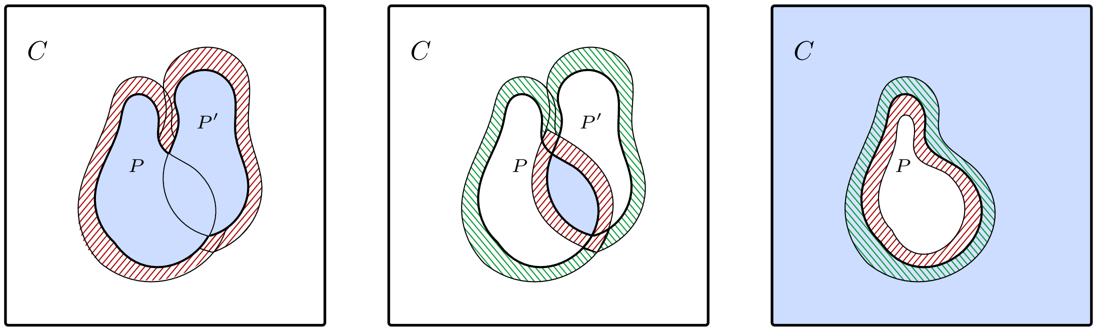


Figure 3.1: Union, intersection and complement of properties.

Corollary 3.5.2. Let C be a class of objects and $P, P' \subseteq C$ properties on C. If P and P' are testable on C, then $P \cup P'$ is a testable property on C.

Proof. Let $\epsilon \in [0, 1]$ and $n \in \mathbb{N}$. Since P and P' are testable, there exists an ϵ-tester for $P|n$ on $C|n$ and an ϵ-tester for $P'|n$ on $C|n$ with constant query complexity. By Lemma 3.5.1 this guarantees the existence of a constant time ϵ-tester for $(P \cup P')|n$ and hence $P \cup P'$ is testable. $\square$

Corollary 3.5.2 also holds in the case of uniform testability, but is not used in this thesis.

Remark 3.5.3. Figure 3.1 shows why for the boolean operations intersection and complement we can not show closure for property testing using a similar argument as in the proof of Lemma 3.5.1. In the case of the intersection of two properties $P, P' \subseteq C$ an object $\mathcal{A}$ can be both ϵ-close to P and P' while being ϵ-far from $P \cap P'$. The decision of both the tester for P and P' is therefore uncertain and hence a tester for $P \cap P'$ can not use the decisions made by the testers for P and P'. In the case of the complement of a property $P \subseteq C$ the property $C \setminus P$ contains objects that are not in P but ϵ-close to P and hence rejection of the tester for P can not be used in deciding property $C \setminus P$. In fact in general property testing is neither closed under intersection nor complement as we will argue in the following.

Lemma 3.5.4. *Property testing is not closed under taking complement.*

Proof. We consider the bounded degree model for graphs. Let $d \in \mathbb{N}$ and $\mathcal{P} \subseteq \mathcal{C}_d$ be the property of graphs of bounded degree d, that are not bipartite. Then for every $\epsilon \in (0, 1)$ the property P can be tested using the tester which accepts every input graph of large enough size and calculating an exact answer for small graphs. This tester works because every graph (of large enough size) is ϵ-close to being not bipartite, as we only need to ensure one triangle by removing at most 3 edges to ensure that the graph has 3 vertices of degree $< d$ and then adding 3 edges to form a triangle on these 3 vertices. But the complement of P, i. e. the property of being a bipartite graph, was shown not to be testable in [73]. $\square$

Lemma 3.5.5. *Property testing is not closed under intersection.*

Proof. We consider the bounded degree model for graphs. Let $d \in \mathbb{N}$, $d \geq 2$ and $\mathcal{P} \subseteq \mathcal{C}_d$ be the property of bipartite graphs. Let further $\mathcal{P}_\triangle \subseteq \mathcal{C}_d$ be the property of graphs that contain a triangle and $\mathcal{P}_{\neg\triangle} \subseteq \mathcal{C}_d$ be the property of all triangle-free graphs. Consider the two properties $\mathcal{P} \cup \mathcal{P}_\triangle$ and $\mathcal{P} \cup \mathcal{P}_{\neg\triangle}$. $\mathcal{P} \cup \mathcal{P}_\triangle$ is testable as every large graph is ϵ-close to $\mathcal{P}_\triangle$ and hence we can accept all graphs larger than some constant and compute the answer precisely for small graphs. Furthermore, $\mathcal{P} \cup \mathcal{P}_{\neg\triangle} = \mathcal{P}_{\neg\triangle}$ and we know from Section 3.3 that $\mathcal{P}_{\neg\triangle}$ is testable. But since $(\mathcal{P} \cup \mathcal{P}_\triangle) \cap (\mathcal{P} \cup \mathcal{P}_{\neg\triangle}) = \mathcal{P}$ and $\mathcal{P}$ is not testable (see [73]) property testing is not closed under intersection. □

3.6 Local reductions

In this section we will introduce local reductions. Assume we want to reduce testing a property $P \subseteq C$ in some property testing model to testing property $P' \subseteq C'$ in some other model and we have a tester T' for P'. This means that, given query access to some input object $\mathcal{A}$, we need to be able to decide whether to accept $\mathcal{A}$ using tester T' as a black box. We can do this if for any object $\mathcal{A} \in C$ there is an object $\mathcal{B} \in C'$ such that the probability with which T' accepts $\mathcal{B}$ coincides with an appropriate acceptance probability for $\mathcal{A}$ and we can simulate query access to $\mathcal{B}$ using query access to $\mathcal{A}$. We will formalise this in the following.

Let $\big(C, \cong, \Sigma, \mathsf{code}, \mathrm{dist}, \mathrm{QA}\big)$ and $\big(C', \cong', \Sigma', \mathsf{code}', \mathrm{dist}', \mathrm{QA}'\big)$ be two property testing models with query access models $\mathrm{QA} = \Big(\ell, (\mathrm{aux}_\mathcal{A})_{\mathcal{A} \in C}, (Q_\mathcal{A})_{\mathcal{A} \in C}, (A_\mathcal{A})_{\mathcal{A} \in C}, (\mathrm{ans}_\mathcal{A})_{\mathcal{A} \in C}\Big)$ and $\mathrm{QA}' = \Big(\ell', (\mathrm{aux}'_\mathcal{A})_{\mathcal{A} \in C'}, (Q'_\mathcal{A})_{\mathcal{A} \in C'}, (A'_\mathcal{A})_{\mathcal{A} \in C'}, (\mathrm{ans}'_\mathcal{A})_{\mathcal{A} \in C'}\Big)$.

Definition 3.6.1. Let $P \subseteq C$ and $P' \subseteq C'$ be two properties. A *local reduction* from P to P' is a function $f : C \to C'$, for which there exists constants $k, t \in \mathbb{N}$ (independent of n), a computable function $g : \{\mathrm{aux}_\mathcal{A} \mid \mathcal{A} \in C\} \to \{\mathrm{aux}'_\mathcal{A} \mid \mathcal{A} \in C'\}$ and a function $h : (0,1) \to (0,1)$ such that for every $\mathcal{A} \in C$ the following properties hold, where $\mathcal{B} := f(\mathcal{A})$.

(LR1) If $|A| = n$ then $|B| = kn$.

(LR2) $g(\mathrm{aux}_\mathcal{A}) = \mathrm{aux}'_\mathcal{B}$.

(LR3) For every query $q \in Q'_\mathcal{B}$ we can adaptively[1] compute t queries $q_1, \ldots, q_t \in Q_\mathcal{A}$ such that the answer to the query q can be computed from the answers to the t queries $q_1, \ldots, q_t$. Formally this means that there are computable functions $S_i : Q'_\mathcal{B} \times (Q_\mathcal{A} \times A_\mathcal{A})^i \to Q_\mathcal{A}$, $i \in [t]$ and $T : Q'_\mathcal{B} \times (Q_\mathcal{A} \times A_\mathcal{A})^t \to A'_\mathcal{B}$ such that for every query $q \in Q'_\mathcal{B}$ the following holds.

$$\mathrm{ans}'_\mathcal{B}(q) = T(q, (q_1, a_1), \ldots, (q_t, a_t)),$$

[1]By adaptively computing queries we mean that the selection of the next query may depend on the answer to the previous query.

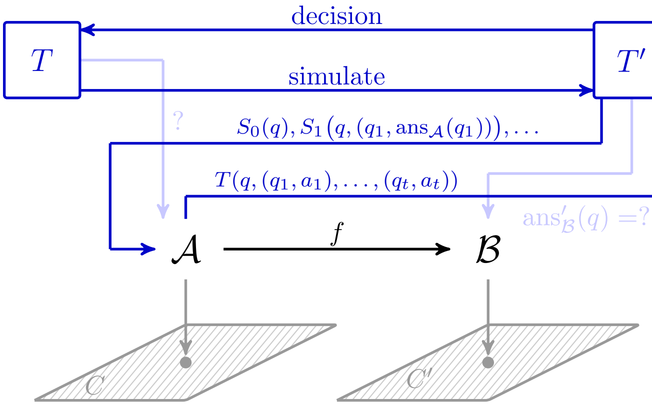


Figure 3.2: Simulating a tester using a local reduction.

where

$$\begin{aligned} q_1 &:= S_0(q), \\ q_i &:= S_{i-1}\big(q, (q_1, a_1), \ldots, (q_{i-1}, a_{i-1})\big) \text{ and} \\ a_j &:= \mathrm{ans}_{\mathcal{A}}\left(q_j\right) \end{aligned}$$

for $i \in \{2, \ldots, t\}$, $j \in \{1, \ldots, t\}$.

(LR4) If $\mathcal{A} \in P$ then $\mathcal{B} \in P'$.

(LR5) If $\mathcal{A}$ is ϵ-far from P then $\mathcal{B}$ is $h(\epsilon)$-far from P'.

Lemma 3.6.2. *Let $P \subseteq C$, $P' \subseteq C'$ be two properties. If there is a local reduction from P to P' then there is a function $h : (0,1) \to (0,1)$ and constants $k, t \in \mathbb{N}$ such that if for some $\epsilon \in (0,1)$ and $n \in \mathbb{N}$ there is a $h(\epsilon)$-tester for $P'|kn$ on $C|kn$ with query complexity f, then there is an ϵ-tester for $P|n$ on $C|n$ with query complexity $t \cdot f(n)$.*

Proof. Let $\tilde{f}$ be a local reduction from P to P' with k, t, g, h as in Definition 3.6.1. Let $\epsilon \in (0,1)$, $n \in \mathbb{N}$ and let T' be a $h(\epsilon)$-tester for $P'|kn$ on $C'|kn$ with query complexity f. We will construct an ϵ-tester T for $P|n$ on $C|n$ as follows. Given $\mathrm{aux}_{\mathcal{A}}$ as input and oracle access to an object $\mathcal{A} \in C|n$ we will first compute $g(\mathrm{aux}_{\mathcal{A}})$. (LR2) implies that $g(\mathrm{aux}_{\mathcal{A}}) = \mathrm{aux}'_{\tilde{f}(\mathcal{A})}$. Since $\tilde{f}(\mathcal{A}) \in C|kn$ by property (LR1), we can simulate T' on $\tilde{f}(\mathcal{A})$ with input $\mathrm{aux}'_{\tilde{f}(\mathcal{A})}$ in the following way. Whenever T' makes a query $q \in Q'_{\tilde{f}(\mathcal{A})}$ we make t queries $q_1 := S_0(q)$, $q_2 := S_1\big(q, (q_1, \mathrm{ans}_{\mathcal{A}}(q_1))\big)$, ..., $q_t := S_{t-1}\big(q, (q_1, \mathrm{ans}_{\mathcal{A}}(q_1)), \ldots, (q_{t-1}, \mathrm{ans}_{\mathcal{A}}(q_{t-1}))\big)$ and answer q with $T(q, (q_1, \mathrm{ans}_{\mathcal{A}}(q_1)), \ldots, (q_t, \mathrm{ans}_{\mathcal{A}}(q_t)))$ which is equal to $\mathrm{ans}'_{\tilde{f}(\mathcal{A})}(q)$ by (LR3). We then accept $\mathcal{A}$ if T' accepts $\tilde{f}(\mathcal{A})$.

Since T' is a property tester for $P'|kn$ the properties (LR4) and (LR5) of local reductions guarantee that T is an ϵ-tester for $P|n$. Furthermore since every query of T' is simulated by t queries T has query complexity $t \cdot f$. $\square$

Lemma 3.6.3. *Let $P \subseteq C$ and $P' \subseteq C'$ be two properties. If there exist a local reduction from P to P' and P is not non-uniformly testable with $o(f(n))$ queries for some function $f : \mathbb{N} \to \mathbb{N}$, then P' is not non-uniformly testable with $o(f(n))$ queries.*

Proof. Let $\tilde{f}$ be a local reduction from P to P' with t, k, g, h as in Definition 3.6.1. Towards a contradiction assume that for every $\epsilon \in (0, 1)$ and every $n \in \mathbb{N}$ there is an ϵ-tester for $P'|n$ on $C'|n$ with query complexity $f'(n) \in o(f(n))$ for some function $f' : \mathbb{N} \to \mathbb{N}$. Hence for every $\epsilon \in (0, 1)$ and every $n \in \mathbb{N}$ there is a $h(\epsilon)$-tester for $P'|kn$ on $C'|kn$ with query complexity $f'(kn)$. Then for every $\epsilon \in (0, 1)$ and every $n \in \mathbb{N}$ there is an ϵ-tester for $P|n$ on $C|n$ with query complexity $t \cdot f'(kn) \in o(f(n))$ by Lemma 3.6.2. This yields a contradiction to P not being non-uniformly testable with $o(f(n))$ queries. □

Example 3.6.4. Let $\sigma = \{R_1, \dots, R_\ell\}$ be a relational signature, $d \in \mathbb{N}$, $d' := d(\mathrm{ar}(\sigma) - 1)$ and $P \subseteq C_d$ be a property such that $\mathcal{G}(\mathcal{A}) \cong \mathcal{G}(\mathcal{B})$ implies that either $\mathcal{A}, \mathcal{B} \in P$ or $\mathcal{A}, \mathcal{B} \notin P$ for any two structures $\mathcal{A}, \mathcal{B} \in C_d$. We will show that for such properties "taking the Gaifman graph" is a local reduction from P in the model of σ-structure of bounded degree d to the property

$$\mathcal{P} := \{G \in \mathcal{C}_d \mid \text{there is } \mathcal{A} \in P \text{ such that } \mathcal{G}(\mathcal{A}) = G\}$$

in the model of bounded degree d' graphs. For this let us define $g(n) = n$, $h(\epsilon) = \frac{\epsilon}{2d\,\mathrm{ar}(\sigma)}$, $k = 1$ and $f(\mathcal{A}) = \mathcal{G}(\mathcal{A})$ for every $n \in \mathbb{N}$, $\epsilon \in (0, 1)$, $\mathcal{A} \in C_d$. Then for constant $t = d \cdot \mathrm{ar}(\sigma)$ and any $\mathcal{A} \in C_d$ on n elements properties (LR1) to (LR5) hold as explained in the following.

(LR1) Is true as the number of vertices of $\mathcal{G}(\mathcal{A})$ is equal to the number of elements of $\mathcal{A}$.

(LR2) Is true as the number of vertices of $\mathcal{G}(\mathcal{A})$ is equal to the number of elements of $\mathcal{A}$.

(LR3) For $i \in [t]$ let

$$S_i\Big((j, k), \big((q_1, a_1), \dots, (q_i, a_i)\big)\Big) = \Big(j, i \bmod \mathrm{ar}(\sigma)\Big)$$

where

$$\begin{aligned} q_1 &:= S_0(q), \\ q_i &:= S_{i-1}\big(q, (q_1, a_1), \dots, (q_{i-1}, a_{i-1})\big) \text{ and} \\ a_j &:= \mathrm{ans}_{\mathcal{A}}\left(q_j\right) \end{aligned}$$

Then $T\Big((j, k), \big((q_1, a_1), \dots, (q_t, a_t)\big)\Big)$ gives the k-th unique element to appear in the tuples $a_1, \dots, a_t$. Obviously $\mathrm{ans}_{\mathcal{G}(\mathcal{A})}((j, k)) = T\Big((j, k), \big((q_1, a_1), \dots, (q_t, a_t)\big)\Big)$ for some adjacency list representation of $\mathcal{G}(\mathcal{A})$.

(LR4) If $\mathcal{A} \in P$ then $f(\mathcal{A}) \in \mathcal{P}$ by definition of $\mathcal{P}$.

(LR5) If $\mathcal{A}$ is ϵ-far from P then we can show that $\mathcal{G}(\mathcal{A})$ is $\frac{\epsilon}{2d\,\mathrm{ar}(\sigma))}$-far from $\mathcal{P}$. Assume towards a contradiction that this is not the case. Then there is a graph $G \in \mathcal{P}$ with $\mathrm{dist}(\mathcal{G}(\mathcal{A}), G) < \frac{\epsilon}{2\,\mathrm{ar}(\sigma)} \cdot d' \cdot n = \frac{\epsilon}{2} \cdot n$. Then we can construct a structure $\mathcal{B}$ such that $\mathcal{G}(\mathcal{B}) = G$ by modifying tuples in $\mathcal{A}$ as follows. For every edge $\{u, v\}$ which has to be added to $\mathcal{G}(\mathcal{A})$ to get G we add tuple $(u, v, \dots, v) \in A^{\mathrm{ar}(\sigma)}$ to $R^{\mathcal{A}}$ for some relation R with $\mathrm{ar}(R) = \mathrm{ar}(\sigma)$. Note that if $\mathrm{ar}(\sigma) < 2$ then all graphs in $\mathcal{P}$ are edgeless and hence we do not have to add edges to $\mathcal{G}(\mathcal{A})$. For every edge $\{u, v\}$ which has to be deleted from $\mathcal{G}(\mathcal{A})$ to get G we replace u by v in all tuples containing both u and v. For this maximally d tuples have to be added to $\mathcal{A}$ and maximally d tuples have to be deleted from $\mathcal{A}$. Therefore $\mathcal{B}$ is $(2d \times \frac{\epsilon}{2} \cdot n)$-close to $\mathcal{A}$. But since $G \in \mathcal{P}$ there is a structure $\mathcal{B}' \in C_d$ with $\mathcal{G}(\mathcal{B}') = G$. But then $\mathcal{B} \in P$ since $\mathcal{G}(\mathcal{B}') = G = \mathcal{G}(\mathcal{B})$ which contradicts the assumption that $\mathcal{A}$ was ϵ-far from P.

Using the local reduction from our example we can show the following theorem which was also shown by Adler and Harwath [2, Theorem 5].

Theorem 3.6.5. *Let $\sigma = \{R_1, \dots, R_\ell\}$ be a relational signature and $d \in \mathbb{N}$ Let $P \subseteq C_d$ be a property. Assume that for any two structures $\mathcal{A}, \mathcal{B} \in C_d$ the relation $\mathcal{G}(\mathcal{A}) \cong \mathcal{G}(\mathcal{B})$ implies that either $\mathcal{A}, \mathcal{B} \in P$ or $\mathcal{A}, \mathcal{B} \notin P$. Then if the property $\mathcal{P} := \{\mathcal{G}(\mathcal{A}) \mid \mathcal{A} \in P\}$ is non-uniformly testable on the class of bounded degree $d' := d(\mathrm{ar}(\sigma) - 1)$ graphs, then P is non-uniformly testable.*

Proof. If $\mathcal{P}$ is non-uniformly testable on the class of graphs of bounded degree d' then for every $\epsilon' \in (0, 1)$ and every $n \in \mathbb{N}$ there is an ϵ'-tester for $\mathcal{P}|n$ on $\mathcal{C}_d|n$ with constant query complexity f. Using the reduction from Example 3.6.4 and Lemma 3.6.2 we get that for every $\epsilon \in (0, 1)$ and every $n \in \mathbb{N}$ there is an ϵ-tester for $P|n$ on $C_d|n$ with constant query complexity $d \cdot \ell \cdot f \in \mathcal{O}(1)$. $\square$

Chapter 4

Related work

In this Section we will briefly survey related research area. As this thesis combines different areas of research, there is a variety of related work. Here our main focus is on algorithmic meta-theorems, property testing, the intersection of the two and lower-bound techniques for showing non-testability.

4.1 Algorithmic meta-theorems

Algorithmic meta-theorems aim for an algorithm for a class of problems and thus solving a wide range of problems simultaneously. The development of algorithmic meta-theorems requires to generalise and unify certain algorithmic techniques, while their negation yields insights into the limitations of certain approaches.

Logic provides us with a tool for the systematic study of algorithmic complexity. In the following we consider algorithmic meta-theorems which are formulated for classes of problems which are definable in some logic. The most well known such algorithmic meta-theorem is Courcelle's theorem which states that any property definable in MSO_2 can be decided in linear time on the class of graphs of bounded treewidth [29]. Note that classes of graphs of bounded treewidth are sparse (for a definition of treewidth see Section 9.3). Further note that MSO is the extension of FO allowing quantification over sets of vertices, i. e. MSO is recursively defined in the same way as FO where we extend FO by a collection of set variables, atomic formulas expressing containment of a vertex in a set and also allow quantification over set variables. MSO_2 is a further extension of MSO allowing quantification over sets of edges. Deciding some properties defined by an MSO-sentence or MSO_2-sentence is computationally hard as many NP-complete problems can be defined in MSO/MSO_2 (e. g. Hamiltonicity can be expressed in MSO_2 but not in MSO and 3-colourability is expressible in MSO).

More precisely, Courcelle's theorem considers a more general problem then deciding a property defined by a MSO_2-sentence, i. e. the sentence is considered to be part of the input. For a logic $\mathcal{L}$ and a class $\mathcal{C}$ of graphs, the model checking problem of $\mathcal{L}$ on $\mathcal{C}$ is defined as the following problem.

Model checking problem of $\mathcal{L}$ on $\mathcal{C}$
Input: A sentence $\varphi \in \mathcal{L}$ and a graph $G \in \mathcal{C}$. **Aim:** Decide whether $G \models \varphi$.

While deciding a property defined in FO can be computed in polynomial time [3,88], FO model checking is PSPACE-complete in general [125] (PSPACE-hardness follows from the PSPACE-completeness of QBF satisfiability as QBF satisfiability is equivalent to FO model checking with a suitable fixed input graph). Furthermore, the FO model checking problem is AW[$*$]-complete and thus not likely to be fixed parameter tractable on general graph classes parametrised by $\|\varphi\|$ [45], i. e. there is no algorithm for the model checking problem of FO with running time $f(\|\varphi\|)|V(G)|^{\mathcal{O}(1)}$ for some function $f : \mathbb{N} \to \mathbb{N}$. Nonetheless, FO model checking is tractable on certain graph classes. Tractability of FO model checking was gradually extended to more general sparse graph classes by utilising locality of FO. The starting point was Seese's Theorem which shows that FO model checking on classes of graphs of bounded degree is fixed-parameter tractable in linear time [119]. The proof of Seese's theorem uses Hanf's theorem and the boundedness of the number of r-types of bounded degree. This was extended by Frick and Grohe to provide a linear time (fixed parameter) algorithm for FO model checking on planar graph classes and classes that are apex-minor free as well as an $\mathcal{O}(n^{1+\epsilon})$ algorithm for classes of locally bounded treewidth [62]. Flum and Grohe proved that for classes excluding a minor [60] FO is tractable, which was further extended by Dawar et al. to classes locally excluding a minor [38]. Dvořák et al. extended this by proving that there is a linear algorithm for FO model checking on classes of bounded expansion and a $\mathcal{O}(n^{1+\epsilon})$ algorithms for classes of locally bounded expansion [50]. The most general sparse graph classes known to be tractable for FO model checking are nowhere dense graph classes due to a result by Grohe et al. [79]. The notion of nowhere denseness is a notion of sparsity and was introduced by Nešetřil and Ossona de Mendez [110, 111]. Intuitively, proving tractable FO model checking uses Gaifman's locality theorem by checking satisfiability locally on r-neighbourhoods and solving an independent set problem.

The result from [79] proving tractability of FO model checking on nowhere dense graph classes is optimal in the sense that FO model checking on somewhere dense classes (classes that are not nowhere dense), which are closed under taking subgraphs, is intractable [41]. However, the condition of being closed under taking subgraphs is necessary as there are classes of graphs which are not sparse but allow tractable FO model checking e. g. classes of bounded cliquewidth [30] and for certain subclasses classes of interval graphs [64].

Meta-theorems have also been developed for optimisation problems. Dawar et al. proved that FO definable optimisation problem admits polynomial time approximation scheme on classes forbidding at least one minor [39]. More recently, Dvořák has worked on approximation problems definable in FO on restricted classes of graphs [48, 49].

Logic plays an important role in the context of database queries. We can model relational databases by relational structures, where each table of the database corresponds to a relation in the structure and each row of a table corresponds to a tuple in the corresponding relation. In addition we can model queries to a relational database by formulas (with free variables) of certain logics. Consider a relational structure $\mathcal{A}$ representing a relational database and any formula $\varphi(\overline{x})$, then any tuple of elements $\overline{a}$ for which $\mathcal{A} \models \varphi(\overline{a})$ is considered an answer to the query represented by φ. A large amount of research has been undertaken considering query evaluation and enumeration in different settings. Here query evaluation relates to the problem of computing the set of all tuples that are answers to a given query and query enumeration refers to the problem of enumerating all answers to a query allowing some delay time between steps, after a preprocessing phase. Evaluating FO queries is well known to be PSPACE-complete [120]. However, on relational structures of bounded treewidth MSO queries can be evaluated in polynomial time [31], which was improved to linear time in the size of the structure and the size of the output [59]. The enumeration problems for MSO can be done with delay proportional to the size of the next output and linear preprocessing on structures of bounded treewidth [15, 96]. FO query enumeration has been shown to be possible with constant delay and linear preprocessing time for bounded degree relational structures [46, 95], for structures of low degree [47] and for structures of bounded expansion [96]. For nowhere dense structures enumerating FO queries can be done with pseudo linear preprocessing [118].

A recent study by Fomin et al. considers the parametrised complexity of edge/vertex edit distance to satisfying an FO-sentence [61]. It is shown that in general the problem is not tractable, however, for some fragments of FO the problem is fixed parameter tractable parametrised by the number of edits.

4.2 Property testing

In computer science we are often faced with the challenge that we need to solve a problem efficiently which in general does not allow an efficient solution. Randomised algorithms have provided some inroads towards providing a solution to this challenge. There are two different varieties of randomised algorithms, exact randomised algorithms with good running time in the average case (Las Vegas algorithm) or randomised approximation algorithm (Monte Carlo algorithm). Property testers are algorithms which use randomness to approximate the solution to a decision problem guaranteeing correctness with high probability. Property testers might be used in practice in cases where running an exact algorithm is infeasible due to the size of the input considered, when an approximate decision with some accuracy guarantees is sufficient or as a preprocessing heuristic preceding a computationally expensive exact algorithm. Even though property testing research is of a theoretical nature, there are efforts being made towards applicability, i. e. developing new testers with good running times as well as improving the dependency of the query complexity on the proximity parameter ϵ.

Property testing was first introduced by Rubinfeld and Sudan in 1996 [117]. They introduce property testing for testing properties of functions over finite domains. Motivated by developing program checkers Rubinfeld and Sudan formalized a notion of closeness, where a function f is considered to be ϵ-close to having a property P, i. e. a class of functions over a finite domain, if there is a function in P that differs from f only on an ϵ-fraction of all possible inputs. Note that in this setting the distance between functions is therefore the number of inputs on which they differ and a tester is allowed to query the function f by evaluating it for a specific input. The emphasis of the paper is on finding local characterisations of low degree univariate and multivariate polynomials that allow property testers.

Since its introduction 25 years ago property testing has received a lot of research interest. We will survey work considering graph property testing in the following sections. However, there are several other settings in which property testing is considered, including testing algebraic properties of functions (see e. g. [16, 26, 113]) and distribution testing (see the survey [25]).

4.2.1 The dense model

The notion of property testing as introduced in [117] was extended to testing graph properties by Goldreich et al. in 1996 in a preliminary version of [71]. In [71] graphs are considered to be boolean functions on pairs of vertices. This corresponds naturally to graphs being encoded in an adjacency matrix. Therefore the size of a graph on $n \in \mathbb{N}$ vertices is considered to be n^2. A property tester is allowed to query this adjacency function, i. e. for a graph G permissible query are tuples $(u, v) \in V(G) \times V(G)$. The oracle answers the query (u, v) with 1 if there is an edge between u and v and 0 otherwise. In order to determine the set of permissible queries, the property tester gets the number $n := |V(G)|$ of vertices as auxiliary information. Distance between graphs is defined in the same way as in the bounded degree model, i. e. as the minimum number of edge modifications necessary to make two graphs isomorphic. However due to the different encoding size, a graph is ϵ-close to a property $\mathcal{P}$, if it can be made into a graph in $\mathcal{P}$ with at most ϵn^2 edges modifications. Note that this notion of closeness can not distinguish "sparse" graphs from each other. This is because for a classes of graphs with average degree bound by some function $f \in o(n)$ and for large enough $n \in \mathbb{N}$ all graphs in this class have less then ϵn^2 edges, implying that all large sparse graphs are close to each other. Therefore the model is unsuitable for sparse graph classes. Furthermore, "sparse" properties like planarity are trivial to test in this model. The model of property testing on graphs as introduced in [71] is therefore often referred to as the dense model or the adjacency matrix model.

Property testing on dense graphs is well understood and closely connected to Szemerédi's Regularity Lemma [121]. This connection was finally established and formalised by Alon et al. who characterised testable properties in the dense model [7] in 2006. This characterisation result was proceeded by a characterisation of properties testable with a one-sided testers [12] and a characterisation of for which graphs H, H induced subgraph freeness can be tested with very small query complexity i. e. polynomial in $\frac{1}{\epsilon}$ [11].

Among the testable properties in the dense model are for example bipartiteness, k-colourable (for $k > 1$) [71], biclique min-bisection, max-clique [69], subgraph freeness (see [4] and citations therein) and induced subgraph freeness (see [11] and citations therein). Whereas, on the other hand there are properties that are not testable with $\Theta(f(n))$ queries for any function $f \in \mathcal{O}(n^2)$ in the dense model [72]. A natural property that can not be tested with $o(n)$ queries is graph isomorphism [58].

4.2.2 The bounded degree model

The bounded degree model or adjacency list model was introduced by Goldreich and Ron in 1997 [73]. Since their seminal work the bounded degree model received a great amount of attention. There are several specific properties for which testers have been developed. Testable properties in the bounded degree model include connectivity, k-edge-connectivity (for $k > 1$), cycle-freeness, Eulerianity [73], degree regularity [69], outerplanarity [131] and testing K_t-subdivision freeness (for $k > 1$) [94]. Furthermore, there are properties which allow for sublinear (but not constant) query testing, including bipartiteness [74] (lower bound [73]), expansion [36, 75, 91, 109] (lower bound [73]) and testing cluster structure (extends expansion) [33].

While there are several different testability results for very specific graph problems, more general result for property testing were obtained by either restricting the class of bounded degree graphs further, or considering a particular class of properties or restricting testers (to e. g. one-sided error testers or POT's). All of these testing results utilise random walks.

In 2009 Czumaj et al. proved that hereditary properties are testable on certain restricted classes of bounded degree graphs namely hereditary, non-expanding graph classes [35]. Here a class of graphs is non-expanding if the class contains only weak expanders (i. e. $h(G) \in \mathcal{O}(1/\log^2(n))$ for every graph G in the class) and a hereditary class of graphs is a class that is closed under vertex deletion. Non-expanding hereditary graph classes include the class of planar graphs, classes of bounded genus and graphs with forbidden minors. However, hereditary properties are not testable with constant query complexity in general as bipartiteness is a hereditary property and is shown in [73] to take at least $\frac{1}{4} \cdot \sqrt{n}$ queries to test, where n is the size of the input graph.

Partly using ideas from [35], Benjamini et al. showed that every minor-closed graph property is testable on bounded degree graphs with constant query complexity [18]. A minor closed property is a set of graphs, which is closed under edge and vertex deletion as well as edge contraction. The query complexity of testing minor-closed properties was, since the seminal work [18], gradually improved from triple exponential in $\frac{1}{\epsilon}$ to polynomial in $\frac{1}{\epsilon}$ employing new techniques [97, 99]. This work relies on minor closed properties being hyperfinite. The testers operate using a partition oracle, which given a vertex v returns the part of the hyperfinite partition of the graph which contains v. Such oracles use a constant number of (neighbour) queries to access a part of the hyperfinite partition. Furthermore, all answers of such a partitioning oracles are consistent with one hyperfinite partition of the graph. Since the seminal work of Hassidim et al. [84], partition oracles have been a very useful tool in property testing and have been used and improved gradually (see e. g. [97–99]).

Finally, this work culminated in testability results for hyperfinite classes and properties. Newman and Sohler proved that properties of hyperfinite graphs (which includes minor-closed properties and planarity) are testable in the bounded degree model and that every property is testable on any class of hyperfinite graphs using similar methods to the work preceding it [112].

Property testing has also been considered for directed graphs in the bounded degree setting. There are two natural extensions to the bounded degree model for undirected graphs introduced by Bender and Ron in [17]. In the unidirectional model only the outdegree of graphs is bounded and testers can only query outgoing edges. In the bidirectional model both indegree and outdegree are bounded and testers can query both incoming and outgoing edges. Besides the introductory paper considering acyclicity and connectivity in both models further work includes some specific results mainly in the bidirectional model [86, 130]. Furthermore, Czumaj et al. considered the relationship between the two models and showed that every property testable in the bidirectional model can be tested with sublinear query complexity in the unidirectional model [34]. Furthermore, Connectivity yields an example of a property which is testable in the bidirectional model but requires $\Omega(\sqrt{n})$ queries in the unidirectional model [17].

4.2.3 Characterisation results in the bounded degree model

While there is no full characterisation of which properties are testable in the bounded degree model, there are several partial results making progress towards a characterisation. We will introduce such results in this subsection. The important technical tool used to prove characterisation result is to show that testers operate in a canonical way. Such canonical tester results for the bounded degree model were shown in [34, 78] and used (adapted to suit the individual settings) in each of the following results.

In 2009 Goldreich and Ron classified properties admitting a one-sided error constant query POT both in the dense and the bounded degree model [76]. We discuss the bounded degree model characterisation in more detail in Section 8. Properties having a one-sided error POT are precisely generalised subgraph freeness properties which are non-propagating (see Section 8.1 for definitions). While this gives a characterisation for testability of any property testable with a one sided-error POT, the characterisation depends on the non-propagation condition which is similar to saying that a slightly restricted modification problem as mentioned in the Introduction (Section 1) is solvable. The authors leave the question when this modification problem is solvable open. Hence the strength of this characterisation might lie mostly in the distinction of which testable property admit a one-sided error POT. The notion of one-sided error POT's can be extended to two-sided error POT's, which were studied by Goldreich and Shinkar [77]. While they exhibit several natural properties in different models that have a two-sided error POT but no one-sided error POT, a characterisation for two-sided error POT's is not known yet.

Ito et al. gave a characterisation of which monotone properties and which hereditary properties are testable by a one-sided error constant query property tester in bounded degree directed graphs [89]. Here both the unidirectional model and the bidirectional model are considered. Note that a property is called monotone if it is closed under vertex and edge deletion. Their characterisation states that a monotone property is one-sided error testable if and only if it is close to a property that is defined by a set of forbidden subgraphs of constant size. A hereditary property is one-sided error testable if and only if it is close to a property that is defined by a set of forbidden induced subgraphs of constant size. We would like to emphasise here that the characterisation is entirely independent of any removal problem being solvable.

Adler and Harwath give combinatorial classifications of testable properties in the bounded degree model. To obtain this result they introduce a notion of locality of a property P of bounded degree relational structures roughly stating that if the relative frequency vector of a structure $\mathcal{A}$ is "similar" to the relative frequency vector of a structure $\mathcal{A}' \in P$ then $\mathcal{A}$ has to be ϵ-close to P. In other words, locality means that the modification problem mentioned in the Introduction (Section 1) is solvable. According to their classification a property is testable if and only if it is local (see Section 5 for more details). However, a classification for which properties the modification problem is solvable is unknown. Furthermore, to consider the question of a characterisation of testable properties in the bounded degree model settled, we would like a statement which does not require the modification problem to be solvable.

Fichtenberger et al. prove that every (infinite) property of bounded degree graphs contains an (infinite) hyperfinite subproperty (were a subproperty is simply a subset which is closed under isomorphism) [56]. The result is obtained by repeatedly making use of a result from Alon [102, Proposition 19.10] which roughly states that for every graph G there exists a constant size graph which realises approximately the same neighbourhood distribution as the original large graph G (see Section 5.2 for more details). However, it is not true that every property which contains a hyperfinite subproperty is testable as e. g. bipartiteness is not testable [73]. While this result is highly non-trivial and gives us insights into which properties are testable, a characterisation requires additional new ideas.

4.2.4 Lower-bound techniques in the bounded degree model

There are three lower bound techniques known for property testing. To obtain lower bounds on the query complexity of property testers, one can employ Yao's principle [126]. It states that the query complexity of a randomised algorithm can be lower bounded by the query complexity of a deterministic algorithm solving the problem correctly on average for any distribution of instances. This allows us, to prove a lower bound by showing that there is a distribution of instances for which any deterministic algorithm fails. This lower bound technique has been used for example by Goldreich and Ron to prove a $\Omega(\sqrt{n})$ lower bound for bipartiteness and expansion [73], by Yoshida to provide a $\Omega(n)$ lower bound for testing CSP's [127, 128] and by Bogdanov et al. to prove a $\Omega(n)$ lower bound for 3-SAT [22].

We can reduce testing a property with a known lower bound to testing an other property yielding a lower bound for the other property. In Section 3.6.1 we introduce the notion of local reductions which reduce property testing problems to one another. Lower bounds proved using local reduction can be found in [22, 70, 129] which we will discuss further in Section 9.

Another method to prove property testing lower bounds was developed using reductions from known lower bound for communication complexity [21]. Roughly, communication complexity refers to the number of bits exchanged in a two party protocol with shared randomness, which aims to decide whether the two parties private inputs have a certain property. While in [21] this method was used to prove lower bounds for property testing of functions, recently the method was extended to graphs by Eden and Rosenbaum [53]. Eden and Rosenbaum demonstrated the methods effectiveness by providing simpler proofs of known lower bounds in graph property testing for problems including estimating the number of edges in a graph and estimating the number of triangles in a graph.

Methods for proving non-testability, i. e. proving that a property can not be tested with constant query complexity, also arise from the partial characterisation results from the previous subsection. For example, an easy corollary of the result from Fichtenberg et al. [56] is that every infinite property of expanders can not be tested in the bounded degree model as it can not contain a hyperfinite subproperty.

4.2.5 The general model

While most of the research in graph property testing focuses on the dense and the bounded degree model, there have been some efforts made into developing testers, which work well for all graphs. However this is not straight forward and results are very hard to obtain. The most important question to ask in this context is how graphs should be represented and what types of queries should be allowed.

One could consider extending the bounded degree model to more general sparse classes. Parnas and Ron introduced a model in which a graph is represented as an adjacency list and a tester is allowed degree and neighbour queries, which allows the consideration of classes of graphs of unbounded degree [114]. They show testability results for testing the diameter of a graph. However, results in [114] are mainly aimed at sparse graph classes as query complexities grow proportional to the relation of edges to vertices. Furthermore Czumaj and Sohler characterised which properties of planar graphs can be tested in this model [37].

Kaufman et al. considered a model in which degree queries, neighbour queries and adjacency queries are allowed and the size of graph is considered to be the number of edges [93]. This model is suitable for input graphs of all edge densities and hence we call this model the general model. Kaufman et al. develop a tester for bipartiteness in this model with an almost tight lower bound. Furthermore, Alon et al. considered testing H-freeness in the general model especially analysing the query complexity needed dependent on the edge density [8].

Furthermore, Iwama and Yoshida consider parametrised complexity in the general model extended by random edge queries [90]. They show testability for some parametrised graph problems that are trivial on bounded degree graphs as well as dense graphs including k-vertex cover, k-feedback vertex set, k-multicut, k-path-freeness and k-dominating set for $k \geq 1$. This work applies standard parametrized complexity methods (such as branch and bound) for property testing.

4.2.6 Testing properties defined by logical formulas

The study of testing properties defined by logic was first proposed by Alon et al. in 1999 in a preliminary version of [9]. They show that regular languages are testable with constant query complexity (i. e. deciding for a word whether it is in a given regular language) in a property testing model for strings equipped with Hamming distance. Since regular languages can be defined in MSO [24], their result shows testability for a certain class of MSO definable properties. Magniez and Rougemont prove testability of regular (ranked) tree languages in a model considering edit distance with an additional operation called moves [103], extending the methods from [9]. Regular tree languages are also MSO definable [44, 122]. Testability of regular (ranked) tree languages considering edit distance is not known and stated as an open problem in [103].

The study of logically defined properties was further developed by Alon et al. in 1999 [5]. They study testability of FO definable properties in the dense model. Their result states that FO-sentences in Σ_2 define properties that are testable with a constant number of queries, while there is a sentence in Π_2 which defines a non-testable property. The testability result is obtained by showing that satisfying a sentence in Σ_2 is equivalent to some generalised colouring problem. To prove testability of these generalised colouring problems the authors obtain a variation of Szemerédi's Regularity Lemma. The non-testable property defined by a sentence in Π_2 is indistinguishable to some graph isomorphism problem, which is shown to be hard to test in the dense model. The methods used in [5] are not applicable in the bounded degree model.

Recall that Newman and Sohler's result states that on a hyperfinite class of graphs any property is testable [112]. Considering the class $\mathcal{C}_{d,k}^{\mathrm{tw}}$ of graphs of bounded degree d and bounded treewidth k, Newman and Sohler's result implies testability of any property on $\mathcal{C}_{d,k}^{\mathrm{tw}}$. This is due to the class of graphs of treewidth at most k being minor-closed, which implies hyperfiniteness of $\mathcal{C}_{d,k}^{\mathrm{tw}}$. However, the testers obtained in [112] are non-uniform. Towards practical applications of property testing, non-uniformity is problematic. Consider the property $\mathcal{P}$ containing precisely all graphs on n vertices for which n is the Gödel number of a Turing machine halting on the empty word. As the halting problem is undecidable [123], property $\mathcal{P}$ is undecidable. However, $\mathcal{P}$ is non-uniformly testable. Recall that non-uniform testability requires the existence of an ϵ-tester for the problem restricted to graphs with n vertices for every $n \in \mathbb{N}$. Hence non-uniform testability of $\mathcal{P}$ follows as for every n a tester exists, i. e. either the tester accepting every graph or the tester rejecting every graph depending on whether the Turing machine with Gödel number n halts on the empty string.

Taking a step towards practical application, Adler and Harwath generalise Newman and Sohler's result to relational structures [2] and show that for properties $\mathcal{P}$ definable in monadic second order logic with counting (CMSO), there is a uniform tester for $\mathcal{P}$ on $\mathcal{C}_{d,k}^{\mathrm{tw}}$ with constant query complexity and poly-logarithmic running time. Here CMSO extends MSO by first-order modular counting quantifiers $\exists^m$, where $\exists^m \varphi$ is true in a structure if the number of its elements for which φ is satisfied is divisible by m.

The result from [2] was further extended by Adler and Fahey [1]. They introduce a new model using an alternative distance function, which allows element modification in addition to tuple modification. The main result obtained in [1] states that uniform testability of properties definable in CMSO on $\mathcal{C}_{d,k}^{\mathrm{tw}}$ in this new model is possible with constant query complexity and constant running time.

4.2.7 Connection to learning and streaming algorithms

Property testing is closely related to the areas of learning and streaming algorithms. In the PAC-learning (Probably Approximately Correct) model, as introduced in [124], we are given a concept class, i. e. a set of subsets of a universe, and our aim is to identify a target concept by sampling elements of the universe and getting the information, if the elements are in the target concept. We require a PAC-learning algorithm to be able to output with high probability a concept that approximates the target concept well for every target concept in the concept class. Goldreich et al. proved in [117] that PAC-learning with a certain sample complexity implies that the concept class is a testable property with asymptotically similar query complexity. However, the bounded degree model for example does not fit the setting in which we can consider PAC-learning because query access is not a boolean function.

PAC-learning has been considered in connections with logic. In this setting, introduced by

Grohe and Turán [82], a background structure $\mathcal{A}$ and $k \in \mathbb{N}$ are fixed. The universe is A^k and the concept class is the set of all sets of tuples, that can be defined by an FO-formula with k free variables and possibly some other properties (like bounded quantifier rank). Here a set of tuples $X \subseteq A^k$ is definable by a formula $\varphi(\overline{x})$ if $\mathcal{A} \models \varphi(\overline{a})$ if and only if $\overline{a} \in X$. PAC-learning has been proved to be possible in this setting for different types of background structures and different restrictions on formulas (see e. g. for recent work in this area [19,20,81]).

Another related algorithmic framework are streaming algorithms (see e. g. the survey [105]). We consider streaming algorithm which obtains a stream of the edges of a graph and have to solve a problem for the graph represented by the stream using a sublinear (in the size of the graph) amount of space. The algorithm might receive the edges in an arbitrary or a random order and it might be allowed multiple passes over the stream of edges. The challenge for such an algorithm is to decide which edges are important for the algorithmic task at hand. Recently, Monemizadeh et al. proved that testable properties in the bounded degree model can be tested by a streaming algorithm with a single pass of a random order stream of the edges in constant space [106]. This was further extended by Czumaj et al. to the general graph model [32].

Chapter 5

Extending results to relational structures

In this chapter we generalise two existing results for bounded degree undirected graphs to bounded degree relational structures. We would like to note that both generalisations are straight forward but given for the sake of completeness as both results are later used for relational structures.

5.1 Canonical tester and a combinatorial characterisation of testable properties in the bounded degree model

The aim of this section is to verify that a theorem, the so called "canonical tester", can be extended to the bounded degree model of relational structures. The canonical tester was proven in [78] for dense graphs and in [34] for bounded degree graphs and states that every testable property can be tested by a property tester of a canonical form. This is a very useful tool for characterisation results in property testing. In [2] Adler and Harwath give a characterisation of the testable properties in the bounded degree model of relational structures. We apply this result in Chapter 6 to show non-testability of a certain property. To prove this characterisation result Adler and Harwath use the canonical tester for relational structures and hence in this section, for the sake of completeness, we fill the gap and generalise the canonical tester to relational structures.

Let σ be a signature, $d \in \mathbb{N}$ and C_d be the class of σ-structures of bounded degree d. The following notion of repairability of a property was introduced in [2, Definition 7] and was called locality in the original work.

Definition 5.1.1 (Definition 7 in [2])**.** Let $\epsilon \in (0,1]$. A property $P \subseteq C_d$ is ϵ-repairable on C_d if there are numbers $r := r(\epsilon) \in \mathbb{N}$, $\lambda := \lambda(\epsilon) > 0$ and $n_0 := n_0(\epsilon) \in \mathbb{N}$ such that for any σ-structure $\mathcal{A} \in P$ and $\mathcal{B} \in C_d$ both on $n \geq n_0$ vertices, if $\|\text{freq}_r(\mathcal{A}) - \text{freq}_r(\mathcal{B})\|_1 < \lambda$ then $\mathcal{B}$ is ϵ-close to P. The property P is repairable on C_d if it is ϵ-repairable on C_d for every $\epsilon \in (0,1]$.

Note that $\text{freq}_r(\mathcal{A})$ denotes the frequency vector of a structure $\mathcal{A}$ defined in Definition 2.3.6. The following theorem [2, Theorem 19] characterises testability in the bounded degree model using the notion of repairability.

Theorem 5.1.2 (Theorem 19 in [2])**.** *For every property $P \in C_d$, P is non-uniformly testable if and only if P is repairable on C_d.*

The proof of Theorem 5.1.2 requires the generalisation of the canonical tester to relational structures (Corollary 5.1.4), which we will prove in the following.

5.1.1 Proving the existence of canonical testers in the property testing model for bounded degree relational structures

In this section our aim is to show that we can transform every property tester on bounded degree relational structures into a property tester, that first samples elements, explores their neighbourhood and then proceeds deterministically.
Similar results were shown by Goldreich and Trevisian [78], proving the existence of a canonical tester for testable properties in the dense graph model, and by Czumaj, Peng and Sohler [34], proving the existence of a canonical tester for properties on bounded degree graphs. The proof we provide here for the existence of a canonical tester for testable properties on bounded degree relational structures is very similar to the proofs given in [78] and [34], since most arguments can be used in our model in a similar way.

Let $\sigma = \{R_1, \dots, R_\ell\}$ be a relational signature. Without loss of generality, we will assume in this section that the universe of all input structures is $[n]$.

Lemma 5.1.3 (Canonical Tester)**.** *Let $d \in \mathbb{N}$ and $P \subseteq C_d$ be a property that is non-uniformly testable with error probability $\delta \in [0, \frac{1}{2})$ and constant query complexity $c = c(\epsilon, \delta)$. For every ϵ there exists a sequence of classes of structures $F = (F_n)_{n \in \mathbb{N}}$ such that for any $n \in \mathbb{N}$*

- *F_n is a set of σ-structures, each of them being the union of* $\text{ar}(\sigma) \cdot q$ *(not necessarily disjoint) q-balls.*
- *The property $P|n := \{\mathcal{A} \in P \mid |A| = n\}$ can be tested with error probability at most $2 \cdot \delta$ by a tester that operates as follows.*
    - *Uniformly sample* $\text{ar}(\sigma) \cdot c$ *elements,*
    - *explore the c-neighbourhoods of the* $\text{ar}(\sigma) \cdot c$ *sampled elements (deterministically) and*

- ∘ *accept the input structure if and only if the substructure induced by all explored elements is not isomorphic to any $\mathcal{F} \in F_n$.*

We call testers operating in the way the tester in Lemma 5.1.3 does, i. e. first sampling elements, exploring their c-neighbourhoods and then deciding deterministically whether to accept the input, *canonical testers.*

Corollary 5.1.4. Let $d \in \mathbb{N}$, $P \subseteq C_d$ be a property and $\delta \in [0, \frac{1}{2})$. If P is non-uniformly testable with error probability δ then there exists a canonical tester for P with constant query complexity and error probability δ.

Proof. Since P is non-uniformly testable for every $\epsilon \in (0, 1)$ and $n \in \mathbb{N}$ there exist an ϵ-tester T_n with error probability δ for $P|n$ on $C_d|n$ with constant query complexity. By Lemma 3.1.7 we can improve the error probability of T_n to $\frac{\delta}{2}$ by repeating T_n some number of times, which depends only on δ, and deciding on the majority of outcomes. Now the claim clearly follows from Lemma 5.1.3. □

We will prove the existence of a canonical tester (Lemma 5.1.3) by the following Lemmas, each containing one step of transforming an arbitrary tester into a canonical tester, where the steps are the following.

- The first step (Lemma 5.1.6) contains the transformation into a tester, that samples elements, explores their neighbourhood and then queries the structure no further.
- In the second step (Lemma 5.1.7) we make the testers decision isomorphism oblivious. This means that we need to make the decision independent of the identity of the samples, i. e. of the exact location the explored substructure is embedded into the structure, since any isomorphism of structures is basically a renaming of the elements of the structure. Additionally the decision needs to be independent of the order we explore the neighbourhood of the samples in.
- The third step (Lemma 5.1.8) generates a tester, which makes a deterministic decision on the basis of the neighbourhoods of the samples.

We further need the following Lemma stating that classes of structures, that are ϵ-far from having a property, are closed under isomorphism.

Lemma 5.1.5. *Let $P \subseteq C_d$ be a property on C_d and $\epsilon \in (0, 1)$. Then $\epsilon\text{-far}_{C_d}(P)$ is closed under isomorphism.*

Proof. Let $\mathcal{A}, \mathcal{B} \in C_d$ be isomorphic structures, such that $\mathcal{A}$ is ϵ-far from P. Let $n := |\mathcal{B}|$. Assume that $\mathcal{B}$ is ϵ-close to P. Therefore, there is a structure $\mathcal{C} \in P$, such that $\text{dist}(\mathcal{B}, \mathcal{C}) \leq \epsilon n$. Then (D1) and (D3) imply that

$$\text{dist}(\mathcal{A}, \mathcal{C}) \leq \text{dist}(\mathcal{A}, \mathcal{B}) + \text{dist}(\mathcal{B}, \mathcal{C}) \leq 0 + \epsilon n.$$

This implies that $\frac{\mathrm{dist}(\mathcal{A},\mathcal{C})}{n} \leq \epsilon$, which contradicts the initial assumption of $\mathcal{A}$ being ϵ-far from P. Thus $\mathcal{B}$ is ϵ-far from P and ϵ-far$_{C_d}(P)$ is closed under isomorphism. □

Lemma 5.1.6. *Let $d \in \mathbb{N}$ and $P \subseteq C_d$ be a property that is non-uniformly testable with error probability $\delta \in [0, \frac{1}{2})$ and query complexity $c = c(\epsilon, \delta)$. For every ϵ there exists an ϵ-tester for P with error probability δ that operates in the following way.*

- *Uniformly sample* $\mathrm{ar}(\sigma) \cdot c$ *elements* $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$,
- *explore the c-neighbourhoods of the* $\mathrm{ar}(\sigma) \cdot c$ *sampled elements (deterministically) and*
- *decide probabilistically without making any further oracle queries to $\mathcal{A}$.*

Proof. Let T be a tester for P with error probability at most δ and query complexity c. We transform T into an ϵ-tester $\tilde{T}$ that, given access to a structure $\mathcal{A}$, operates as follows:

(1) Sample $\mathrm{ar}(\sigma) \cdot c$ elements $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c} \in A$ and explore their c-neighbourhoods, i. e. for $i = 1, \dots, \mathrm{ar}(\sigma) \cdot c$ we explore the neighbourhood of s_i with breadth-first search. We get the structure $\mathcal{N}_c^{\mathcal{A}}(s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c})$.

(2) We will emulate the execution of T on the structure $\pi(\mathcal{A}) = \big(A, \pi(R_1^{\mathcal{A}}), \dots, \pi(R_\ell^{\mathcal{A}})\big)$ as described below, where $\pi : V \to V$ is a permutation we will select during the emulation of T and $\pi(R_i^{\mathcal{A}}) = \Big\{\big(\pi(r_1), \dots, \pi(r_{\mathrm{ar}(R_i)})\big) \mid \big(r_1, \dots, r_{\mathrm{ar}(R_i)}\big) \in R_i^{\mathcal{A}}\Big\}$.
To emulate T on $\pi(\mathcal{A})$, we run T normally, but answer the queries as follows:
Let (x, j) be the next query T is making.

(2a) If $\pi(x)$ is not defined yet, we choose $a \in \{s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}\}$ uniformly out of all elements in $\{s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}\}$, that are not selected to be the π-image of any element yet, and set $\pi(x) := a$.

(2b) We determine the answer to the query $(\pi(x), j)$. If it is $\perp$, we return $\perp$ as the answer to the query (x, j). Otherwise let the answer to $(\pi(x), j)$ be $(R, b_1, \dots, b_{\mathrm{ar}(R)}) \in \sigma \times [n]^{\mathrm{ar}(R)}$. For all of the b_i, that are not in the image of π yet, we select uniformly a random element $y_i \in A$, such that $\pi(y_i)$ is not defined yet and set $\pi(y_i) := b_i$. We return $\big(R, \pi^{-1}(b_1), \dots, \pi^{-1}(b_{\mathrm{ar}(R)})\big)$ as the answer to the query (x, j).

After answering all queries, we uniformly allocate π-images for the remaining elements out of the elements, that are not in the image of π yet.

(3) Accept $\mathcal{A}$, if in (2) T accepts $\pi(\mathcal{A})$.

First note, that we can answer all queries in (2) by using $\mathcal{N}_c^{\mathcal{A}}(s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c})$ from step (1), since we are either querying elements in $\{s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}\}$ or neighbours of elements we queried before and thus the radius of c in step (1) suffices. Additionally $\mathrm{ar}(\sigma) \cdot c$ elements are sufficient, because we are defining π-image for no more than $\mathrm{ar}(\sigma)$ elements of $\{s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}\}$ per query (note that the element x we assign a π-image for in step (2a) is always one of the b_i). Therefore

$\tilde{T}$ operates in the way we claimed and we are left with the task to argue that $\tilde{T}$ is indeed an ϵ-tester for P.

To ensure that $\tilde{T}$ is an ϵ-tester for P, the choice of the permutation π should be independent of the execution of T.
Therefore let $\pi' \in \mathcal{S}_n$ be a arbitrary permutation of the elements of $\mathcal{A}$. We want to prove that $\mathbb{P}[\pi = \pi'] = \frac{1}{|\mathcal{S}_n|} = \frac{1}{n!}$.
Let $a_i \in A$ be the i-th element $\tilde{T}$ is selecting a π-image for. We can write

$$\begin{aligned} \mathbb{P}\Big[\pi = \pi'\Big] =& \mathbb{P}\Big[\pi(a_1) = \pi'(a_1)\Big] \cdot \\ & \mathbb{P}\Big[\pi(a_2) = \pi'(a_2) \Bigm| \pi(a_1) = \pi'(a_1)\Big] \cdot \\ & \qquad \vdots \\ & \mathbb{P}\Big[\pi(a_n) = \pi'(a_n) \Bigm| \pi(a_j) = \pi'(a_j) \text{ f.a. } j < n\Big] \end{aligned} \tag{5.1}$$

Let $n_{\pi',i} := |\{j \leq i \mid \pi'(a_j) \in \{s_1, \ldots, s_{\mathrm{ar}(\sigma)\cdot c}\}\}|$. In case the image of a_{i+1} is selected in (2a) we get:

$$\begin{aligned} \mathbb{P}\Big[\pi(a_{i+1}) = \pi'(a_{i+1}) \Bigm| \pi(a_j) =& \pi'(a_j) \text{ f.a. } j \leq i\Big] \\ &= \mathbb{P}\left[\pi'(a_{i+1}) \in \{s_1, \ldots, s_{\mathrm{ar}(\sigma)\cdot c}\} \;\middle|\; \begin{array}{c} \pi(a_j) = \pi'(a_j) \\ \text{f.a. } j \leq i \end{array}\right] \cdot \\ &\quad \mathbb{P}\left[\begin{array}{l} \pi'(a_{i+1}) \text{ is} \\ \text{selected in (2a)} \end{array} \;\middle|\; \begin{array}{l} \pi(a_j) = \pi'(a_j) \text{ f.a. } j \leq i, \\ \pi'(a_{i+1}) \in \{s_1, \ldots, s_{\mathrm{ar}(\sigma)\cdot c}\} \end{array}\right] \\ &= \frac{\binom{n-i-1}{\mathrm{ar}(\sigma)c - n_{\pi',i} - 1}}{\binom{n-i}{\mathrm{ar}(\sigma)c - n_{\pi',i}}} \cdot \frac{1}{\mathrm{ar}(\sigma)c - n_{\pi',i}} \\ &= \frac{1}{n-i} \end{aligned}$$

In the case the image of a_{i+1} is selected in (2b) we get

$$\begin{aligned} \mathbb{P}\Big[\pi(a_{i+1}) = \pi'(a_{i+1}) \Bigm| \pi(a_j) =& \pi'(a_j) \text{ f.a. } j \leq i\Big] \\ &= \mathbb{P}\left[\begin{array}{l} a_{i+1} \text{ is selected as preimage} \\ \text{of } \pi'(a_{i+1}) \text{ in } (2b) \end{array} \;\middle|\; \pi(a_j) = \pi'(a_j) \text{ f.a. } j \leq i\right] \\ &= \frac{1}{n-i} \end{aligned}$$

In the case that the image of a_{i+1} is selected after all queries are answered the probability $\mathbb{P}\Big[\pi(a_{i+1}) = \pi'(a_{i+1}) \Bigm| \pi(a_j) = \pi'(a_j) \text{ f.a. } j \leq i\Big] = \frac{1}{n-i}$ is given. Combined with equation (5.1) this gives us $\mathbb{P}[\pi = \pi'] = \frac{1}{n!}$.

Since $\mathcal{A}$ is isomorphic to $\pi(\mathcal{A})$ ($\pi : A \to \pi(A)$ is an isomorphism) and properties are closed under isomorphism, $\mathcal{A} \in P$ if and only if $\pi(\mathcal{A}) \in P$. Assume that $\mathcal{A} \in P$ and therefore $\pi(\mathcal{A}) \in P$. Since π is chosen independent of T, T accepts $\pi(\mathcal{A})$ with probability $1 - \delta$, since T is an ϵ-tester with error probability δ. Hence $\tilde{T}$ accepts $\mathcal{A}$ with probability $1 - \delta$, because $\tilde{T}$ accepts $\mathcal{A}$ whenever T accepts $\pi(\mathcal{A})$.

Now assume that $\mathcal{A}$ is ϵ-far from having property P. According to Lemma 5.1.5 $\pi(\mathcal{A})$ is ϵ-far from having property P and thus T rejects $\pi(\mathcal{A})$ with probability $1 - \delta$, since π is chosen independent of T. This implies that $\tilde{T}$ rejects $\mathcal{A}$ with probability $1 - \delta$ and therefore $\tilde{T}$ is an ϵ-tester for P with error probability δ, that operates in the way we claimed.

□

Lemma 5.1.7. *Let $d \in \mathbb{N}$ and $P \subseteq C_d$ be a property and T an ϵ-tester with error probability $\delta \in [0, \frac{1}{2})$, that first samples $\mathrm{ar}(\sigma) \cdot c$ elements, explores their c-balls and then decides probabilistically (like the tester we get in Lemma 5.1.6). Then there exists a ϵ-tester for P with error probability δ that operates in the following way:*

- *Uniformly sample $\mathrm{ar}(\sigma) \cdot c$ elements $s_1, \dots, s_{\mathrm{ar}(\sigma) \cdot c}$,*
- *explore the c-neighbourhoods of the $\mathrm{ar}(\sigma) \cdot c$ sampled elements (deterministically) and*
- *decide probabilistically only depending on the substructure $\mathcal{N}_c^{\mathcal{A}}(s_1, \dots, s_{\mathrm{ar}(\sigma) \cdot c})$.*

Proof. We will transform T into a tester with the required properties in two steps. We first construct a tester $\hat{T}$, that is independent of the identities of the elements in the explored substructure but depends on an order of the elements in the explored substructure. In the second step, we will construct a tester $\tilde{T}$, that is independent of any order of the elements in the explored substructure in addition.

Construction of $\hat{T}$: We can encode the exploration of the c-neighbourhood of the sampled elements $s_1, \dots, s_{\mathrm{ar}(\sigma) \cdot c}$ into a query sequence, where we consider a query sequence to be a sequence of queries, each of them of the form $(a, j) = (R, b_1, \dots, b_{\mathrm{ar}(R)})$, where $R \in \sigma$, $a \in [n]$, $(b_1, \dots, b_{\mathrm{ar}(R)}) \in [n]^{\mathrm{ar}(R)} \cup \{\perp\}$ and $1 \leq j \leq d$. We consider query sequences to be oblivious of the identities of the elements, which we can achieve by renaming the elements in the query sequence by some canonical order, i. e. rename the i-th element that appears in the query sequence with i. Let Q be the set of query sequences we can encounter, when setting T to explore any union of $\mathrm{ar}(\sigma) \cdot c$ c-balls. Note that a query sequence is a union of c-balls with a total order on its elements.

For a query sequence $\alpha \in Q$ and $\overline{s} = (s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c})$, let $p_{\overline{s},\alpha}$ be the probability of T accepting when sampling $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$ and producing the query sequence α. Note that α is independent of the actual identities of the samples. For $\alpha \in Q$ we set

$$p_\alpha := \sum_{\overline{s}\in[n]^{\mathrm{ar}(\sigma)\cdot c}} \frac{p_{\overline{s},\alpha}}{n^{\mathrm{ar}(\sigma)\cdot c}}$$

to be the expected probability of accepting $\mathcal{A}$, when obtaining the query sequence α.

Now let $\hat{T}$ be the tester, that, given access to a σ-structure $\mathcal{A}$, samples $\mathrm{ar}(\sigma)\cdot c$ elements $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$, explores there c-neighbourhood with query sequence α and accepts $\mathcal{A}$ with probability p_α.

Claim 1. For any σ-structure $\mathcal{A}$, the probability of $\hat{T}$ accepting $\mathcal{A}$ equals the probability of T accepting a random isomorphic copy of $\mathcal{A}$.

Proof of Claim 1. We identify every isomorphic copy of a structure $\mathcal{A}$ with a permutation $\pi \in \mathcal{S}_n$. The permutation π represents the isomorphic copy $\pi(\mathcal{A})$ as defined in the proof of Lemma 5.1.6. For a query sequence $\alpha \in Q$ that is acquired when exploring the neighbourhood of elements $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$, $\pi(\alpha)$ shall denote the query sequence the tester T acquires, when exploring the neighbourhood of $\pi(s_1), \dots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})$ given access to $\pi(\mathcal{A})$.

Let us fix $\overline{s} = (s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}) \subseteq [n]^{\mathrm{ar}(\sigma)\cdot c}$ of size $\mathrm{ar}(\sigma)\cdot c$ and the query sequence $\alpha \in Q$, which T acquires exploring the substructure $\mathcal{N}_c^{\mathcal{A}}(s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c})$. For any $\pi \in \mathcal{S}_n$ and any query $(a, j) = (R, b_1, \dots, b_{\mathrm{ar}(R)})$ to the structure $\mathcal{A}$, the answer to the π-image of the query $(\pi(a), j)$ to the structure $\pi(\mathcal{A})$ will be $(R, \pi(b_1), \dots, \pi(b_{\mathrm{ar}(b)}))$, which follows directly from the definition of $\pi(\mathcal{A})$. Since we rename the elements appearing in query sequences in a canonical way, this implies, that the query sequence obtained by T exploring the neighbourhood of $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$, given access to $\mathcal{A}$, equals the query sequence obtained by T exploring the neighbourhood of $\pi(s_1), \dots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})$, given access to $\pi(\mathcal{A})$. Hence

$$\mathbb{P}\big[T \text{ accepts } \pi(\mathcal{A}) \mid T \text{ samples } \pi(s_1), \dots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})\big] = p_{(\pi(s_1),\dots,\pi(s_{\mathrm{ar}(\sigma)\cdot c})),\alpha}.$$

The probability of T accepting a random isomorphic copy of $\mathcal{A}$ is the expected probability of T accepting, when choosing a random π and giving T access to $\pi(\mathcal{A})$. Therefore we get the following:

$$
\begin{aligned}
\mathbb{E}_\pi \Big[\mathbb{P}\big[T \text{ accepts } \pi(\mathcal{A}) \;\big|\; T \text{ samples } \pi(s_1), \dots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})\big] \Big] \\
&= \sum_{\pi \in \mathcal{S}_n} \frac{\mathbb{P}\big[T \text{ accepts } \pi(\mathcal{A}) \;\big|\; T \text{ samples } \pi(s_1), \dots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})\big]}{n!} \\
&= \sum_{\pi \in \mathcal{S}_n} \frac{p_{(\pi(s_1), \dots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})), \alpha}}{n!} \\
&= \sum_{\overline{s}' \in [n]^{\mathrm{ar}(\sigma)\cdot c}} \frac{p_{\overline{s}', \alpha}}{n^{\mathrm{ar}(\sigma)\cdot c}} \\
&= c_\alpha \\
&= \mathbb{P}\big[\hat{T} \text{ accepts } \mathcal{A} \;\big|\; \hat{T} \text{ samples } s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}\big].
\end{aligned}
$$

Since the probability of sampling either $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$ or $\pi(s_1), \dots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})$ for a fixed π is equal to $\frac{1}{n^{\mathrm{ar}(\sigma)\cdot c}}$, the claim follows.

■

Construction of $\tilde{T}$: For any $\alpha \in Q$ let C_α be the set of all structures that are isomorphic to the structure underlying the query sequence α, that is the structure without the order on the elements. For every union $\mathcal{B}$ of (up to) $\mathrm{ar}(\sigma) \cdot c$ different c-balls we set

$$
c_{\mathcal{B}} := \sum_{\substack{\alpha \in Q, \\ C_\alpha \ni \mathcal{B}}} \frac{p_\alpha}{|\{\alpha \in Q \mid \mathcal{B} \in C_\alpha\}|}
$$

to be the expected probability of accepting $\mathcal{A}$, when the explored structure is $\mathcal{B}$.

Now let $\tilde{T}$ be the tester which, given access to a σ-structure $\mathcal{A}$, samples $\mathrm{ar}(\sigma) \cdot c$ elements $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$, explores their c-neighbourhood and accepts $\mathcal{A}$ with probability $c_{\mathcal{N}_c^{\mathcal{A}}(s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c})}$.

Claim 2. For any σ-structure $\mathcal{A}$, the probability of $\tilde{T}$ accepting $\mathcal{A}$ equals the probability of $\hat{T}$ accepting a random representation of a random isomorphic copy of $\mathcal{A}$.

Proof of Claim 2. We identify every representation of an isomorphic copy of a structure $\mathcal{A}$ with a tuple $\overline{\pi} = (\pi, \pi_0, \dots, \pi_{n-1})$ of $n+1$ permutations, where $\pi \in \mathcal{S}_n$ and $\pi_i \in \mathcal{S}_{\deg_{\mathcal{A}}(i)}$ for $0 \le i \le n-1$. Here the permutation π gives us the actual isomorphic copy, namely $\pi(\mathcal{A})$ as defined in the proof of Lemma 5.1.6, and the permutations π_i account for getting all the equivalent representations of $\pi(\mathcal{A})$ by reordering the tuples containing a certain element $\pi(i)$. We denote the copy of $\mathcal{A}$ identified with a tuple $\overline{\pi}$ by $\overline{\pi}(\mathcal{A})$. Note that when picking a tuple $\overline{\pi}$ uniformly every isomorphic copy $\pi(\mathcal{A})$ of $\mathcal{A}$ is equally likely to appear as the number of tuples $\overline{\pi}$ in which the first entry equals π is the same for every permutation π. For a query sequence $\alpha \in Q$ that is acquired when exploring the neighbourhood of a elements $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$, $\overline{\pi}(\alpha)$

shall denote the query sequence the tester T acquires, when exploring the neighbourhood of $\pi(s_1), \ldots, \pi(s_{\mathrm{ar}(\sigma)\cdot c})$ given access to $\overline{\pi}(\mathcal{A})$.

Let us fix a query sequence $\alpha \in Q$ and the underlying structure $\mathcal{B}$, which is the union of $\mathrm{ar}(\sigma) \cdot c$ c-balls. Therefore $\mathcal{B} \in C_\alpha$. Letting $\overline{\pi}$ range over all tuples $(\pi, \pi_0, \ldots, \pi_{n-1}) \in \mathcal{S}_n \times \left(\mathcal{S}_{\deg_{\mathcal{A}}(i)}\right)_{i=0}^{n-1}$, we get the following:

$$\begin{aligned}
\mathbb{E}_{\overline{\pi}}\Big[\mathbb{P}\big[\hat{T} \text{ accepts } \overline{\pi}(\mathcal{A}) \bigm| \hat{T} \text{ obtains } \overline{\pi}(\alpha)\big]\Big] &= \sum_{\overline{\pi}=(\pi,\pi_1,\ldots,\pi_n)} \frac{\mathbb{P}\big[\hat{T} \text{ accepts } \overline{\pi}(\mathcal{A}) \bigm| \hat{T} \text{ obtains } \overline{\pi}(\alpha)\big]}{n! \cdot \prod_{i=0}^{n-1} \deg_{\mathcal{A}}(\pi(i))!} \\
&= \sum_{\substack{\alpha' \in Q, \\ \mathcal{B} \in C_{\alpha'}}} \frac{p_{\alpha'}}{|\{\alpha' \in Q \mid \mathcal{B} \in C_{\alpha'}\}|} \\
&= c_{\mathcal{B}} \\
&= \mathbb{P}\big[\tilde{T} \text{ accepts } \mathcal{A} \bigm| \tilde{T} \text{ obtains } \alpha\big],
\end{aligned}$$

where in the second equality we use, that $\mathcal{B} \in C_\alpha$ implies that $\mathcal{B} \in C_{\overline{\pi}(\alpha)}$, and, that the set of tuples $\overline{\pi}$ fixing a certain query sequence $\alpha' \in Q$ is the same for every query sequence and therefore we have to weigh all α' equally.

Taking into account that the probability of $\hat{T}$ or $\tilde{T}$ obtaining a certain query sequence is equal (it depends on sampling the right sample set), the claim follows. ■

To show that $\tilde{T}$ is a ϵ-tester for P with error probability δ, let us first independently choose $\pi \in \mathcal{S}_n$ and a tuples $\overline{\pi}' = (\pi', \pi'_0, \ldots, \pi'_{n-1}) \in \mathcal{S}_n \times \left(\mathcal{S}_{\deg_{\mathcal{A}}(i)}\right)_{i=0}^{n-1}$ uniformly at random. We observe that the structure $\overline{\pi}'(\mathcal{A})$ is a random isomorphic copy of $\mathcal{A}$ and, due to the independence of π and $\overline{\pi}'$, the structure $\pi \circ \overline{\pi}'(\mathcal{A})$ is a random isomorphic copy of $\overline{\pi}'(\mathcal{A})$.
Let us now assume that $\mathcal{A} \in C_d$ has property P. Since properties are closed under isomorphism and isomorphisms are closed under composition, the structure $\pi \circ \overline{\pi}'(\mathcal{A})$ has property P. Therefore T accepts $\pi \circ \overline{\pi}'(\mathcal{A})$ with probability at least $1 - \delta$. According to claim 1 this implies that $\hat{T}$ accepts $\overline{\pi}'(\mathcal{A})$ with probability at least $1 - \delta$. Applying claim 2 we get that $\tilde{T}$ accepts $\mathcal{A}$ with probability at least $1 - \delta$.
For any structure $\mathcal{A} \in C_d$ that is ϵ-far from having property P, Lemma 5.1.5 implies, that $\pi \circ \overline{\pi}'(\mathcal{A}) \in \epsilon - \mathrm{far}_{C_d}(P)$. Therefore T rejects $\pi \circ \overline{\pi}'(\mathcal{A})$ with probability at least $1 - \delta$. Using claim 1 and claim 2 we get that $\tilde{T}$ rejects $\mathcal{A}$ with probability at least $1 - \delta$. Therefore $\tilde{T}$ is an ϵ-tester with error probability δ, who's decision depends only on the explored substructure. □

Now we are able to carry out the final step of the transformation of any tester into a canonical tester. This step yields a tester, that queries $\mathcal{N}_c^{\mathcal{A}}(S)$ and then decides deterministically.

Lemma 5.1.8. *Let C_d be a class of σ-structures of bounded degree $d : \mathbb{N} \to \mathbb{R}_{\geq 0}$. Let $P \subseteq C_d$ be a property and T a ϵ-tester with error probability $\delta \in [0, \frac{1}{2})$, that first samples* $\mathrm{ar}(\sigma) \cdot c$ *elements*

$s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$, *explores their c-balls and then decides probabilistically depending only on the substructure* $\mathcal{N}_c^{\mathcal{A}}(s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c})$ *(like the tester we get in Lemma 5.1.7). Then there exists a* ϵ*-tester for* P *with error probability* 2δ *that operates in the following way:*

- *Uniformly samples* $\mathrm{ar}(\sigma) \cdot c$ *elements* $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$,
- *explore the c-neighbourhoods of the* $\mathrm{ar}(\sigma) \cdot c$ *sampled elements (deterministically) and*
- *makes a deterministic decision based on* $\mathcal{N}_c^{\mathcal{A}}(s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c})$.

Proof. For any union of $\mathrm{ar}(\sigma) \cdot c$ c-balls $\mathcal{B}$ we define $c_{\mathcal{B}}$ to be the probability of T accepting, when the explored substructure is isomorphic to $\mathcal{B}$, as defined in Lemma 5.1.7.
Let $\tilde{T}$ be the tester, that, given access to a structure $\mathcal{A}$, samples $\mathrm{ar}(\sigma)\cdot c$ elements $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$, explores the c-neighbourhood of $s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}$ and then accepts $\mathcal{A}$ if $c_{\mathcal{N}_c^{\mathcal{A}}(s_1,\dots,s_{\mathrm{ar}(\sigma)\cdot c})} \geq \frac{1}{2}$.

To show that $\tilde{T}$ is a ϵ-tester we use the following claim.

Claim 1. Let $p_1, \dots, p_k \in [0, 1]$ for some $k \in \mathbb{N}_{>0}$ and $\sum_{i=1}^{k} \frac{p_i}{k} \geq 1-\delta$. Then at least $\lceil (1-2\delta)k \rceil$ of the p_i are larger or equal than $\frac{1}{2}$.

Proof of Claim 1. Assume that the proposition is false, then at least $\lfloor 2\delta k \rfloor + 1$ of the p_i are smaller than $\frac{1}{2}$. Therefore, by using $\lfloor 2\delta k \rfloor + 1 \geq \lceil 2\delta k \rceil \geq 2\delta k$ and bounding the remaining p_i by 1 from above, we get:

$$\sum_{i=1}^{k} \frac{p_i}{k} < \frac{2\delta k \cdot \frac{1}{2} + (1-2\delta)k}{k} = 1 - \delta,$$

which is a contradiction to the assumption. ■

To show that $\tilde{T}$ is an ϵ-tester with error probability 2δ for P, let us first assume that $\mathcal{A} \in P$. Since $\tilde{T}$ accepts with probability $\left\lfloor p + \frac{1}{2} \right\rfloor = \begin{cases} 1, & \text{for } p \geq \frac{1}{2} \\ 0, & \text{for } p < \frac{1}{2} \end{cases}$, if T accepts with probability $p \in [0, 1]$, the probability that $\tilde{T}$ accepts $\mathcal{A}$ can be bounded from below as follows:

$$\begin{aligned}
&\mathbb{P}\big[\tilde{T} \text{ accepts } \mathcal{A}\big] \\
&= \sum_{s_1,\dots,s_{\mathrm{ar}(\sigma)\cdot c} \in A} \mathbb{P}\big[\tilde{T} \text{ samples } s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}\big] \cdot \mathbb{P}\big[\tilde{T} \text{ accepts } \mathcal{A} \mid \tilde{T} \text{ samples } s_1, \dots, s_{\mathrm{ar}(\sigma)\cdot c}\big] \\
&= \sum_{s_1,\dots,s_{\mathrm{ar}(\sigma)\cdot c} \in A} \frac{1}{n^{\mathrm{ar}(\sigma)\cdot c}} \left\lfloor c_{\mathcal{N}_c^{\mathcal{A}}(s_1,\dots,s_{\mathrm{ar}(\sigma)\cdot c})} + \frac{1}{2} \right\rfloor \\
&\geq 1 - 2\delta,
\end{aligned}$$

where the last inequality uses claim 1 and that

$$\mathbb{P}[T \text{ accepts } \mathcal{A}] = \sum_{s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c} \in A} \frac{1}{n^{\mathrm{ar}(\sigma)\cdot c}} c_{\mathcal{N}_c^{\mathcal{A}}(s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c})} \geq 1-\delta,$$

since T is a tester and $\mathcal{A} \in P$.

Now assume that the structure $\mathcal{A}$ is ϵ-far from having property P. Since $\tilde{T}$ rejects with probability $\left\lceil \frac{1}{2} - p \right\rceil = \begin{cases} 1, & \text{for } p < \frac{1}{2} \\ 0, & \text{for } p \geq \frac{1}{2} \end{cases}$, if T accepts with probability $p \in [0,1]$, the probability that $\tilde{T}$ rejects $\mathcal{A}$ can be bound from below as follows:

$$\begin{aligned}
&\mathbb{P}\big[\tilde{T} \text{ rejects } \mathcal{A}\big] \\
&= \sum_{s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c} \in A} \mathbb{P}\big[\tilde{T} \text{ samples } s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c}\big] \cdot \mathbb{P}\big[\tilde{T} \text{ rejects } \mathcal{A} | \tilde{T} \text{ samples } s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c}\big] \\
&= \sum_{s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c} \in A} \frac{1}{n^{\mathrm{ar}(\sigma)\cdot c}} \left\lceil \frac{1}{2} - c_{\mathcal{N}_c^{\mathcal{A}}(s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c})} \right\rceil \\
&\geq 1 - 2\delta,
\end{aligned}$$

where the last inequality uses, that

$$\mathbb{P}[T \text{ rejects } \mathcal{A}] = \sum_{s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c} \in A} \frac{1}{n^{\mathrm{ar}(\sigma)\cdot c}} (1 - c_{\mathcal{N}_c^{\mathcal{A}}(s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c})}) \geq 1-\delta,$$

and therefore, according to claim 1, at least $\lceil (1-2\delta)\binom{n}{\mathrm{ar}(\sigma)\cdot c} \rceil$ of the $c_{\mathcal{N}_c^{\mathcal{A}}(s_1,\ldots,s_{\mathrm{ar}(\sigma)\cdot c})}$ are larger than or equal to $\frac{1}{2}$.
Hence $\tilde{T}$ is an ϵ-tester with error probability 2δ for P. $\square$

We are now able to conclude the existence of a canonical tester as in Lemma 5.1.3.

Proof of Lemma 5.1.3. Let T be any non-uniform ϵ-tester with error probability δ and query complexity $c = c(\epsilon, \delta)$ for the property P on C_d.
We first use Lemma 5.1.6, Lemma 5.1.7 and Lemma 5.1.8 to transform T into a ϵ-tester $\tilde{T}$ with error probability 2δ for P on C_d, that, given access to a structure $\mathcal{A}$, uniformly samples $\mathrm{ar}(\sigma) \cdot c$ elements $s_1, \ldots, s_{\mathrm{ar}(\sigma)\cdot c}$, explore the c-neighbourhoods of the $\mathrm{ar}(\sigma) \cdot c$ sampled elements and makes a deterministic decision based on $\mathcal{N}_c^{\mathcal{A}}(s_1, \ldots, s_{\mathrm{ar}(\sigma)\cdot c})$.
We now define F_n to be the set of all structures $\mathcal{B}$, such that $\mathcal{B}$ is the unions of $\mathrm{ar}(\sigma) \cdot c$ c-balls and $\hat{T}$ rejects when the explored substructure is $\mathcal{B}$. Therefore $\tilde{T}$ is a tester, that has all the requirements we requested. $\square$

5.2 Small structures approximating neighbourhood distributions

The aim of this Section is to introduce a result proved by Alon [102, Proposition 19.10] for simple bounded degree graphs which states that for any given precision λ there is a constant size n_0 such that the neighbourhood distribution of any bounded degree graph G can be approximately realised with precision λ by a "small" bounded degree graph H with no more than n_0 vertices. This result gives us insights into the structure of bounded degree properties. And as such it was extensively used in [56] to prove that every testable property of bounded degree graphs has a hyperfinite subproperty. This on the other hand implies that if a property of bounded degree graphs is a class of expanders it can not be testable. To this end we will utilise the result by Alon [102, Proposition 19.10] later when proving non-testability for certain properties. While the existence of such small graphs approximating the neighbourhood distribution of large graphs is unquestionably very useful for property testing, the question of (algorithmically) determining such a small graph H for a given precision λ and graph G is also of interest. An answer was determined in the special case of high girth bounded degree graphs [55].

The result proved by Alon [102, Proposition 19.10] for simple graphs holds for relational structures also and can be proven with a very similar argument as for simple graphs given in [102, Proposition 19.10]. For completeness sake we will give a proof for relational structures in the following.

Let σ be a signature, $d \in \mathbb{N}$ and C_d be the class of σ-structures of bounded degree d. We let T_r be the set of all r-types of bounded degree d and $\rho_{\mathcal{A},r}$ the r-neighbourhood distribution defined in Definition 2.3.7.

Definition 5.2.1 (Sampling distance)**.** For two σ-structures $\mathcal{A}, \mathcal{B} \in C_d$ we define the sampling distance of depth r as

$$\delta_{\odot}^{r}(\mathcal{A}, \mathcal{B}) := \max_{X \subseteq T_r} |\rho_{\mathcal{A},r}(X) - \rho_{\mathcal{B},r}(X)|.$$

Then the sampling distance of $\mathcal{A}$ and $\mathcal{B}$ is defined as

$$\delta_{\odot}(\mathcal{A}, \mathcal{B}) := \sum_{r=0}^{\infty} \frac{1}{2^r} \cdot \delta_{\odot}^{r}(\mathcal{A}, \mathcal{B}).$$

Note that $\delta_{\odot}^{r}(\mathcal{A}, \mathcal{B})$ is just the total variance distance between the distributions $\rho_{\mathcal{A},r}, \rho_{\mathcal{B},r}$, and it holds that $\delta_{\odot}^{r}(\mathcal{A}, \mathcal{B}) = \frac{1}{2} \|\mathrm{freq}_r(\mathcal{A}) - \mathrm{freq}_r(\mathcal{B})\|_1$, where $\|\mathrm{freq}_r(\mathcal{A}) - \mathrm{freq}_r(\mathcal{B})\|_1$ is the expression used in Definition 5.1.1.

We can now state a relational structure version of [102, Proposition 19.10].

Theorem 5.2.2. *For every $\lambda > 0$ there is a positive integer n_0 such that for every σ-structure $\mathcal{A} \in C_d$ there is a σ-structure $\mathcal{H} \in C_d$ such that $|H| \leq n_0$ and $\delta_{\odot}(\mathcal{A}, \mathcal{H}) \leq \lambda$.*

Proof. Let $r := \lceil \log(\frac{2}{\lambda}) \rceil$ and t be the number of r-types of bounded degree d, i. e. $t := |T_r|$. Let $\mathrm{dist} : \mathbb{R}^t \times \mathbb{R}^t \to [0, \infty)$ be the metric defined by

$$\mathrm{dist}(\overline{x}, \overline{y}) = \max_{I \subseteq [t]} \Big| \sum_{i \in I} \overline{x}_i - \sum_{i \in I} \overline{y}_i \Big|.$$

Since $[0,1]^t$ is a bounded and closed subset of $\mathbb{R}^t$ we get by the Heine-Borel Theorem that $[0,1]^t$ is compact. Hence there is a finite subset $\mathcal{U}$ of the set of all open $\frac{\lambda}{8}$-balls $\{U_{\lambda/8}(\overline{x}) \mid \overline{x} \in \mathbb{R}^t\}$ such that $[0,1]^t \subseteq \bigcup_{U \in \mathcal{U}} U$, where the open $\frac{\lambda}{8}$- ball around $\overline{x} \in \mathbb{R}^t$ is defined as $U_{\lambda/8}(\overline{x}) := \{\overline{y} \in \mathbb{R}^t \mid \mathrm{dist}(\overline{x}, \overline{y}) < \frac{\lambda}{8}\}$.

Let $P_r \subseteq [0,1]^t$ be the set of all frequency vectors $\mathrm{freq}_r(\mathcal{A})$ where $\mathcal{A}$ ranges through all finite σ-structures of bounded degree d. Note that we defined dist in such a way that $\delta^r_\odot(\mathcal{A}, \mathcal{B}) = \mathrm{dist}(\mathrm{freq}_r(\mathcal{A}), \mathrm{freq}_r(\mathcal{B}))$ for any $\mathcal{A}, \mathcal{B} \in C_d$. Hence $\mathrm{freq}_r(\mathcal{A})$ and $\mathrm{freq}_r(\mathcal{B})$ cannot be contained in the same open $\frac{\lambda}{8}$-ball $U \in \mathcal{U}$ for every two σ-structures $\mathcal{A}, \mathcal{B} \in C_d$ with $\delta^r_\odot(\mathcal{A}, \mathcal{B}) > \frac{\lambda}{4}$. Hence any maximal family of σ-structures $\mathcal{A}_1, \mathcal{A}_2, \dots$ of bounded degree d with $\delta^r_\odot(\mathcal{A}_i, \mathcal{A}_j) > \frac{\lambda}{4}$ for $i \neq j$ has to be finite. Let $\mathcal{A}_1, \dots, \mathcal{A}_m$ be such a maximal family. We need the following claim.

Claim 1. For any two σ-structures $\mathcal{A}, \mathcal{B} \in C_d$ and any $r \in \mathbb{N}_{\geq 1}$ the following equation holds.

$$\delta_\odot(\mathcal{A}, \mathcal{B}) \leq 2 \cdot \delta^r_\odot(\mathcal{A}, \mathcal{B}) + \frac{1}{2^r}. \tag{5.2}$$

Proof of Claim 1. Note that $\delta^i_\odot(\mathcal{A}, \mathcal{B}) \leq \delta^{i+1}_\odot(\mathcal{A}, \mathcal{B})$ for any $i \in \mathbb{N}$. This is the case as $\rho_{\tilde{\mathcal{A}},r}(X) = \rho_{\tilde{\mathcal{A}},r+1}(X^{\mathrm{ext}})$ for any $\tilde{\mathcal{A}} \in C_d$, any $X \subseteq T_r$ and $X^{\mathrm{ext}} := \{\tau \in T_{r+1} \mid \tau|_r \in X\}$ the set of extensions of types in τ, where $\tau|_r$ is the restriction of type τ to radius r, i. e. $\tau|_r$ is the r-type such that $(\mathcal{N}^{\mathcal{B}}_r(b), b) \in \tau|_r$ for $(\mathcal{B}, b) \in \tau$.

Furthermore note that

$$\sum_{i=0}^r \frac{1}{2^i} = 2(1 - \frac{1}{2}) \sum_{i=0}^r \frac{1}{2^i} = 2 \sum_{i=0}^r (\frac{1}{2^i} - \frac{1}{2^{i+1}}) = 2 - \frac{1}{2^r}$$

and hence we get, making use of that $\sum_{i=0}^\infty \frac{1}{2^i}$ is a geometric series

$$\begin{aligned}
\delta_\odot(\mathcal{A}, \mathcal{B}) &= \sum_{i=0}^r \frac{1}{2^i} \delta^i_\odot(\mathcal{A}, \mathcal{B}) + \sum_{i=r+1}^\infty \frac{1}{2^i} \delta^i_\odot(\mathcal{A}, \mathcal{B}) \\
&\leq \delta^r_\odot(\mathcal{A}, \mathcal{B}) \sum_{i=0}^r \frac{1}{2^i} - \sum_{i=0}^r \frac{1}{2^i} + \sum_{i=0}^\infty \frac{1}{2^i} \\
&\leq 2 \cdot \delta^r_\odot(\mathcal{A}, \mathcal{B}) - (2 - \frac{1}{2^r}) + 2,
\end{aligned}$$

where we use that $\delta^i_\odot(\mathcal{A}, \mathcal{B}) \leq 1$ in the first inequality. ∎

By construction of the family $\mathcal{A}_1, \cdots \mathcal{A}_m$ we have that for any σ-structure $\mathcal{A}$ of bounded degree d there is $i \in [m]$ such that $\delta_{\odot}^{r}(\mathcal{A}, \mathcal{A}_i) \leq \frac{\lambda}{4}$. Then by Claim 1 and choice of r

$$\delta_{\odot}(\mathcal{A}, \mathcal{A}_i) \leq 2 \cdot \delta_{\odot}^{r}(\mathcal{A}, \mathcal{A}_i) + \frac{1}{2^r} \leq \frac{\lambda}{2} + \frac{1}{2^r} \leq \lambda.$$

Hence by setting $n := \max_{1 \leq i \leq m} |A_i|$ we have proven the theorem. $\square$

5.3 Summary

In this section we verified that two results which we require for relational structures can be generalised from simple graphs to relational structures. More specifically, we proved a relational structure version of the canonical tester from [34] and a relational structure version of [102, Theorem 19.10]. Both generalisations are straight forward using the same argumentation as in the original proofs.

Chapter 6

Classifying testability of first-order properties by prefix classes

In this chapter we consider testability of FO definable properties in the bounded degree model according to prefix classes, inspired by a similar study by Alon et al. [6] for the dense graph model. It is easy to observe that properties defined by sentences without quantifier alterations (sentences that are in $\Sigma_1 \cup \Pi_1$) are testable. Every sentence φ in Σ_1 is equivalent to a sentence of the form $\exists x_1 \dots \exists x_k \psi(x_1, \dots, x_k)$ where $\psi(x_1, \dots, x_k)$ is a quantifier-free formula. Editing the substructure required by ψ into a structure takes a constant amount of edge modifications implying that every large enough structure is ϵ-close to satisfying φ. This implies that we can test satisfiability of such a formula by checking precisely for every small structure and by accepting any large enough structure. Conversely, every sentence φ in Π_1 is equivalent to a sentence of the form $\neg\exists x_1 \dots \exists x_k \psi(x_1, \dots, x_k)$ where $\psi(x_1, \dots, x_k)$ is a quantifier-free formula. Therefore testing φ amounts to testing absence of a finite set of forbidden induced substructure, which can be done similarly to testing subgraph freeness [73]. For sentences with at least one quantifier alternation testability is less clear. In this chapter we prove the following theorem classifying which prefix classes of FO yield testable properties in the bounded degree model.

Theorem 6.0.1. *Every FO-sentence $\varphi \in \Sigma_2$ defines a testable property in the bounded degree model. On the other hand, there is a property in Π_2 which is not testable in the bounded degree model.*

To simplify the argument we obtain the non-testable property in the relational structure model and argue in Section 6.3 how to obtain the result in the bounded degree model for graphs. Testability for sentences in Σ_2 is obtained for both models.

It is worth taking note that the dividing line in Theorem 6.0.1 is the same as for FO properties in the dense graph model [6]. This is surprising taking into account the very different nature of the two models. Specifically, the dense model characterisation uses a variation of the regularity lemma which cannot be applied in the bounded degree model. In order to prove Theorem 6.0.1 we develop new proof techniques combining graph theory, logic and property testing. Furthermore, Theorem 6.0.1 answers the open question whether FO definable properties are testable in the bounded degree model raised by Adler and Harwath in [2].

Proof outline To prove that there is a property defined by a sentence in Π_2 which is not testable in the bounded degree model we define an FO-sentence $\varphi_{Ⓩ}$ which encodes an elaborate construction of a class of relational structures. The construction uses the zig-zag product which was introduced by Reingold, Vadhan and Wigderson in [115]. We give a detailed overview of the construction at the beginning of Section 6.1.2. The sentence $\varphi_{Ⓩ}$ defines a class of edge expanders which we will show in Section 6.1.3. To prove expansion we first show what the structural appearance of the models of $\varphi_{Ⓩ}$ is (Lemma 6.1.15), for which the core part is proving connectivity (Lemma 6.1.14). We further use the structural appearance of the models of $\varphi_{Ⓩ}$ to prove expansion using the preservation of edge expansion by the zig-zag product (Proposition 6.1.12). We prove that the property $P_{Ⓩ}$ defined by $\varphi_{Ⓩ}$ is not testable in Section 6.2 (Theorem 6.2.1). We use Theorem 5.2.2 to show the existence of arbitrarily large structures whose frequency vectors approximately look like the frequency vector of a model of $\varphi_{Ⓩ}$ but which are far from $P_{Ⓩ}$. These structures being far from $P_{Ⓩ}$ is ensured by the edge expansion of the models of $\varphi_{Ⓩ}$. Non-testability of $P_{Ⓩ}$ follows using Theorem 5.1.2. We further show in Section 6.2 that every sentence which only has d-regular models is d-equivalent to a sentence in Π_2 (Lemma 6.2.3). Note that this can be applied to our property as d-regularity of models of $\varphi_{Ⓩ}$ is ensured in the construction. Combining Theorem 6.2.1 and Lemma 6.2.3 we obtain that there is a sentence in Π_2 which defines a non-testable property (Theorem 6.2.4). Finally, in Section 6.3 we show that we can obtain a sentence in Π_2 which defines a property of simple undirected graphs and is not testable in the bounded degree model for graphs (Theorem 6.3.1). This is shown by carefully replacing tuples by d-regular graph gadgets and arguing that edge expansion is maintained (Proposition 6.3.4).

In Section 6.4 we prove that every sentence in Σ_2 defines a testable property (Theorem 6.4.1). This applies to both the bounded degree model of relational structures and the bounded degree model for graphs. We obtain Theorem 6.4.1 by proving that every property defined by a sentence in Σ_2 is a union of properties each of which is indistinguishable from a property in Π_1. The notion of indistinguishability of properties was introduced in [6] for the dense model and entails that testability of two indistinguishable properties is equivalent. Since properties defined by sentences in Π_1 are testable and property testing is closed under unions (Corollary 3.5.2) this shows testability of every property defined by a sentence in Σ_2. The main challenge here is to deal with the interactions between existentially quantified variables and universally quantified variables. Intuitively, the degree bound limits the structure that can be imposed by the universally quantified variables. Using this, we are able to deal with the existential variables together with these interactions. We prove that editing in a required constant size substructure for every structure of a property results in a property which is indistinguishable.

Combining Theorem 6.3.1 and Theorem 6.4.1 proves Theorem 6.0.1.

6.1 A class of expanders definable in FO

In this section we construct a formula which defines an infinite property consisting of expanders. The construction is based on the zig-zag product of graphs introduced in [115], which maintains expansion as was shown in [115]. Since a detailed understanding of the zig-zag product is key for the construction of our formula, we will introduce the zig-zag product and its properties in detail in the following.

6.1.1 Expansion and the zig-zag product

In this section we recall a construction of a class of expanders introduced in [87]. This class is defined by recursively applying some graph operations (taking the zig-zag product, squaring) which are defined via so called rotation maps. Rotation maps represent graphs similar to adjacency lists. They depend on an ordering on the neighbours of each vertex. Furthermore, fixing an ordering of neighbours for every vertex corresponds to a rotation map and hence in particular for every graph there exists a rotation map. Note that in this subsection all graphs are multigraphs (graphs with parallel edges and self-loops) as in the original work [115].

Definition 6.1.1. Let $D \in \mathbb{N}$ and $G = (V, E, f)$ be a D-regular graph on N vertices and I be a set of size D. Then a *rotation map* of G is a function $\mathrm{ROT}_G : V \times I \to V \times I$ such that for every two not necessary different vertices $u, v \in V$

$$|\{(i, j) \in I \times I \mid \mathrm{ROT}_G(u, i) = (v, j)\}| = 2 \cdot |\{e \in E \mid f(e) = \{u, v\}\}|$$

and ROT_G is self inverse, i.e. $\mathrm{ROT}_G(\mathrm{ROT}_G(v, i)) = (v, i)$ for all $v \in V$, $i \in I$.

There is a tight connection between certain properties of a graph and the eigenvalues of its adjacency matrix, which we will recall in the following. For a D-regular graph $G = (V, E, f)$ we let the normalised adjacency matrix M of G be defined by

$$M_{u,v} := \frac{1}{D} \cdot |\{e \mid f(e) = \{u, v\}\}|.$$

Definition 6.1.2 (Spectrum of a graph)**.** Let $D \in \mathbb{N}$, $G = (V, E, f)$ be a D-regular graph and M the normalised adjacency matrix of G. The eigenvalues of M are called the *spectrum* of G. We denote the eigenvalues of M by $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$. We define

$$\lambda(G) := \max\{|\lambda_2|, |\lambda_N|\}.$$

We say that a graph is an (N, D, λ)-graph, if G has N vertices, is D-regular and $\lambda(G) \leq \lambda$.

Since M is real, symmetric, contains no negative entries and all columns sum up to 1, all its eigenvalues are in the real interval $[-1, 1]$. Furthermore 1 is always an eigenvalue of M corresponding to the eigenvector $\mathbf{1}_N := (1, \ldots, 1)^t$ as G is D-regular and M is normalised.

The spectrum encodes the following properties of a graph.

Lemma 6.1.3 ([87])**.** *The graph G is connected if and only if $\lambda_2 < 1$. Furthermore, if G is connected, then G is bipartite if and only if $\lambda_N = -1$.*

There is also the following connection between $h(G)$ and $\lambda(G)$, where $h(G)$ is the expansion ratio defined in Definition 2.2.1.

Theorem 6.1.4 ([10, 43])**.** *Let G be a D-regular graph on N vertices. Then*

$$h(G) \geq \frac{D - D \cdot \lambda(G)}{2}.$$

This implies that for a sequence $\{G_N\}_{N\in\mathbb{N}}$ of graphs of increasing number of vertices, if there is a constant $\epsilon < 1$ such that $\lambda(G_N) \leq \epsilon$ for all $N \in \mathbb{N}$, then the sequence $\{G_N\}_{N\in\mathbb{N}}$ is a family of $\frac{D(1-\varepsilon)}{2}$-expanders.

We now define the basic graph operations used to recursively define a class of expanders and their properties.

Definition 6.1.5. Let $G = (V, E, f)$ be a D-regular graph on N vertices with rotation map $\mathrm{ROT}_G : V \times I \to V \times I$ where I is a set of size D. Then the *square of* G, denoted by G^2, is a D^2-regular graph on V with rotation map $\mathrm{ROT}_{G^2}(u, (k_1, k_2)) := (w, (\ell_2, \ell_1))$, where

$$\begin{aligned} \mathrm{ROT}_G(u, k_1) =& (v, \ell_1) \text{ and} \\ \mathrm{ROT}_G(v, k_2) =& (w, \ell_2), \end{aligned}$$

and $u, v, w \in V$, $k_1, k_2, \ell_1, \ell_2 \in I$.

Note that the edges of G^2 correspond to walks of length 2 in G and the adjacency matrix of G^2 is the square of the adjacency matrix of G. Note here that if G is bipartite then G^2 is not connected, which can be easily explained by using Lemma 6.1.3.

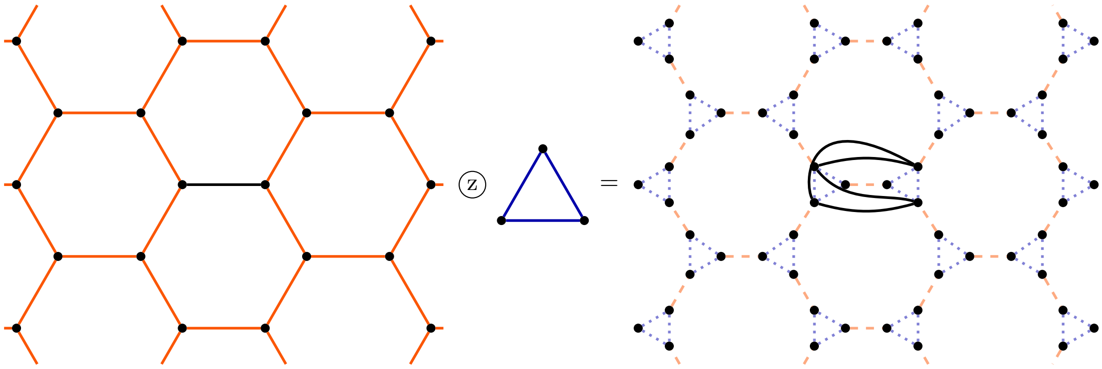


Figure 6.1: Zig-zag product of a 3-regular grid with a triangle.

Lemma 6.1.6 ([115]). *If G is an (N, D, λ)-graph then G^2 is an (N, D^2, λ^2)-graph.*

Definition 6.1.7. Let $G_1 = (V_1, E_1, f_1)$ be a D_1-regular graph on N_1 vertices, I_1 a set of size D_1 and $\mathrm{ROT}_{G_1} : V_1 \times I_1 \to V_1 \times I_1$ a rotation map of G_1. Let $G_2 = (I_1, E_2, f_2)$ be a D_2-regular graph, let I_2 be a set of size D_2 and $\mathrm{ROT}_{G_2} : I_1 \times I_2 \to I_1 \times I_2$ be a rotation map of G_2. Then the *zig-zag product of* G_1 *and* G_2, denoted by $G_1 ⓩ G_2$, is the D_2^2-regular graph on vertex set $V_1 \times I_1$ with rotation map given by $\mathrm{ROT}_{G_1 ⓩ G_2}((v, k), (i, j)) := ((w, \ell), (j', i'))$, where

$$\begin{aligned}
&\mathrm{ROT}_{G_2}(k, i) = (k', i'),\\
&\mathrm{ROT}_{G_1}(v, k') = (w, \ell'), \text{ and}\\
&\mathrm{ROT}_{G_2}(\ell', j) = (\ell, j'),
\end{aligned}$$

and $v, w \in V_1$, $k, k', \ell, \ell' \in I_1$, $i, i', j, j' \in I_2$.

The zig-zag product $G_1 ⓩ G_2$ can be seen as the result of the following construction. First pick some numbering of the vertices of G_2. Then replace every vertex in G_1 by a copy of G_2 where we colour edges from G_1, say, red, and edges from G_2 blue. We do this in such a way that the i-th edge in G_1 of a vertex v will be incident to vertex i of the copy of G_2 corresponding to v. Then for every red edge (v, w) and for every tuple $(i, j) \in I_2 \times I_2$ we add an edge to the zig-zag product $G_1 ⓩ G_2$ connecting v' and w' where v' is the vertex reached from v by taking its i-th blue edge and w' can be reached from w by taking its j-th blue edge. Figure 6.1 shows an example, wherein the graph on the right hand side we show the 4 edges that are added to the zig-zag product for the highlighted edge of the graph on the left hand side.

Theorem 6.1.8 ([115]). *If G_1 is an (N_1, D_1, λ_1)-graph and G_2 is a (D_1, D_2, λ_2)-graph then $G_1 ⓩ G_2$ is an $(N_1 \cdot D_1, D_2^2, g(\lambda_1, \lambda_2))$-graph, where*

$$g(\lambda_1, \lambda_2) = \frac{1}{2}(1 - \lambda_2^2)\lambda_1 + \frac{1}{2}\sqrt{(1 - \lambda_2^2)^2\lambda_1 + 4\lambda_2^2}.$$

This function has the following properties.

1. *If both* $\lambda_1 < 1$ *and* $\lambda_2 < 1$ *then* $g(\lambda_1, \lambda_2) < 1$.
2. $g(\lambda_1, \lambda_2) < \lambda_1 + \lambda_2$.

We can now recursively define a class of expanders. We use the expander construction from [87].

Definition 6.1.9 ([87])**.** Let D be a sufficiently large prime power (e.g. $D = 2^{16}$). Let H be a $(D^4, D, 1/4)$ expander (an explicit constructions for H exist, see [115].) We define $\{G_\ell\}_{\ell\in\mathbb{N}_{>0}}$ by

$$\begin{aligned} G_1 &:= H^2 \text{ and} \\ G_\ell &:= G_{\ell-1}^2 ⓩ H \text{ for } \ell > 1. \end{aligned}$$

Proposition 6.1.10 ([87])**.** For every $\ell \in \mathbb{N}_{>0}$, the graph G_ℓ is a $(D^{4\ell}, D^2, 1/2)$-graph.

We further require the following lemma.

Lemma 6.1.11. *Let G be a D-regular graph and S be the set of vertices of a connected component of G^2. Then $\lambda(G^2[S]) < 1$.*

Proof of Lemma 6.1.11. Let $1 = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$ be the eigenvalues of $G^2[S]$. By definition we have that $G^2[S]$ is connected and therefore Lemma 6.1.3 implies that $\lambda_1 > \lambda_2$. Now assume that -1 is an eigenvalue of $G^2[S]$ with eigenvector $\overline{v}$. Then the vector $\overline{v}'$ defined by $\overline{v}'_s = \overline{v}_s$ for all $s \in S$ and $\overline{v}'_s = 0$ otherwise is the eigenvector for eigenvalue -1 of the graph G^2. But G^2 can not have a negative eigenvalue as every eigenvalue of G^2 is a square of a real number. Therefore $\lambda_1 \neq \lambda_N$ and $\lambda(G^2[S]) < 1$ as claimed. □

6.1.2 Defining the formula $\varphi_{ⓩ}$

In this section we construct a formula $\varphi_{ⓩ}$, that defines a class of relational structures with binary relations only (edge-coloured graphs) whose underlying undirected graphs are expander graphs, arising from the zig-zag product [115]. The motivation behind this is that a property tester is not able to decide whether a graph is connected or contains a few large connected components. But a property tester deciding whether a graph is a model of this formula has to be able to distinguish them, as being an expander ensures that we have to delete more than an ϵ-fraction of the edges to disconnect the graph into large connected components for some constant $\epsilon > 0$.

We start with a high-level description of the formula. Let $\{G_m\}_{m\in\mathbb{N}_{>0}}$ be as in Definition 6.1.9. Loosely speaking, each model of our formula is a structure which consists of the disjoint union of $G_1, \ldots, G_n$ for some $n \in \mathbb{N}_{>0}$ with some underlying tree structure connecting

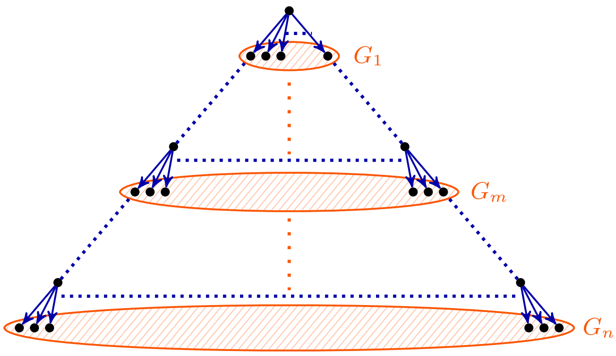


Figure 6.2: Schematic representation of a model of $\varphi_{Ⓩ}$, where the parts in red (grey) only contain relations from E and relations in F are blue (black). Relation R and L are omitted.

G_{m-1} to G_m for all $m \in \{2, \dots, n\}$. For illustration see Figure 6.2. The tree structure enables us to provide an FO-checkable certificate for the expander construction from Definition 6.1.9. The tree structure is a D^4-ary tree, that is used to connect a vertex v of G_{m-1} to every vertex of the copy of H which will replace v in G_m. We use D^4 relations $\{F_k\}_{k \in ([D]^2)^2}$ to enforce an ordering on the D^4 children of each vertex. We use additional relations to encode rotation maps as follows. For $i, j \in [D]^2$ let $E_{i,j}$ be a binary relation. For every pair $i, j \in [D]^2$ we represent an edge e with $f(e) = \{v, w\}$ for two not necessarily distinct vertices v, w in G_m by the two tuples $(v, w) \in E_{i,j}^{\mathcal{A}}$ and $(w, v) \in E_{j,i}^{\mathcal{A}}$. This allows us to encode the relationship $\mathrm{ROT}_{G_m}(v, i) = (w, j)$ in FO using the formula '$E_{i,j}(v, w)$'.

We use auxiliary relations R and L_k for $k \in ([D]^2)^2$, to force the models to be degree-regular. The relation R contains the tuple (r, r) for the root r of the tree, and L_k will contain the tuple (v, v) for every leaf v of the tree.

We now give the precise definition of the formula. Remember that $[n] := \{0, 1, \dots, n-1\}$ for $n \in \mathbb{N}$. Let

$$\sigma := \Big\{ \{E_{i,j}\}_{i,j \in [D]^2}, \{F_k\}_{k \in ([D]^2)^2}, R, \{L_k\}_{k \in ([D]^2)^2} \Big\}, \tag{6.1}$$

where $E_{i,j}$, F_k, R and L_k are binary relation symbols for $i, j \in [D]^2$ and $k \in ([D]^2)^2$. For convenience we introduce auxiliary relations E and F with the property that for every σ-structure we have $E^{\mathcal{A}} := \bigcup_{i,j \in [D]^2} E_{i,j}^{\mathcal{A}}$ and $F^{\mathcal{A}} := \bigcup_{k \in ([D]^2)^2} F_k^{\mathcal{A}}$. In any formula we can reverse using these auxiliary relations by replacing formulas of the form "$E(x, y)$" by "$\bigvee_{i,j \in [D]^2} E_{i,j}(x, y)$" and formulas of the form "$F(x, y)$" by "$\bigvee_{k \in ([D]^2)^2} F_k(x, y)$" below. We use the following formula to identify the root

$$\varphi_{\mathrm{root}}(x) := \forall y \neg F(y, x). \tag{6.2}$$

We now define a formula φ_{tree}, which expresses that any model restricted to the relation F locally looks like a D^4-ary tree. More precisely, the formula defines that the structure has exactly one root, that every vertex apart from the root has exactly one parent and every vertex has either no children or exactly one child for each of the D^4 relations F_k. It also defines the self-loops used to make the structure degree regular. That is the root has an R-self-loop replacing the incoming f-edge and every leaf has D^4 L-self-loops to replace the D^4 outgoing F-edges.

$$\begin{aligned}\varphi_{\text{tree}} := {} & \exists^{=1} x \varphi_{\text{root}}(x) \wedge \forall x \Big(\big(\varphi_{\text{root}}(x) \wedge R(x,x)\big) \vee \big(\exists^{=1} y F(y,x) \wedge \neg\exists y R(x,y) \wedge \neg \exists y R(y,x)\big)\Big) \wedge \\ & \forall x \Bigg(\Big[\neg \exists y F(x,y) \wedge \bigwedge_{k \in ([D]^2)^2} L_k(x,x) \wedge \forall y \big(y \neq x \to \bigwedge_{k \in ([D]^2)^2} \neg L_k(x,y) \wedge \bigwedge_{k \in ([D]^2)^2} \neg L_k(y,x)\big)\Big] \\ & \vee \Big[\neg \exists y \bigvee_{k \in ([D]^2)^2} \big(L_k(x,y) \vee L_k(y,x)\big) \wedge \\ & \bigwedge_{k \in ([D]^2)^2} \exists y_k \Big(x \neq y_k \wedge F_k(x,y_k) \wedge (\bigwedge_{k' \in ([D]^2)^2, k' \neq k} \neg F_{k'}(x,y_k)) \wedge \forall y (y \neq y_k \to \neg F_k(x,y))\Big)\Big]\Bigg).\end{aligned} \tag{6.3}$$

The formula $\varphi_{\text{rotationMap}}$ will define the necessary properties the relations in E need to have in order to encode rotation maps of D^2-regular graphs. For this we make sure that the edge colours encode a map, i.e. for any pair of a vertex x and index $i \in [D]^2$ there is only one pair of vertex y and index $j \in [D]^2$ such that $E_{i,j}(x,y)$ holds and that the map is self inverse, i.e. if $E_{i,j}(x,y)$ then $E_{j,i}(y,x)$.

$$\begin{aligned}\varphi_{\text{rotationMap}} := & \forall x \forall y \Big(\bigwedge_{i,j \in [D]^2} (E_{i,j}(x,y) \to E_{j,i}(y,x))\Big) \wedge \\ & \forall x \Big(\bigwedge_{i \in [D]^2} \Big(\bigvee_{j \in [D]^2} \big(\exists^{=1} y E_{i,j}(x,y) \wedge \bigwedge_{\substack{j' \in [D]^2 \\ j' \neq j}} \neg \exists y E_{i,j'}(x,y)\big)\Big)\Big).\end{aligned} \tag{6.4}$$

We now define a formula φ_{base} which expresses that the root r of the tree has a self-loop (r,r) in each relation $E_{i,j}$ and that the D^2 children of the root form G_1. Let H be the $(D^4, D, 1/4)$-graph from Definition 6.1.9. We assume that H has vertex set $([D]^2)^2$. We then identify vertex $k \in ([D]^2)^2$ with the element $a \in A$ for which $(r,a) \in F_k^{\mathcal{A}}$ for the root r. Let $\mathrm{ROT}_H : ([D]^2)^2 \times [D] \to ([D]^2)^2 \times [D]$ be any rotation map of H. Fixing a rotation map for H fixes the rotation map for H^2. Recall that $G_1 := H^2$. We can define G_1 by a conjunction over all edges of G_1.

$$\begin{aligned}\varphi_{\text{base}} := & \forall x \Big(\varphi_{\text{root}}(x) \to \Big[\bigwedge_{i,j \in [D]^2} \Big(E_{i,j}(x,x) \wedge \forall y \Big(x \neq y \to \big(\neg E_{i,j}(x,y) \wedge \neg E_{i,j}(y,x)\big)\Big)\Big) \wedge \\ & \bigwedge_{\substack{\mathrm{ROT}_{H^2}(k,i) = (k',i') \\ k,k' \in ([D]^2)^2 \\ i,i' \in [D]^2}} \exists y \exists y' \big(F_k(x,y) \wedge F_{k'}(x,y') \wedge E_{i,i'}(y,y')\big)\Big]\Big).\end{aligned} \tag{6.5}$$

We will now define a formula $\varphi_{\text{recursion}}$ which will ensure that level ℓ of the tree contains G_ℓ.

Recall that $G_\ell := G^2_{\ell-1} ⓩ H$. We therefore express that if there is a path of length two between two vertices x, z then for every pair $i, j \in [D]$ there is an edge connecting the corresponding children of x and z according to the definition of the zig-zag product. Here it is important that x and z either both have no children in the underlying tree structure or they both have children. This will also be encoded in the formula.

$$\varphi_{\text{recursion}} := \forall x \forall z \Bigg[\Big(\neg \exists y F(x,y) \wedge \neg \exists y F(z,y) \Big) \vee \bigwedge_{\substack{k'_1, k'_2 \in [D]^2 \\ \ell'_1, \ell'_2 \in [D]^2}} \Big(\exists y \big[E_{k'_1, \ell'_1}(x,y) \wedge E_{k'_2, \ell'_2}(y,z) \big] \to \bigwedge_{\substack{i,j,i',j' \in [D], k, \ell \in ([D]^2)^2 \\ \mathrm{ROT}_H(k,i) = ((k'_1,k'_2),i') \\ \mathrm{ROT}_H((\ell'_2,\ell'_1),j) = (\ell,j')}} \exists x' \exists z' \big[F_k(x,x') \wedge F_\ell(z,z') \wedge E_{(i,j),(j',i')}(x',z') \big] \Big) \Bigg]. \tag{6.6}$$

We finally let

$$\varphi_{ⓩ} := \varphi_{\text{tree}} \wedge \varphi_{\text{rotationMap}} \wedge \varphi_{\text{base}} \wedge \varphi_{\text{recursion}}. \tag{6.7}$$

This concludes defining the formula.

6.1.3 Proving expansion of the property defined by the formula $\varphi_{ⓩ}$

We define the following degree bound

$$d := 2D^2 + D^4 + 1, \tag{6.8}$$

where D is the degree of the base expander H used in the construction from Definition 6.1.9. The degree bound d is chosen in such a way to allow for any element of a σ-structure in C_d to be in $2D^2$ E-relations (G_m is D^2 regular and every edge of G_m is modelled by two directed edges), to have either D^4 F-children or D^4 L-self-loops and to either have one F-parent or be in one R-self-loop. Let $I := \{0\} \sqcup ([D]^2)^2 \sqcup [D]^2$ be an index set. We define the *underlying graph* $U(\mathcal{A})$ of a model $\mathcal{A}$ of $\varphi_{ⓩ}$ to be the undirected graph with vertex set A given by rotation map $\mathrm{ROT}_{U(\mathcal{A})} : A \times I \to A \times I$ defined by

$$\mathrm{ROT}_{U(\mathcal{A})}(v,i) := \begin{cases} (v,0) & \text{if } i = 0 \text{ and } (v,v) \in R^{\mathcal{A}}, \\ (w,j) & \text{if } i = 0 \text{ and } (w,v) \in F_j^{\mathcal{A}}, \\ (w,0) & \text{if } i \in ([D]^2)^2 \text{ and } (v,w) \in F_i^{\mathcal{A}}, \\ (v,i) & \text{if } i \in ([D]^2)^2 \text{ and } (v,v) \in L_i^{\mathcal{A}}, \\ (w,j) & \text{if } i \in [D]^2 \text{ and } (v,w) \in E_{i,j}^{\mathcal{A}}. \end{cases}$$

We can understand this rotation map as labelling the tuples containing an element v as follows: $(v,v) \in R^{\mathcal{A}}$ or $(w,v) \in F_k^{\mathcal{A}}$ respectively will be labelled by 0, $(v,w) \in F_k^{\mathcal{A}}$ or $(v,v) \in L_k^{\mathcal{A}}$

respectively will be labelled by k and $(v, w) \in E_{i,j}^{\mathcal{A}}$ will be labelled by i. Note that $U(\mathcal{A})$ is (D^2+D^4+1)-regular. We chose the notion of an underlying graph here instead of the Gaifman graph as it is more convenient in particular for using results from [115]. However the Gaifman graph can be obtained from the underlying graph by ignoring self-loops and multiple edges. In this section we will show the following.

Proposition 6.1.12. The underlying undirected graphs of models of $\varphi_{ⓩ}$ are a family of ϵ-expander for some $\epsilon > 0$.

We will show Proposition 6.1.12 in several steps analysing the structure of models of $\varphi_{ⓩ}$ in detail. For this let $\mathcal{A}$ be a model of $\varphi_{ⓩ}$. Let $\mathcal{A}|_F := (A, (F_k^{\mathcal{A}})_{k\in([D]^2)^2})$ be the $\{(F_k)_{k\in([D]^2)^2}\}$-structure obtained from $\mathcal{A}$ by forgetting relations $\{E_{i,j}\}_{i,j\in[D]^2}$, R and $\{L_k\}_{k\in([D]^2)^2}$. Recall that we denote the Gaifman graph of $\mathcal{A}|_F$ by $G(\mathcal{A}|_F)$. Let $\mathcal{A}|_E$ be the $\{(E_{i,j})_{i,j\in[D]^2}\}$-structure $(A, (E_{i,j}^{\mathcal{A}})_{i,j\in[D]^2})$ obtained from $\mathcal{A}$ by forgetting relations $\{F_k^{\mathcal{A}}\}_{k\in([D]^2)^2}$, R and $\{L_k\}_{k\in([D]^2)^2}$. We further define the *underlying graph* $U(\mathcal{A}|_E)$ of $\mathcal{A}|_E$ as the undirected graph specified by the rotation map $\mathrm{ROT}_{U(\mathcal{A}|_E)}$ defined by $\mathrm{ROT}_{U(\mathcal{A}|_E)}(v, i) := (w, j)$ if $(v, w) \in E_{i,j}^{\mathcal{A}}$. This is well defined as $\mathcal{A} \models \varphi_{\text{rotationMap}}$.

We use the substructures $G(\mathcal{A}|_F)$ and $U(\mathcal{A}|_E)$ to express the structural properties of models of $\varphi_{ⓩ}$. More precisely we want to prove that $G(\mathcal{A}|_F)$ is a rooted complete tree and $U(\mathcal{A}|_E)$ is the disjoint union of the expanders $G_1, \ldots, G_n$ for some $n \in \mathbb{N}$ (Lemma 6.1.15). To prove this we use two technical lemmas (Lemma 6.1.13 and Lemma 6.1.14). Lemma 6.1.13 intuitively shows that the children in $G(\mathcal{A}|_F)$ of each connected part of $U(\mathcal{A}|_E)$ form the zig-zag product with H of the square of the connected part. Lemma 6.1.14 shows that $G(\mathcal{A}|_F)$ is connected. To prove Proposition 6.1.12 we use that a tree with an expander on each level has good expansion. Loosely speaking, this is true because cutting the tree 'horizontally' takes many edge deletions and for cutting the tree 'vertically' we cut many expanders.

Lemma 6.1.13. *Let $\mathcal{A}$ be a model of $\varphi_{ⓩ}$ and assume S is the set of all vertices belonging to a connected component of $(U(\mathcal{A}|_E))^2$ not containing the root and let $S' := \{w \in A \mid (v, w) \in F^{\mathcal{A}}, v \in S\}$. If $S' \neq \emptyset$ then $U(\mathcal{A}|_E)[S']$ is a connected component of $U(\mathcal{A}|_E)$ and $U(\mathcal{A}|_E)[S'] \cong ((U(\mathcal{A}|_E))^2[S])ⓩ H$.*

We use connected components of $(U(\mathcal{A}|_E))^2$, as the square of a connected component of $U(\mathcal{A}|_E)$ may not be connected, in which case the zig-zag product with H of the square of the connected component cannot be connected.

Proof of Lemma 6.1.13. Assume that $S' \neq \emptyset$. We first show that

$$U(\mathcal{A}|_E)[S'] \cong ((U(\mathcal{A}|_E))^2[S])ⓩ H.$$

For this we use the following two claims.

Claim 1. If $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2[S] \textcircled{z} H}((u,k),(i,j)) = ((w,\ell),(j',i'))$ for some $u,w \in S$, $k,\ell \in ([D]^2)^2$, $i,j,i',j' \in [D]$ then there is $v \in S$ such that $(u,v) \in E^{\mathcal{A}}_{k'_1,\ell'_1}$ and $(v,w) \in E^{\mathcal{A}}_{k'_2,\ell'_2}$ where $\mathrm{ROT}_H(k,i) = ((k'_1,k'_2),i')$ and $\mathrm{ROT}_H((\ell'_2,\ell'_1),j) = (\ell,j')$.

Proof of Claim 1. By assumption we have $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2[S] \textcircled{z} H}((u,k),(i,j)) = ((w,\ell),(j',i'))$. This implies that $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2[S]}(u,(k'_1,k'_2)) = (w,(\ell'_2,\ell'_1))$ for $\mathrm{ROT}_H(k,i) = ((k'_1,k'_2),i')$ and $\mathrm{ROT}_H((\ell'_2,\ell'_1),j) = (\ell,j')$ by definition of the zig-zag product. Since $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2[S]}$ is equal to $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2}$ restricted to elements of the set S, we have that $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2}(u,(k'_1,k'_2)) = (w,(\ell'_2,\ell'_1))$. Then by definition of squaring $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2}(u,(k'_1,k'_2)) = (w,(\ell'_2,\ell'_1))$ implies that there is v such that $\mathrm{ROT}_{U(\mathcal{A}|_E)}(u,k'_1) = (v,\ell'_1)$ and $\mathrm{ROT}_{U(\mathcal{A}|_E)}(v,k'_2) = (w,\ell'_2)$. This implies the claim by the definition of $\mathrm{ROT}_{U(\mathcal{A}|_E)}$. ∎

Claim 2. If $(u,v) \in E^{\mathcal{A}}_{k'_1,\ell'_1}$ and $(v,w) \in E^{\mathcal{A}}_{k'_2,\ell'_2}$ for some $u,v,w \in A$, $k'_1,k'_2,\ell'_1,\ell'_2 \in ([D]^2)^2$ and there is $u' \in A$ with $(u,u') \in F^{\mathcal{A}}$ then there is $w' \in A$ such that $(w,w') \in F^{\mathcal{A}}$. Furthermore for any $i,i',j,j' \in [D]$ there are $\tilde{u},\tilde{w} \in A$, $k,\ell \in ([D]^2)^2$ such that $(\tilde{u},\tilde{w}) \in E^{\mathcal{A}}_{(i,j),(j'i')}$ for $(u,\tilde{u}) \in F^{\mathcal{A}}_k$ and $(w,\tilde{w}) \in F^{\mathcal{A}}_\ell$ where $\mathrm{ROT}_H(k,i) = ((k'_1,k'_2),i')$ and $\mathrm{ROT}_H((\ell'_2,\ell'_1),j) = (\ell,j')$.

Proof of Claim 2. We only use that $\mathcal{A} \models \varphi_{\text{recursion}}$. Since $\varphi_{\text{recursion}}$ has the form $\forall x \forall z \psi(x,z)$ for some formula $\psi(x,z)$ we know that $\mathcal{A} \models \psi(u,w)$. Since $(u,u') \in F^{\mathcal{A}}$ we have

$$\mathcal{A} \not\models \neg\exists y F(u,y) \land \neg\exists y F(w,y).$$

Since additionally

$$\mathcal{A} \models \exists y \big[E_{k'_1,\ell'_1}(u,y) \land E_{k'_2,\ell'_2}(w,z)\big],$$

this implies that

$$\mathcal{A} \models \bigwedge_{\substack{i,j,i',j' \in [D], k,\ell \in ([D]^2)^2 \\ \mathrm{ROT}_H(k,i)=((k'_1,k'_2),i') \\ \mathrm{ROT}_H((\ell'_2,\ell'_1),j)=(\ell,j')}} \exists x' \exists z' \big[F_k(u,x') \land F_\ell(w,z') \land E_{(i,j),(j',i')}(x',z')\big].$$

Since this conjunction is not empty this implies that there exists an element w' such that $(w,w') \in F^{\mathcal{A}}$. More precisely for any $i,i',j,j' \in [D]$ there are $\tilde{u},\tilde{w}$ as stated. ∎

We will argue that for every element $w \in S$ there is a $w' \in S'$ such that $(w,w') \in F^{\mathcal{A}}$. For this pick any $u' \in S'$. Let $u \in S$ be the element such that $(u,u') \in F^{\mathcal{A}}$. By combining Lemma 6.1.11 and Theorem 6.1.8 and Lemma 6.1.3 it follows that $(U(\mathcal{A}|_E)^2[S]) \textcircled{z} H$ is a connected graph. Therefore, there is a path $(u'_0,\dots,u'_m)$ in $(U(\mathcal{A}|_E)^2[S]) \textcircled{z} H$ from $u'_0 = (u,(k_1,k_2))$ to $u'_m = (w,(\ell_1,\ell_2))$ for some $k_1,k_2,\ell_1,\ell_2 \in [D]^2$. By Claim 1 there is a path $(u_0,v_0,u_1,v_1,\dots u_{m-1},v_{m-1},u_m)$ in $U(\mathcal{A}|_E)$ from $u_0 = u$ to $u_m = w$. By inductively using Claim 2 on the path we find w' such that $(w,w') \in F^{\mathcal{A}}$.

Combining this with $\mathcal{A} \models \varphi_{\text{tree}}$ implies that the map $f : S \times ([D]^2)^2 \to S'$, given by $f(v,k) = u$ if $(v,u) \in F^{\mathcal{A}}_k$, is well defined. Furthermore, Claim 1 and Claim 2 imply that if $\mathrm{ROT}_{(U(\mathcal{A}|_E))^2[S] \circledZ H}((u,k),(i,j)) = ((w,\ell),(j',i'))$ then $\mathrm{ROT}_{(U(\mathcal{A}|_E))[S']}(f((u,k)),(i,j)) = (f((w,\ell)),(j',i'))$. This proves that f maps each edge in $((U(\mathcal{A}|_E))^2[S]) \circledZ H$ injectively to an edge in $U(\mathcal{A}|_E)[S']$. Then the map f together with the corresponding edge map is an isomorphism from $((U(\mathcal{A}|_E))^2[S]) \circledZ H$ to $U(\mathcal{A}|_E)$ as both are D^2-regular.

Moreover, $U(\mathcal{A}|_E)[S'] \cong ((U(\mathcal{A}|_E))^2[S]) \circledZ H$ implies that $U(\mathcal{A}|_E)[S']$ is connected and D^2-regular. Since $\mathcal{A} \models \varphi_{\text{rotationMap}}$ enforces that $U(\mathcal{A}|_E)$ is D^2-regular, no vertex $v \in S'$ can have neighbours which are not in S' and therefore $U(\mathcal{A}|_E)[S']$ is a connected component of $U(\mathcal{A}|_E)$. □

Lemma 6.1.14. *Let $\mathcal{A} \in C_d$ be a model of $\varphi_{\circledZ}$. Then $G(\mathcal{A}|_F)$ is connected.*

Proof. Assume that this is false and $G(\mathcal{A}|_F)$ has more than one connected component. Since $\mathcal{A} \models \varphi_{\text{tree}}$ there is exactly one element v such that $\mathcal{A} \models \varphi_{\text{root}}(v)$. Therefore we can pick G' to be a connected component of $G(\mathcal{A}|_F)$ which does not contain v. For the next claim we should have in mind that $(\mathcal{A}|_F)[V(G')]$ can be understood as a directed graph in which every vertex has in-degree 1 and the corresponding undirected graph G' is connected. Hence $(\mathcal{A}|_F)[V(G')]$ must consist of a set of disjoint directed trees whose roots form a directed cycle. Consequently G' has the structure as given in the following claim.

Claim 1. *G' contains a cycle $(c_0,\dots,c_{\ell-1})$ and for every vertex v of G' there is exactly one path $(p_0,\dots,p_m)$ in G' with $p_0 = v$, p_m on the cycle and p_i not on the cycle for all $i \in [m]$.*

Proof of Claim 1. Let v_0 be any vertex in G' and let $S_0 = \{v_0\}$. We will now recursively define v_i to be the vertex of G' such that $(v_i, v_{i-1}) \in F^{\mathcal{A}}$. Such a vertex always exists and is unique by choice of G'. We also let $S_i := S_{i-1} \cup \{v_i\}$. Since A is finite the chain $S_0 \subseteq S_1 \subseteq \dots \subseteq S_i \subseteq \dots$ must become stationary at some point. Let $i \in \mathbb{N}$ be the minimum index such that $S_{i-1} = S_i$ and let $j < i$ be such that $v_i = v_j$. Then $(v_i, v_{i-1}, \dots, v_{j+1}, v_j)$ is a cycle in G' as by construction $(v_k, v_{k-1}) \in F^{\mathcal{A}}$ which implies that $\{v_k, v_{k-1}\}$ is an edge in the Gaifman graph $G(\mathcal{A}|_F)$. Let $C = \{c_0,\dots,c_{\ell-1}\}$ be the vertices of the cycle. Since G' is connected a path such as in the claim always exists. So let us argue that such a path is unique. Assume there are two different such path $(p_0,\dots,p_m)$ and $(p'_0,\dots,p'_{m'})$ and assume that $p_m = c_i$ and $p'_{m'} = c_j$. Let $k \leq \min\{m,m'\}$ be the minimum index such that $p_k \neq p'_k$. Such an index must exist as the paths are different and as $p_0 = p'_0 = v$ we also know that $k \geq 1$. Since $\mathcal{A} \models \varphi_{\text{tree}}$ for every vertex w of G' there can only be one vertex w' of G' such that $(w', w) \in F^{\mathcal{A}}$. As $p_{m-1} \notin C$ and $(c_{(i-1) \bmod \ell}, p_m) \in F^{\mathcal{A}}$ this means that $(p_m, p_{m-1}) \in F^{\mathcal{A}}$. Applying the argument inductively we get that $(p_k, p_{k-1}) \in F^{\mathcal{A}}$. The same argument works for the path $(p'_0,\dots,p'_{m'})$ and therefore $(p'_k, p'_{k-1}) \in F^{\mathcal{A}}$. By the choice of k we know that $p_{k-1} = p'_{k-1}$ and $p_k \neq p'_k$ which contradicts $\mathcal{A} \models \varphi_{\text{tree}}$. ■

Let S_0 be the vertex set of the connected component of $U(\mathcal{A}|_E)$ with $c_0 \in S_0$. Note that S_0 might not be contained in G'.

We now recursively define the infinite sequence of sets $S_i := \{w \in A \mid (v, w) \in F^{\mathcal{A}}, v \in S_{i-1}\}$ for every $i \in \mathbb{N}_{>0}$. Let $m_i := \max_{v \in S_i \cap V} \min_{j \in \{0,\dots,\ell-1\}}\{\mathrm{dist}_{G'}(c_j, v)\}$ and let $v_i \in S_i \cap V$ be a vertex of distance m_i from C in G'. Note here that m_i is well defined as $c_{i \bmod \ell} \in S_i$.

Claim 2. $U(\mathcal{A}|_E)[S_i] = (U(\mathcal{A}|_E)[S_{i-1}])^2 \ⓩ H$.

Proof of Claim 2. We show the stronger statement that $U(\mathcal{A}|_E)[S_i]$ is a connected component of $U(\mathcal{A}|_E)$ and $(U(\mathcal{A}|_E)[S_i])^2 \ⓩ H = U(\mathcal{A}|_E)[S_{i+1}]$ and $\lambda(U(\mathcal{A}|_E)[S_i]) < 1$ for $i \in \mathbb{N}$ by induction.

First observe that $U(\mathcal{A}|_E)[S_0]$ is a connected component of $U(\mathcal{A}|_E)$ by choice of S_0. To argue that $\lambda(U(\mathcal{A}|_E)[S_0]) < 1$ let $\tilde{S} := \{w \in A \mid (w, v) \in F^{\mathcal{A}}, v \in S_0\}$.

We now argue that $(U(\mathcal{A}|_E))^2[\tilde{S}]$ is a connected component of $(U(\mathcal{A}|_E))^2$. Assuming the contrary, either a connected component of $(U(\mathcal{A}|_E))^2$ contains vertices from both $\tilde{S}$ and $A \setminus \tilde{S}$ or $(U(\mathcal{A}|_E))^2[\tilde{S}]$ splits into more than one connected component. Let S' be the vertices of a connected component as in the first case. Then $|S'| > 1$ and hence S' can not contain the root as the root is not in any E-relation with any other elements. Hence by Lemma 6.1.13 we get a connected component of $U(\mathcal{A}|_E)$ on the children of S' containing vertices both from S_0 and from $A \setminus S_0$, which contradicts S_0 being a connected component of $U(\mathcal{A}|_E)$. Now let S' be a connected component as in the second case, and pick S' such that it does not contain the root. Then by Lemma 6.1.13 S_0 must have a non-empty intersection with at least two connected components of $U(\mathcal{A}|_E)$ which is a contradiction.

Thus, by Lemma 6.1.11 we have that $\lambda((U(\mathcal{A}|_E))^2[\tilde{S}]) < 1$. Additionally by Lemma 6.1.13 $U(\mathcal{A}|_E)[S_0] = ((U(\mathcal{A}|_E))^2[\tilde{S}]) \ⓩ H$. Then Theorem 6.1.8 and $\lambda(H) < 1$ ensure that $\lambda(U(\mathcal{A}|_E)[S_0]) < 1$.

For $i > 1$ inductively we assume that $\lambda(U(\mathcal{A}|_E)[S_{i-1}]) < 1$ which implies by Lemma 6.1.6 and Lemma 6.1.3 that $(U(\mathcal{A}|_E)[S_{i-1}])^2$ is a connected component[1] of $(U(\mathcal{A}|_E))^2$ and that $(U(\mathcal{A}|_E))^2[S_{i-1}] = (U(\mathcal{A}|_E[S_{i-1}]))^2$. Since $c_{i \bmod \ell} \in S_i$ by Lemma 6.1.13 we have $U(\mathcal{A}|_E)[S_i]$ is a connected component of $U(\mathcal{A}|_E)$ and $U(\mathcal{A}|_E)[S_i] = (U(\mathcal{A}|_E)[S_{i-1}])^2 \ⓩ H$. Additionally this proves $\lambda(U(\mathcal{A}|_E)[S_i]) < 1$ using Lemma 6.1.6 and Theorem 6.1.8. ■

Claim 3. For every $v \in S_i$ there is $w \in V$ such that $(v, w) \in F^{\mathcal{A}}$.

Proof of Claim 3. By Claim 2 we have that $U(\mathcal{A}|_E)[S_{i+1}] = (U(\mathcal{A}|_E)[S_i])^2 \ⓩ H$. This means that by definition of squaring and the zig-zag product we know that $|S_{i+1}| = D^4 \cdot |S_i|$. But because in addition $\mathcal{A} \models \varphi_{\text{tree}}$ we know that every element $v \in S_i$ will contribute to no more then D^4 elements to S_{i+1}. This means by construction of S_{i+1} that for every element in S_i there must be $w \in V$ such that $(v, w) \in F^{\mathcal{A}}$. ■

[1] We remark that the statement that $(U(\mathcal{A}|_E)[S_{i-1}])^2$ is a connected component does not directly follow from the fact that $U(\mathcal{A}|_E)[S_{i-1}]$ is a connected component of $U(\mathcal{A}|_E)$, as the square of a connected bipartite graph is not necessarily connected.

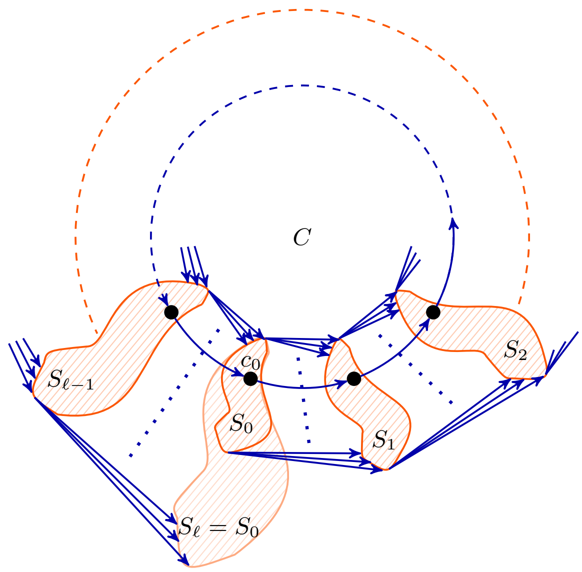


Figure 6.3: Illustration of the proof of Lemma 6.1.14.

Therefore, for every $i \in \mathbb{N}_{>0}$ there is $w_i \in V$ such that $(v_i, w_i) \in F^{\mathcal{A}}$ where v_i is the vertex of distance m_i from C in G' picked above. Let $(u_0, \ldots, u_{m_i})$ be the path in G' from $u_0 = v_i$ to $u_{m_i} \in C$. Note that it is impossible that $w_i = u_1$. This is true as for the path $(u_0, ..., u_{m_i})$, we have that $(u_{j+1}, u_j) \in F^{\mathcal{A}}$ for all $j \in [m_i]$. Furthermore, since $v_i = u_0 \neq u_1$, assuming that $w_i = u_1$ would imply $(v_i, u_1), (u_2, u_1) \in F^{\mathcal{A}}$, which contradicts $\mathcal{A} \models \varphi_{\text{tree}}$. Then $(w_i, u_0, \ldots, u_{m_i})$ is a path in G' from w_i to C. Since $w_i \in S_{i+1}$ by construction, Claim 1 implies that $m_{i+1} \geq m_i + 1$. Therefore $m_i \geq i + m_0$ inductively. But this yields a contradiction, because $\ell + m_0 \leq m_\ell = m_0$ and the length of the cycle $\ell > 0$. See Figure 6.3 for an illustration. Therefore $G(\mathcal{A}|_F)$ must be connected. $\square$

Lemma 6.1.15. *Let $\mathcal{A} \in C_d$ be a (finite) model of $\varphi_{Ⓩ}$. Then $|A| = \sum_{m=0}^{n} D^{4m}$ for some $n \in \mathbb{N}$; $G(\mathcal{A}|_F)$ is a D^4-ary complete rooted tree, where the root is the unique element $r \in A$ for which $\mathcal{A} \models \varphi_{\text{root}}(r)$; and $U(\mathcal{A}|_E)[T_m] \cong G_m$ where G_m is defined as in Definition 6.1.9 and T_m is the set of vertices of distance m to r in the tree $G(\mathcal{A}|_F)$ for any $m \in \{1, \ldots, n\}$. Furthermore, for every $n \in \mathbb{N}$ there is a model of $\varphi_{Ⓩ}$ of size $\sum_{m=0}^{n} D^{4m}$.*

Proof. Lemma 6.1.14 combined with $\mathcal{A} \models \varphi_{\text{tree}}$ proves that $G(\mathcal{A}|_F)$ is a rooted tree. Let n be the greatest distance of any vertex in $G(\mathcal{A}|_E)$ to the root and let T_m be the vertices of distance m to the root for $m \leq n$. Then $U(\mathcal{A}|_E)[T_1] \cong G_1$ because $\mathcal{A} \models \varphi_{\text{base}}$. Since $\lambda(G_m) < 1$ for every $m \in \mathbb{N}_{>0}$ we can use Lemma 6.1.13 to prove by induction that $U(\mathcal{A}|_E)[T_m] \cong G_m$ for every $m \in \{1, \ldots, n\}$. Since G_m has D^{4m} vertices this proves that $\mathcal{A}$ has $\sum_{m=0}^{n} D^{4m}$ vertices. Furthermore, for $n \in \mathbb{N}$ the existence of a model of $\varphi_{Ⓩ}$ of size $\sum_{m=0}^{n} D^{4m}$ is straightforward by the construction of the formula $\varphi_{Ⓩ}$. $\square$

Now we are ready to finish the proof of Proposition 6.1.12.

Proof of Proposition 6.1.12. We will prove that the models of $\varphi_{Ⓩ}$ are a class of ϵ-expanders for $\epsilon := \frac{D^2}{12}$. Let $\mathcal{A}$ be the model of $\varphi_{Ⓩ}$ of size $\sum_{m=0}^{n} D^{4m}$ and $S \subseteq A$ arbitrary with $|S| \leq \sum_{m=0}^{n} \frac{D^{4m}}{2}$. Let T_m be the vertices of distance m to the root of the tree $G(\mathcal{A}|_F)$ and let $S_m := T_m \cap S$.

We can assume that $|S| > 1$ as every vertex has degree at least ϵ. Let us first assume that $|S_m| \leq \frac{D^{4m}}{2}$ for all $m \in [n]$. Then because G_m is an $\frac{D^2}{4}$-expander (this follows directly from Theorem 6.1.4 as $\lambda(G_m) \geq \frac{1}{2}$) and $U(\mathcal{A}|_E)[T_m] \cong G_m$ we know that

$$|\langle S, \overline{S}\rangle_{U(\mathcal{A})}| \geq \sum_{m=1}^{n} \frac{D^2}{4}|S_m| \geq \frac{D^2}{12}\sum_{m=0}^{n}|S_m| = \frac{D^2}{12}|S|.$$

Now assume the opposite and choose m' to be the largest index such that

$$|S_{m'}| > \frac{|T_{m'}|}{2} = \frac{D^{4m'}}{2}. \tag{6.9}$$

We will use the following claim.

Claim 1. $\sum_{m=0}^{\tilde{m}-1}|T_m| \leq \frac{1}{2}|T_{\tilde{m}}|$ for all $\tilde{m} \leq n$.

Proof of Claim 1. Inductively, we argue that

$$\sum_{m=0}^{\tilde{m}-1}|T_m| = \sum_{m=0}^{\tilde{m}-2}|T_m| + |T_{\tilde{m}-1}| \leq \frac{1}{2}(3|T_{\tilde{m}-1}|) \leq \frac{1}{2}|T_{\tilde{m}}|.$$

■

Claim 1 implies that

$$\frac{3}{4}\cdot|T_n| \geq \frac{1}{2}|T_n| + \frac{1}{2}\sum_{m=0}^{n-1}|T_m| = \frac{1}{2}|A| \geq |S| \geq |S_n|.$$

In the case that $m' = n$, this implies

$$|\langle S, \overline{S}\rangle_{U(\mathcal{A})}| \geq \frac{D^2}{4}(|T_n| - |S_n|) \geq \frac{D^2}{16}|T_n| \geq \frac{D^2}{12}\cdot|S|.$$

Assume now that $m' < n$. Since S is the disjoint union of all S_m we know that the set $\langle S, \overline{S}\rangle_{U(\mathcal{A})}$ contains the disjoint sets $\langle S_m, T_m \setminus S_m\rangle_{U(\mathcal{A})}$, $\langle T_{m'} \setminus S_{m'}, T_{m'}\rangle_{U(\mathcal{A})}$ and $\langle S_{m'}, T_{m'+1} \setminus S_{m'+1}\rangle_{U(\mathcal{A})}$ for all $m \in \{m'+1, \ldots, n\}$. For illustration see Figure 6.4. Since every vertex in $T_{m'}$ has D^4 neighbours in $T_{m'+1}$ and on the other hand every vertex in $T_{m'+1}$ has one neighbour in $T_{m'}$ we know that

$$\begin{aligned}|\langle S_{m'}, T_{m'+1} \setminus S_{m'+1}\rangle_{U(\mathcal{A})}| &= |\langle S_{m'}, T_{m'+1}\rangle_{U(\mathcal{A})}| - |\langle S_{m'}, S_{m'+1}\rangle_{U(\mathcal{A})}| \\ &\geq D^4|S_{m'}| - |S_{m'+1}| \\ &\geq D^4\left(|S_{m'}| - \frac{D^{4m'}}{2}\right).\end{aligned}$$

Since additionally

$$\frac{|T_{m'}|}{2} \geq |T_{m'} \setminus S_{m'}| = D^{4m'} - |S_{m'}|$$

and G_m is an $\frac{D^2}{4}$-expander for every m we get

$$\begin{aligned}|\langle S, \overline{S}\rangle_{U(\mathcal{A})}| &\geq \sum_{m>m'} \frac{D^2}{4}|S_m| + \frac{D^2}{4}|T_{m'} \setminus S_{m'}| + D^4\left(|S_{m'}| - \frac{D^{4m'}}{2}\right) \\ &= \frac{D^2}{4}\sum_{m>m'}|S_m| + \frac{D^2}{4}(D^{4m'} - |S_{m'}|) + D^4|S_{m'}| - D^4 \cdot \frac{D^{4m'}}{2} \\ &\qquad + \frac{D^2}{4}|S_{m'}| - \frac{D^2}{4}|S_{m'}| \\ &= \frac{D^2}{4}\sum_{m>m'}|S_m| + \left(D^4 - \frac{D^2}{2}\right)|S_{m'}| - \left(D^4 - \frac{D^2}{2}\right)\frac{D^{4m'}}{2} + \frac{D^2}{4}|S_{m'}| \\ &\overset{\text{Equation 6.9}}{\geq} \frac{D^2}{4}\sum_{m>m'}|S_m| + \left(D^4 - \frac{D^2}{2}\right)\frac{D^{4m'}}{2} - \left(D^4 - \frac{D^2}{2}\right)\frac{D^{4m'}}{2} + \frac{D^2}{4}|S_{m'}| \\ &\overset{\text{Equation 6.9}}{\geq} \frac{D^2}{4}\sum_{m>m'}|S_m| + \frac{D^2}{8}|S_{m'}| + \frac{D^2}{8}\left(\frac{|T_{m'}|}{2}\right) \\ &\overset{\text{Claim 1}}{\geq} \frac{D^2}{4}\sum_{m>m'}|S_m| + \frac{D^2}{8}|S_{m'}| + \frac{D^2}{8}\sum_{m<m'}|T_m| \\ &\overset{|T_m|\geq|S_m|}{\geq} \frac{D^2}{12}|S|.\end{aligned}$$

By choice of ϵ this shows that the models of $\varphi_{Ⓩ}$ are a class of ϵ-expanders. □

Hence we have constructed an FO-sentence $\varphi_{Ⓩ}$ which defines a property of structures, whose underlying undirected graphs are expanders. In the next section we use the expansion property of models of $\varphi_{Ⓩ}$ to prove that the property defined by $\varphi_{Ⓩ}$ is not testable.

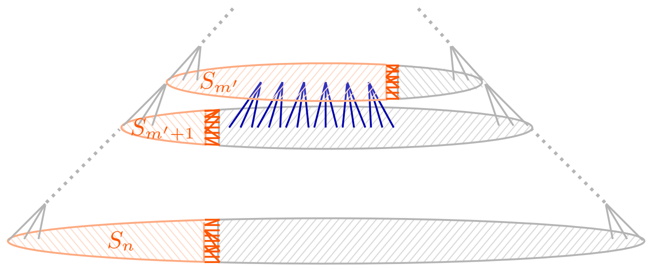


Figure 6.4: Schematic representation of S crossing edges (orange and blue) in the underlying undirected graph in the case of $m' < n$.

6.2 On the non-testability of a Π_2-property

We first prove that the property defined by the sentence $\varphi_{Ⓩ}$ is not testable and then we argue that the sentence $\varphi_{Ⓩ}$ is d-equivalent to a sentence in Π_2 where d is the degree bound defined in Equation 6.8. We let $P_{Ⓩ} := P_{\varphi_{Ⓩ}}$ for the sentence $\varphi_{Ⓩ}$ from Section 6.1.2. We also let σ be the signature from Equation 6.1.

Theorem 6.2.1. *$P_{Ⓩ}$ is not testable on C_d.*

Proof. we prove that $P_{Ⓩ}$ is not repairable and get non-testability of $P_{Ⓩ}$ with Theorem 5.1.2. Let $\epsilon := \frac{1}{144D^2}$ and let $r \in \mathbb{N}$, $\lambda > 0$ and $n_0 \in \mathbb{N}$ be arbitrary. We set $\lambda' := \frac{\lambda}{t \cdot 2^{r+1}}$, where t denotes the number of r-types of bounded degree d, and let n_0' be the positive integer from Theorem 5.2.2 corresponding to λ'. We now pick $n \in \mathbb{N}$ such that $n = \sum_{i=0}^{k} D^{4i}$ for some $k \in \mathbb{N}$, $n \geq 4n_0$ and $n \geq \frac{4n_0'}{\lambda}$. Let $\mathcal{A} \in C_d$ be a model of $\varphi_{Ⓩ}$ on n elements. By Theorem 5.2.2 there is a structure $\mathcal{H} \in C_d$ on $m \leq n_0'$ elements such that the sampling distance of $\mathcal{A}$ and $\mathcal{B}$ (Definition 5.2.1) satisfies $\delta_{\odot}(\mathcal{A}, \mathcal{H}) \leq \lambda'$. Let $\mathcal{B}$ be the structure consisting of $\lfloor \frac{n}{m} \rfloor$ copies of $\mathcal{H}$ and $n \bmod m$ isolated elements (elements not being contained in any tuple). Note that by choice of $\mathcal{B}$ we have that $|A| = |B|$.

We will first argue that $\mathcal{B}$ is in fact ϵ-far from having the property $P_{Ⓩ}$. First we rename the elements from B in such a way that $A = B$ and the number $\sum_{\tilde{R} \in \sigma} |\tilde{R}^{\mathcal{A}} \Delta \tilde{R}^{\mathcal{B}}|$ of tuple modifications to turn $\mathcal{A}$ and $\mathcal{B}$ into the same structure is minimal. Let us pick a partition $A = B = S \sqcup S'$ in such a way that $(S \times S') \cap \tilde{R}^{\mathcal{B}} = \emptyset$, $(S' \times S) \cap \tilde{R}^{\mathcal{B}} = \emptyset$ for any $\tilde{R} \in \sigma$ and $||S| - |S'||$ is minimal among all such partitions. Assume that $|S| \leq |S'|$. Since the connected components of the Gaifman graph $G(\mathcal{B})$ are of size at most m we know that $||S| - |S'|| \leq m$. This is the case as if $||S| - |S'|| > m$ we can get a partition $B = T \sqcup T'$ with $||T| - |T'|| < ||S| - |S'||$ by picking all elements of any connected component of $G(\mathcal{B})$, which is contained in S', and moving these elements from S' to S. Since $|S| \leq |S'|$ and $m \leq \frac{n}{4}$ we know that $\frac{n}{4} \leq |S| \leq \frac{n}{2}$. Since $(S \times S') \cap \tilde{R}^{\mathcal{B}} = \emptyset$ we know that $\mathcal{A}$ and $\mathcal{B}$ must differ in at least all tuples that correspond to an S and S' crossing edge in $U(\mathcal{A})$ i.e. an edge in $\langle S, S'\rangle_{U(\mathcal{A})}$. Hence

$$\begin{aligned}
\sum_{\tilde{R}\in\sigma} |\tilde{R}^{\mathcal{A}} \triangle \tilde{R}^{\mathcal{B}}| &\geq |\langle S, S'\rangle_{U(\mathcal{A})}| \\
&\overset{\text{Def 2.2.1}}{\geq} |S| \cdot h(\mathcal{A}) \\
&\overset{\text{Prop. 6.1.12}}{\geq} \frac{n}{4} \cdot \frac{D^2}{12} \\
&= \frac{1}{48} D^2 n \\
&\geq \frac{1}{144 D^2} dn.
\end{aligned}$$

Therefore $\mathcal{B}$ is ϵ-far from being in $P_{\textcircled{z}}$.

However, the frequency vectors (Definition 2.3.6) of $\mathcal{A}$ and $\mathcal{B}$ are similar as the following shows, proving that $P_{\textcircled{z}}$ is not repairable. Let $\tau_1, \dots, \tau_t$ be a list of all r-types and $T_r := \{\tau_1, \dots, \tau_t\}$. We further denote the σ-structure containing a single element and no tuples by $\mathcal{K}_1$.

$$\begin{aligned}
\|\mathrm{freq}_r(\mathcal{A}) - \mathrm{freq}_r(\mathcal{B})\|_1 &= \sum_{i=1}^{r} |\rho_{\mathcal{A},r}(\{\tau_i\}) - \rho_{\mathcal{B},r}(\{\tau_i\})| \\
&= \sum_{i=1}^{t} \left|\rho_{\mathcal{A},r}(\{\tau_i\}) - \frac{n \bmod m}{n} \cdot \rho_{\mathcal{K}_1,r}(\{\tau_i\}) - \left\lfloor\frac{n}{m}\right\rfloor \cdot \frac{m}{n} \cdot \rho_{\mathcal{H},r}(\{\tau_i\})\right| \\
&\leq \sum_{i=1}^{t} \left|\rho_{\mathcal{A},r}(\{\tau_i\}) - \rho_{\mathcal{H},r}(\{\tau_i\})\right| + \sum_{i=1}^{t} \left|\frac{n \bmod m}{n} \cdot \rho_{\mathcal{K}_1,r}(\{\tau_i\})\right| \\
&\qquad + \sum_{i=1}^{t} \left|\rho_{\mathcal{H},r}(\{\tau_i\}) - \left\lfloor\frac{n}{m}\right\rfloor \cdot \frac{m}{n} \cdot \rho_{\mathcal{H},r}(\{\tau_i\})\right| \\
&\leq \sum_{i=1}^{t} \left|\rho_{\mathcal{A},r}(\{\tau_i\}) - \rho_{\mathcal{H},r}(\{\tau_i\})\right| + \frac{2m}{n} \\
&\leq t \cdot \sup_{X\subseteq T_r} |\rho_{\mathcal{A},r}(X) - \rho_{\mathcal{H},r}(X)| + \frac{2m}{n} \\
&\leq t \cdot 2^r \cdot \delta_{\odot}(\mathcal{A}, \mathcal{H}) + \frac{2m}{n} \\
&\leq \frac{\lambda}{2} + \frac{\lambda}{2} \\
&= \lambda.
\end{aligned}$$

The last inequality holds by choice of λ' and Theorem 5.2.2. $\square$

We now argue that φ_{Z} is d-equivalent to a sentence in Π_2. Let us first observe the following.

Remark 6.2.2. Any Hanf sentence $\exists^{\geq m} x\, \phi_\tau(x)$ is short for

$$\exists x_1 \dots \exists x_m \Big(\bigwedge_{1 \leq i,j \leq m, i \neq j} x_i \neq x_j \wedge \bigwedge_{1 \leq i \leq m} \phi_\tau(x_i) \Big),$$

where $\phi_\tau(x_i)$ is the formula from Lemma 2.4.6. Observe that $\phi_\tau(x_i)$ can be expressed by a formula in Σ_2. Hence any Hanf sentence is in Σ_2.

Lemma 6.2.3. *Let $d \in \mathbb{N}$, σ any signature and let φ be a sentence in* FO$[\sigma]$*. If every model of φ is d-regular, then φ is d-equivalent to a sentence in Π_2.*

The lemma can be equivalently stated by the following syntactic formulation. Let φ^d_{reg} be the FO-sentence expressing that every element has degree d. Then for every FO-sentence φ the sentence $\varphi \wedge \varphi^d$reg is d-equivalent to a sentence in Π_2.

Proof of Lemma 6.2.3. By Theorem 2.4.7 φ is d-equivalent to a sentence φ' in Hanf normal form. We can further assume that φ' is a DNF of Hanf sentences (Lemma 2.4.3). Let therefore

$$\varphi' = \bigvee_{i=1}^{n} \Big(\bigwedge_{j=1}^{\ell_i} \psi_{i,j} \wedge \bigwedge_{j=1}^{k_i} \neg \chi_{i,j} \Big),$$

where $\psi_{i,j}, \chi_{i,j}$ are Hanf sentences. Since every $\psi_{i,j}$ is of the form $\exists^{\geq m} x \phi_\tau(x)$, where τ is an r-type of bounded degree d and $\phi_\tau(x)$ is the formula expressing that x has r-type τ, we can further assume that for every r-ball $(\mathcal{B}, b) \in \tau$ with centre b, we have that $\deg_{\mathcal{B}}(\tilde{b}) = d$ for every element $\tilde{b} \in B$ with $\text{dist}_{\mathcal{B}}(b, \tilde{b}) < r$. This is not a restriction as we assumed that every model of φ is d-regular.

We already know that every Hanf-sentence is in Σ_2 by Remark 6.2.2. This implies, using De Morgan's law, that every negated Hanf-sentence is equivalent to a sentence in Π_2. We can further show the following claim which relies on every model of φ being d-regular.

Claim 1. Every $\psi_{i,j}$ is d-equivalent to a sentence in Π_1.

Proof of Claim 1. Assume that $\psi_{i,j} = \exists^{\geq m} x \phi_\tau(x)$ for some i, j. Let $(\mathcal{B}, b) \in \tau$ be an r-ball with centre b in τ. Let further $B := \{b_0, b_1, \dots, b_k\}$ be the set of elements of $\mathcal{B}$ where $b_0 := b$. We set

$$\begin{aligned}\phi'_\tau(x_0) := \exists x_1, \dots, \exists x_k \Big[\bigwedge_{0 \leq i < j \leq k} (x_i \neq x_j) \wedge \bigwedge_{R \in \sigma} \Big(& \bigwedge_{(b_{i_1}, \dots, b_{i_{\text{ar}(R)}}) \in R^{\mathcal{B}}} R(x_{i_1}, \dots, x_{i_{\text{ar}(R)}}) \\ & \wedge \bigwedge_{(b_{i_1}, \dots, b_{i_{\text{ar}(R)}}) \in B^{\text{ar}(R)} \setminus R^{\mathcal{B}}} \neg R(x_{i_1}, \dots, x_{i_{\text{ar}(R)}}) \Big) \Big],\end{aligned}$$

and $\psi'_{i,j} := \exists^{\geq m} x_0 \phi'_\tau(x_0)$. We now argue that $\psi_{i,j}$ is d-equivalent to $\psi'_{i,j}$. Let therefore $\mathcal{A} \in C_d$ be any σ-structure. By construction of $\phi'_\tau(x_0)$ we directly get that if $\mathcal{A} \models \psi_{i,j}$ then $\mathcal{A} \models \psi'_{i,j}$. Therefore assume $\mathcal{A} \models \psi'_{i,j}$. Then there are m elements $a_1, \dots, a_m \in A$ such that $\mathcal{A} \models \phi'_\tau(a_i)$ for every $i \in \{1, \dots, m\}$. Fix any $i \in \{1, \dots, m\}$. By definition of $\phi'_\tau(x_0)$ there is an injective map $f : \mathcal{B} \to \mathcal{N}_r^{\mathcal{A}}(a_i)$ such that $f(b) = a_i$ and $\big(b_{i_1}, \dots, b_{i_{\mathrm{ar}(R)}}\big) \in R^{\mathcal{B}}$ if and only if $\big(f(b_{i_1}), \dots, f(b_{i_{\mathrm{ar}(R)}})\big) \in R^{\mathcal{A}}$ for any tuple $\big(b_{i_1}, \dots, b_{i_{\mathrm{ar}(R)}}\big) \in B^{\mathrm{ar}(R)}$ and any $R \in \sigma$. We will now argue that f is even an isomorphism from $\mathcal{B}$ to $\mathcal{N}_r^{\mathcal{A}}(a_i)$. For this to be true we have to argue that the image of the map f is precisely $N_r^{\mathcal{A}}(a_i)$. Assume this is false and $a \in N_r^{\mathcal{A}}$ is not in the image of f. Since $a \in N_r^{\mathcal{A}}$ there must be a path $(a_i, p_1, \dots, p_{\ell-1}, a)$ from a_i to a of length $\ell \leq r$ in the Gaifman graph of $\mathcal{A}$. Hence there must be a tuple $\bar{t} \in R^{\mathcal{A}}$ for some relation $R \in \sigma$ which contains both a and $p_{\ell-1}$. Since we assumed that $\deg_{\mathcal{B}}(\tilde{b}) = d$ for every element $\tilde{b} \in B$ with $\mathrm{dist}_{\mathcal{B}}(b, \tilde{b}) < r$ we know that $\deg_{\mathcal{A}[f(B)]}(\tilde{a}) = d$ for any $\tilde{a} \in f(B)$ with $\mathrm{dist}(a_i, \tilde{a}) < r$. Hence $p_{\ell-1} \notin f(B)$ as otherwise $\deg_{\mathcal{A}}(p_{\ell-1}) \geq d+1$ since $a \notin f(B)$. Using this argument inductively proves that $a_i \notin f(B)$ which contradicts the assumption that $f(b) = a_i$. Hence $(\mathcal{N}_r^{\mathcal{A}}, a_i) \in \tau$ and therefore $\mathcal{A} \models \exists^{\geq m} x \phi_\tau(x)$. This proves that $\psi_{i,j}$ is d-equivalent to $\psi'_{i,j}$. Observing that $\psi'_{i,j}$ is indeed in Π_1 concludes the proof. ■

Let $\psi'_{i,j} \in \Pi_1$ be the sentence from Claim 1. Then a straightforward argument shows that φ is d-equivalent to

$$\varphi'' := \bigvee_{i=1}^{n} \Big(\bigwedge_{j=1}^{\ell_i} \psi'_{i,j} \wedge \bigwedge_{j=1}^{k_i} \neg\chi_{i,j} \Big).$$

Let $\exists \overline{x}^{i,j} \tilde{\psi}_{i,j}(\overline{x}^{i,j})$ be a prenex normal form of $\psi'_{i,j}$ where $\tilde{\psi}_{i,j}(\overline{x}^{i,j})$ is a quantifier-free formula and $\overline{x}^{i,j}$ is a tuple of variables. Let further $\forall \overline{y}^{i,j} \exists \overline{z}^{i,j} \tilde{\chi}_{i,j}(\overline{y}^{i,j}, \overline{z}^{i,j})$ be a prenex normal form of $\neg\chi_{i,j}$ where $\tilde{\chi}_{i,j}(\overline{y}^{i,j}, \overline{z}^{i,j})$ is a quantifier-free formula and $\overline{y}^{i,j}, \overline{z}^{i,j}$ are two tuples of variables. Since $\psi_{i,j}$ and $\chi_{i,j}$ are sentences we can assume that the tuples $\overline{x}^{i,j}, \overline{y}^{i,j}, \overline{z}^{i,j}$ contain pairwise different variables. This implies that we can move the quantifiers to the front of the formula as long as $\forall \overline{y}^{i,j}$ appears before $\exists \overline{z}^{i,j}$ in the quantifier prefix for all i, j. Therefore a prenex normal form of φ'' is

$$\forall \overline{y}^{1,1}, \dots, \forall \overline{y}^{1,k_1}, \dots, \forall \overline{y}^{n,1}, \dots, \forall \overline{y}^{n,k_n} \exists \overline{z}^{1,1}, \dots, \exists \overline{z}^{1,k_1}, \dots, \exists \overline{z}^{n,1}, \dots, \exists \overline{z}^{n,k_n}$$
$$\exists \overline{x}^{1,1}, \dots, \exists \overline{x}^{1,\ell_1}, \dots, \exists \overline{x}^{n,1}, \dots, \exists \overline{x}^{n,\ell_n} \bigvee_{i=1}^{n} \Big(\bigwedge_{j=1}^{\ell_i} \tilde{\psi}_{i,j}(\overline{x}^{i,j}) \wedge \bigwedge_{j=1}^{k_i} \tilde{\chi}_{i,j}(\overline{y}^{i,j}, \overline{z}^{i,j}) \Big)$$

proving that $\varphi'' \in \Pi_2$. Since φ is therefore d-equivalent to a formula in Π_2 this proves the claimed statement. □

Theorem 6.2.4. *There is a $d \in \mathbb{N}$ such that there exists a property on C_d definable by a formula in Π_2 that is not testable.*

Proof. Pick $d = 2D^2 + D^4 + 1$ for any large prime power D. Then using the construction from [115] we can find a $(D^4, D, 1/4)$-graph H. By Theorem 6.2.1, using this base expander H for the construction of the formula $\varphi_{\textcircled{z}}$ we get a property which is not testable on C_d. Since all models of $\varphi_{\textcircled{z}}$ are d-regular by construction, Lemma 6.2.3 gives us that $\varphi_{\textcircled{z}}$ is d-equivalent to a formula in Π_2. □

6.3 Extension to simple graphs

The construction of a non-testable FO definable property given in the previous Sections relies on edge colours as a tool for modelling. This raises the question of whether FO definable properties are testable on the class $\mathcal{C}_d$ of simple undirected graphs of bounded degree d. In this Section we give a negative answer to this by interpreting the edge-coloured directed graphs of our previous examples in undirected graphs. We encode σ-structures by representing each type of directed edge by a constant size graph gadget, maintaining the degree regularity. We then translate the formula $\varphi_{\textcircled{z}}$ into a formula $\psi_{\textcircled{z}}$ of which these converted graphs are models. Therefore we obtain a class of simple undirected degree regular expanders, that is defined by an FO-sentence, and obtain the analogous theorem.

Theorem 6.3.1. *There are degree bounds $d \in \mathbb{N}$ such that there exists a property of simple undirected graphs on $\mathcal{C}_d$ definable by a formula in Π_2 that is not testable.*

We now give the construction to convert σ-structures into simple undirected graphs in detail. Let D, d and σ be as defined in Section 6.1.2. We first construct the following arrow-graph gadgets.

Let $G^d(u, v)$ be the simple undirected graph with vertex set

$$\{u, v, u_0, \dots, u_{d-2}\}$$

and edge set

$$\Big\{\{u, u_i\}, \{v, u_i\}, \{u_i, u_j\} \mid i, j \in [d-2], i \neq j\Big\}.$$

Let $H^d(u, v)$ be the simple graph with vertex set

$$\left\{u, v, u_i, u'_j, v_i, v'_j \,\middle|\, i \in \left[\left\lfloor\frac{d-1}{2}\right\rfloor\right], j \in \left[\left\lceil\frac{d-1}{2}\right\rceil\right]\right\}$$

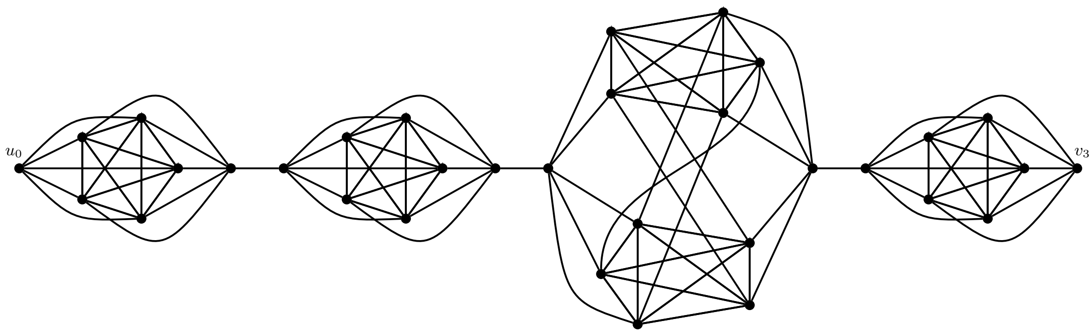


Figure 6.5: Illustration of $P^6_{3,2}(u_0, v_3)$.

and edge set

$$\begin{aligned}
&\Big\{\{u,u_i\},\{v,v_i\},\{u_i,v_i\} \;\Big|\; i \in \Big[\Big\lfloor \frac{d-1}{2} \Big\rfloor\Big]\Big\} \cup \\
&\Big\{\{u,u'_j\},\{v,v'_j\},\{u'_j,v'_j\}\} \;\Big|\; j \in \Big[\Big\lceil \frac{d-1}{2} \Big\rceil\Big]\Big\} \cup \\
&\Big\{\{u_i,u_k\},\{v_i,v_k\} \;\Big|\; i,k \in \Big[\Big\lfloor \frac{d-1}{2} \Big\rfloor\Big], i \neq k\Big\} \cup \\
&\Big\{\{u'_j,u'_k\},\{v'_j,v'_k\} \;\Big|\; j,k \in \Big[\Big\lceil \frac{d-1}{2} \Big\rceil\Big], j \neq k\Big\} \cup \\
&\Big\{\{u_i,v'_j\},\{u'_j,v_i\} \;\Big|\; i \in \Big[\Big\lfloor \frac{d-1}{2} \Big\rfloor\Big], j \in \Big[\Big\lceil \frac{d-1}{2} \Big\rceil\Big]\Big\}.
\end{aligned}$$

Finally, for every $\ell \in \mathbb{N}$ and $0 \leq p \leq \ell$, we let $P^d_{\ell,p}(u_0, v_\ell)$ be the simple undirected graph consisting of ℓ copies $G^d(u_0, v_0), \ldots, G^d(u_{p-1}, v_{p-1})$, $G^d(u_{p+1}, v_{p+1}), \ldots, G^d(u_\ell, v_\ell)$ of $G^d(u, v)$, one copy $H^d(u_p, v_p)$ of $H^d(u, v)$ and additional edges $\{v_i, u_{i+1}\}$ for each $i \in [\ell]$. Note that $P^d_{\ell,p}(u_0, v_\ell)$ has $\ell \cdot (d+1) + 2d$ vertices, the vertices u_0 and v_ℓ have degree $d-1$ and every other vertex has degree d. See Figure 6.5 for an example.

Let $\mathcal{A} \in C_d$ be any σ-structure and let $\ell := 2 \cdot (3D^4 + 1)$. We obtain a simple graph $G_\mathcal{A}$ with bounded degree d out of $\mathcal{A}$ with the following operations.

(E) For every $i_0, i_1, i_2, i_3 \in [D]$ we define $p = \sum_{k=0}^{3} i_k \cdot D^k$ and replace every tuple $(a, b) \in E^\mathcal{A}_{(i_0,i_1),(i_2,i_3)}$ by $P^d_{\ell,p}(u_0, v_\ell)$ and additional edges $\{a, u_0\}$ and $\{v_\ell, b\}$. Here all vertices of $P^d_{\ell,p}(u_0, v_\ell)$ are pairwise distinct and new, and we call them *auxiliary vertices*. Call this gadget graph an $E_{(i_0,i_1),(i_2,i_3)}$*-arrow with end-vertices a and b.*

(F) For every $i_0, i_1, i_2, i_3 \in [D]$ we define $p = D^4 + \sum_{k=0}^{3} i_k \cdot D^k$ and replace every tuple $(a, b) \in F^\mathcal{A}_{((i_0,i_1),(i_2,i_3))}$ by $P^d_{\ell,p}(u_0, v_\ell)$ and additional edges $\{a, u_0\}$ and $\{v_\ell, b\}$. Here all vertices of $P^d_{\ell,p}(u_0, v_\ell)$ are pairwise distinct and new, and we call them *auxiliary vertices*. Call this gadget graph an $F_{((i_0,i_1),(i_2,i_3))}$*-arrow with end-vertices a and b.*

(L) For every $i_0, i_1, i_2, i_3 \in [D]$ we define $p = 2D^4 + \sum_{k=0}^{3} i_k \cdot D^k$ and replace every tuple

$(x, y) \in L^{\mathcal{A}}_{((i_0,i_1),(i_2,i_3))}$ by $P^d_{\ell,p}(u_0, v_\ell)$ and additional edges $\{a, u_0\}$ and $\{v_\ell, b\}$. Here all vertices of $P^d_{\ell,p}(u_0, v_\ell)$ are pairwise distinct and new, and we call them *auxiliary vertices*. Call this gadget graph an $L_{((i_0,i_1),(i_2,i_3))}$*-arrow with end-vertices* a *and* b.

(R) We define $p = 3D^4$ and replace every tuple $(a, b) \in R^{\mathcal{A}}$ by $P^d_{\ell,p}(u_0, v_\ell)$ and additional edges $\{a, u_0\}$ and $\{v_\ell, b\}$. Here all vertices of $P^d_{\ell,p}(u_0, v_\ell)$ are pairwise distinct and new, and we call them *auxiliary vertices*. Call this gadget graph an R*-arrow with end-vertices* a *and* b.

All vertices, that are not *auxiliary*, are called *original vertices*. Note that the location p of the gadget $H^d(v_0, v_\ell)$ uniquely encodes the colour of the original directed coloured edge. Also note that each arrow defined above has a direction as the gadget $H^d(v_0, v_\ell)$ is always located in the first half of the path $P^d_{\ell,p}(u_0, v_\ell)$.

The following is easy to observe from the construction.

Remark 6.3.2. For every vertex x of $G_{\mathcal{A}}$ the following proposition is true. x is an original vertex if and only if x is contained in no triangle.

We now construct the formula $\psi_{Ⓩ}$. For that we let $\delta(x)$ be a formula in the language of undirected graphs σ_{Graph}, saying 'x is an original vertex', which is easy to do by Remark 6.3.2. We further let $\beta(x)$ be a formula saying 'x is an inner vertex of either an $E_{i,j}$-arrow or an F_k-arrow or an L_k-arrow or an R-arrow for any $i, j \in [D]^2$, $k \in ([D]^2)^2$'. Here an inner vertex of an arrow refers to any vertex but the two end vertices. Let $\alpha^E_{i,j}$, α^F_k, α^L_k and α^R be the following σ_{Graph}-formulas. Let $\alpha^E_{i,j}(x, y)$ say 'x and y are the end-vertices of an induced $E_{i,j}$-arrow' for $i, j \in [D]^2$, similarly, let $\alpha^F_k(x, y)$ say 'x and y are the end-vertices of an induced F_k-arrow' for $k \in ([D]^2)^2$. Furthermore let $\alpha^L_k(x, y)$ say 'x and y are the end-vertices of an induced L_k-arrow' for $k \in ([D]^2)^2$ and $\alpha^R(x, y)$ say 'x and y are the end-vertices of an induced R-arrow'. Since the size of all arrow-graph gadgets depends on d the formulas $\beta, \alpha^E_{i,j}$, α^F_k, α^L_k and α^R are straight forward to construct.

Given $\varphi_{Ⓩ}$, formula $\psi_{Ⓩ}$ is obtained as follows. In the formula $\varphi_{Ⓩ}$ we replace each expression $E_{i,j}(x, y)$ by $\alpha^E_{i,j}(x, y)$, each $F_k(x, y)$ by $\alpha^F_k(x, y)$, each $L_k(x, y)$ by $\alpha^L_k(x, y)$ and each $R(x, y)$ by $\alpha^R(x, y)$. In addition, we relativise all quantifiers to the original vertices by replacing every expression of the form $\exists x\, \chi$ by $\exists x\, (\delta(x) \wedge \chi)$ and every expression of the form $\forall x\, \chi$ by $\forall x\, (\delta(x) \rightarrow \chi)$. Let us call the resulting formula $\psi'_{Ⓩ}$. Then we set $\psi_{Ⓩ}$ to be the conjunction of the formula $\psi'_{Ⓩ}$ and the formula $\forall x(\neg\delta(x) \rightarrow \beta(x))$. Let

$$\mathcal{P}_{Ⓩ} := \{G \in \mathcal{C}_d \mid G \models \psi_{Ⓩ}\}.$$

The following is clear by the construction of $G_{\mathcal{A}}$ and $\psi_{Ⓩ}$.

Lemma 6.3.3. *For any $\mathcal{A} \in C_d$ the following proposition is true. $\mathcal{A} \models \varphi_{Ⓩ}$ if and only if $G_{\mathcal{A}} \models \psi_{Ⓩ}$. Additionally we have that if $G \in \mathcal{C}_d$ is a model of $\psi_{Ⓩ}$ then $G \cong G_{\mathcal{A}}$ for some $\mathcal{A} \in C_d$.*

Proof. First assume that $\mathcal{A} \models \varphi_{Ⓩ}$. Then by the construction of $G_{\mathcal{A}}$ and $\psi'_{Ⓩ}$ we get that $G_{\mathcal{A}} \models \psi'_{Ⓩ}$. Since in the construction of $G_{\mathcal{A}}$ no auxiliary vertex is added which is not part of an $E_{i,j}$-arrow or an F_k-arrow or an L_k-arrow or an R-arrow for any $i, j \in [D]^2$, $k \in ([D]^2)^2$ we additionally get that $G_{\mathcal{A}} \models \forall x(\neg\delta(x) \to \beta(x))$.

Let us on the other hand assume that $G \models \psi_{Ⓩ}$. Then $G \models \forall x(\neg\delta(x) \to \beta(x))$. Hence G consists of a set of original vertices that are connected according to $\psi'_{Ⓩ}$ with $E_{i,j}$-arrow or an F_k-arrow or an L_k-arrow or an R-arrow. Hence we can reverse the Operations (E), (F), (L) and (R) to obtain $\mathcal{A}_G$ the corresponding model of $\varphi_{Ⓩ}$ for which $G_{\mathcal{A}_G} = G$. □

In the following, we show that $\mathcal{P}_{Ⓩ}$ is a family of expanders, which allows us to prove non-testability analogously as in the relational structure case. We remark that one could also prove the non-testability of $\mathcal{P}_{Ⓩ}$ by showing that the aforementioned transformation (from σ-structures to simple graphs) is more or less a local reduction that preserves the testability of properties.

Proposition 6.3.4. The models of $\psi_{Ⓩ}$ is a family of ξ-expanders, for some constant $\xi > 0$.

The strategy here is to consider different cases dependent on how the number of original vertices relates to the number of auxiliary vertices contained in some set $S \subseteq V(G)$ of size at most $\frac{V(G)}{2}$ for a model G of $\psi_{Ⓩ}$. Since original vertices are only connected to auxiliary vertices, we get well connectedness of S if the number of auxiliary vertices in comparison to the number of original vertices contained in S is small. On the other hand, using a similar argument we obtain well connectedness of S in the case that the amount of auxiliary vertices is large in comparison to the amount of original vertices in S. In the case that the number of original and auxiliary vertices differs not too much we can use expansion of the structure $\mathcal{A} \in P_{Ⓩ}$ corresponding to G to prove that S is well connected to the rest of G. We give the formal proof in the following.

Proof of Proposition 6.3.4. Let G be a model of $\psi_{Ⓩ}$ and let $\mathcal{A}$ be the corresponding model of $\varphi_{Ⓩ}$ which exists due to Lemma 6.3.3. Let $S \subseteq V(G)$ such that $|S| \leq \frac{|V(G)|}{2}$. Let $V_{\text{original}} \sqcup V_{\text{auxiliary}} = V(G)$ be the partition of $V(G)$ into original and auxiliary vertices. Let $S_{\text{original}} := V_{\text{original}} \cap S$ and $S_{\text{auxiliary}} := V_{\text{auxiliary}} \cap S$.

First note that by the above definitions every tuple in $\mathcal{A}$ corresponds to a constant number $c := 2 \cdot (3D^4 + 1) \cdot (d + 1) + 2d$ of auxiliary vertices in $V_{\text{auxiliary}}$ (each of the copies of gadget $P^d_{\ell,p}(u_0, v_\ell)$ contains $d + 1$ vertices and gadget $H^d(v_0, v_\ell)$ contains $2d$ vertices), where $d = 2D^2 + D^4 + 1$ (see Equation 6.8).

Assume $|S_{\text{original}}| > \frac{2}{dc} \cdot |S|$. Then there are $|S| - |S_{\text{original}}| < \frac{dc-2}{2} \cdot |S_{\text{original}}|$ vertices in $S_{\text{auxiliary}}$. This implies that at least $d \cdot |S_{\text{original}}| - \frac{2}{c} \cdot \frac{dc-2}{2} \cdot |S_{\text{original}}|$ of the arrows incident to a vertex in S_{original} contribute at least one edge to $\langle S, V \setminus S\rangle_G$ and therefore

$$\begin{aligned}\langle S, V \setminus S\rangle_G &\geq d \cdot |S_{\text{original}}| - \frac{dc-2}{c} \cdot |S_{\text{original}}| \\ &= \frac{2}{c} \cdot |S_{\text{original}}| \\ &\geq \frac{4}{dc^2} \cdot |S|.\end{aligned}$$

Assume $\frac{1}{2dc} \cdot |S| < |S_{\text{original}}| \leq \frac{2}{dc} \cdot |S|$. Let $\epsilon = \frac{D^2}{12}$ as defined in the proof of Proposition 6.1.12. Since each edge in the underlying graph $U(\mathcal{A})$ corresponds to exactly one arrow-graph gadget in G we get that $\langle S, V \setminus S\rangle_G \geq \langle S_{\text{original}}, V_{\text{original}} \setminus S_{\text{original}}\rangle_{U(\mathcal{A})}$. Since $\mathcal{A}$ is d-regular and every edge gets replaced by c auxiliary vertices we get $|V(G)| = (1 + \frac{dc}{2}) \cdot |A|$. Hence

$$|S_{\text{original}}| \leq \frac{2}{dc} \cdot |S| \leq \frac{1}{dc} \cdot |V(G)| = \frac{2+dc}{2dc} \cdot |A|$$

and $|A \setminus S_{\text{original}}| \geq \left(\frac{2dc}{2+dc} - 1\right) \cdot |S_{\text{original}}|$. Hence from Proposition 6.1.12 we directly get

$$\begin{aligned}\langle S, V(G) \setminus S\rangle_G &\geq \langle S_{\text{original}}, V_{\text{original}} \setminus S_{\text{original}}\rangle_{U(\mathcal{A})} \\ &= \epsilon \cdot \min\{|S_{\text{original}}|, |A \setminus S_{\text{original}}|\} \\ &\geq \epsilon \cdot \min\left\{\frac{1}{2dc}, \frac{dc}{2+dc}\right\} \cdot |S|.\end{aligned}$$

Now assume $|S_{\text{original}}| \leq \frac{1}{2dc} \cdot |S|$. Therefore there are $|S| - |S_{\text{original}}| \geq |S| - \frac{1}{2dc} \cdot |S|$ vertices in $S_{\text{auxiliary}}$. Of these at least $\frac{2dc-1}{2dc} \cdot |S| - |S_{\text{original}}|dc \geq \frac{dc-1}{2dc} \cdot |S|$ vertices in $S_{\text{auxiliary}}$ that are not in a connected component with any element from S_{original} in the graph $G[S]$. Since any connected component of $G[S]$ with no vertices in S_{original} contains at most c vertices, we get that

$$\langle S, V \setminus S\rangle_G \geq \frac{dc-1}{2dc^2} \cdot |S|.$$

By setting $\xi = \min\{\frac{dc-1}{2dc^2}, \epsilon\frac{1}{2dc}, \epsilon\frac{dc}{2+dc}, \frac{4}{dc^2}\} > 0$ we proved the claimed. $\square$

Now we obtain Theorem 6.3.1 from Proposition 6.3.4 with the same methods used to prove Theorem 6.2.4 from Proposition 6.1.12. Alternatively, we can use a result by Fichtenberger, Sohler and Peng [56] stating that every testable property contains a hyperfinite subproperty. Since a property consisting of expanders cannot contain a hyperfinite subproperty, which is an almost immediate consequence from the definitions of these concepts, this implies Theorem 6.3.1. The result in [56] is however shown for simple undirected graphs and relies on the result by Alon [102] and the canonical tester which we both generalise in Chapter 5. Hence confirming validity for the result from [56] for relational structures seems more lengthy then the approach we use in Section 6.2.

6.4 On the testability of all Σ_2-properties

In this section we let $\sigma = \{R_1, \dots, R_m\}$ be any relational structures and C_d the set of σ-structure of bounded degree d. In this section, we prove the following.

Theorem 6.4.1. *Every property defined by a sentence in Σ_2 is testable in the bounded degree model.*

We adapt the notion of indistinguishability of [6] from the dense model to the bounded degree model for relational structures.

Definition 6.4.2. Two properties $P, Q \subseteq C_d$ are called *indistinguishable* if for every $\epsilon \in (0,1)$ there exists $N = N(\epsilon)$ such that for every structure $\mathcal{A} \in P$ with $|A| > N$ there is a structure $\tilde{\mathcal{A}} \in Q$ with the same universe, that is ϵ-close to $\mathcal{A}$; and for every $\mathcal{B} \in Q$ with $|B| > N$ there is a structure $\tilde{\mathcal{B}} \in P$ with the same universe, that is ϵ-close to $\mathcal{B}$.

The following lemma follows from the definitions, and is similar to [6], though we make use of the canonical testers for bounded degree structures from Section 5.1.

Lemma 6.4.3. *If $P, Q \subseteq C_d$ are indistinguishable properties, then P is testable on C_d if and only if Q is testable on C_d.*

Proof. For any $\epsilon \in (0,1)$ we let $N(\epsilon)$ be the constant from Definition 6.4.2 for P and Q.

First assume that P is testable on C_d and fix an $\epsilon \in (0,1)$. Let T be an $\frac{\epsilon}{2}$-tester for P on C_d with constant query complexity. Repeating T and deciding on the majority of outcomes we can get an $\frac{\epsilon}{2}$-tester T' with error probability $\frac{1}{6}$ and constant query complexity as in Lemma 3.1.7. By Corollary 5.1.4 there is a canonical $\frac{\epsilon}{2}$-tester T'' with error probability $\frac{1}{6}$ and constant query complexity for P on C_d. Therefore there are $s, r \in \mathbb{N}$ such that T'' samples s elements at random from the input structure, explores their r-neighbourhood and makes a deterministic decision on whether to accept the input structure depending only on the distribution of r-neighbourhoods seen.

Let $\epsilon' := \min\{\frac{\epsilon}{2}, \frac{1}{16srd^{r+1}\operatorname{ar}(\sigma)^{r-1}}\}$. Now consider the following algorithm T'''. For any input structure $\mathcal{A} \in C_d$ on $n < N(\epsilon')$ elements we compute precisely whether $\mathcal{A} \in Q$ and answer accordingly. For every input structure on $n \geq N(\epsilon')$ elements we invoke the tester T'' and accept $\mathcal{A}$ if and only if T'' accepts $\mathcal{A}$. The query complexity of this procedure is clearly constant.

Now let us first assume $\mathcal{A} \in Q$ and $|A| \geq N(\epsilon')$. Then by the definition of indistinguishability there is a structure $\tilde{\mathcal{A}} \in P$ which is ϵ'-close to $\mathcal{A}$. Hence $\tilde{\mathcal{A}}$ differs from $\mathcal{A}$ in at most $\epsilon' dn$ tuples. Observe that an r-neighbourhood contains at most

$$d + d^2 \operatorname{ar}(\sigma) + \cdots + d^r \operatorname{ar}(\sigma)^{r-1} \leq rd^r \operatorname{ar}(\sigma)^{r-1}$$

tuples. Hence $\mathcal{A}$ and $\tilde{\mathcal{A}}$ differ in at most $\epsilon' r d^{r+1} \operatorname{ar}(\sigma)^{r-1} n$ neighbourhoods. Therefore with probability

$$1 - s \cdot \frac{\epsilon' r d^{r+1} \operatorname{ar}(\sigma)^{r-1} n}{n} \geq \frac{15}{16}$$

T''' picks s elements from $\mathcal{A}$ such that they have the same r-neighbourhood in $\mathcal{A}$ and $\tilde{\mathcal{A}}$. Here we want to remark that the probability of T''' picking s elements which have the same r-neighbourhood in $\mathcal{A}$ and $\tilde{\mathcal{A}}$ and the probability of T''' accepting $\mathcal{A}$ are not independent. But since T''' picks elements uniformly at random the probability of T''' accepting $\mathcal{A}$ assuming the r-neighbourhoods of the elements sampled are amongst the neighbourhoods on which $\mathcal{A}$ and $\tilde{\mathcal{A}}$ agree is still at least $\frac{5}{6} - \frac{1}{16} \geq \frac{3}{4}$. Hence the probability of T''' both sampling s vertices which have the same r-neighbourhood in $\mathcal{A}$ and $\tilde{\mathcal{A}}$ and accepting is at least $\frac{15}{16} \cdot \frac{3}{4} \geq \frac{2}{3}$. Hence T''' accepts $\tilde{\mathcal{A}}$ with probability at least $\frac{2}{3}$.

Now assume that $\mathcal{A} \in C_d$ with $|A| \geq N(\epsilon')$ is ϵ-far from being in Q. Then $\mathcal{A}$ is at least $\frac{\epsilon}{2}$-far from being in P. Since T'' is an $\frac{\epsilon}{2}$-tester for P, $\mathcal{A}$ must get rejected with probability at least $\frac{5}{6} \geq \frac{2}{3}$. This proves that T''' is an ϵ-tester for Q. □

High-level idea of proof of Theorem 6.4.1. Let $\varphi \in \Sigma_2$. We prove that the property defined by φ can be written as the union of properties, each of which is defined by another formula φ' in Σ_2 where the structure induced by the existentially quantified variables is a fixed structure $\mathcal{M}$ (see Claim 2). With some further simplification of φ', we obtain a formula φ'' in Σ_2 which expresses that the structure has to have $\mathcal{M}$ as an induced substructure and every set of elements of fixed size ℓ has to induce some structure from a set of structures $\mathfrak{H}$, and depending on the structure from $\mathfrak{H}$ a set of ℓ elements induces there might be some connections to the elements of $\mathcal{M}$ (see Claim 3). We now define a formula ψ in Π_1 such that the property defined by ψ is indistinguishable from the property defined by φ'' in the sense that we can transform any structure satisfying ψ, into a structure satisfying φ'' by modifying no more then a small fraction of the tuples and vice versa (see Claim 6). The intuition behind this is that every structure satisfying φ'' can be made to satisfy ψ by removing the structure $\mathcal{M}$ while on the other hand for every structure which satisfies ψ we can plant the structure $\mathcal{M}$ to make it satisfy φ''. Since it is a priori unclear how the existentially and universally quantified variables interact, we have to define ψ very carefully. Here it is important to note that the number of occurrences of structures in $\mathfrak{H}$ forcing an interaction with $\mathcal{M}$ is limited because of the degree bound (see Claim 4). Thus such structures can not be allowed to occur for models of ψ, as here the number of occurrences can not be limited in any way. Since properties defined by a formula in Π_1 are testable, this implies with the indistinguishability of ψ and φ'' that the property defined by φ'' is testable. Furthermore by the fact that testable properties are closed under union by Lemma 3.5.1, we reach the conclusion that any property defined by a formula in Σ_2 is testable.

As described in the proof outline we will not directly give a tester for the property P_φ but decompose φ into simpler cases. However, every simplification of φ used is computable, and therefore (as we do not parametrise by any attribute of the formula) the proof below yields a construction of an ϵ-tester for P_φ for every $\epsilon \in (0,1)$ and every $\varphi \in \Sigma_2$.

For the full proof of Theorem 6.4.1, we use the following definition.

Definition 6.4.4. Let $\mathcal{A}$ be a σ-structure with $A = \{a_1, \dots, a_t\}$. Let $\overline{z} = (z_1, \dots, z_t)$ be a tuple of variables. Then we define $\iota^{\mathcal{A}}(\overline{z})$ as follows.

$$\iota^{\mathcal{A}}(\overline{z}) := \bigwedge_{R \in \sigma} \Bigg(\bigwedge_{\left(a_{i_1}, \dots, a_{i_{\mathrm{ar}(R)}}\right) \in R^{\mathcal{A}}} R\big(z_{i_1}, \dots, z_{i_{\mathrm{ar}(R)}}\big) \wedge \bigwedge_{\left(a_{i_1}, \dots, a_{i_{\mathrm{ar}(R)}}\right) \in A^{\mathrm{ar}(R)} \setminus R^{\mathcal{A}}} \neg R\big(z_{i_1}, \dots, z_{i_{\mathrm{ar}(R)}}\big) \Bigg) \wedge \bigwedge_{\substack{i,j \in [t] \\ i \neq j}} (z_i \neq z_j).$$

Note that for every σ-structure $\mathcal{A}'$ and $\overline{a}' = (a'_1, \dots, a'_t) \in (A')^t$ we have that $\mathcal{A}' \models \iota^{\mathcal{A}}(\overline{a}')$ if and only if $a_i \mapsto a'_i$, $i \in \{1, \dots, t\}$ is an isomorphism from $\mathcal{A}$ to $\mathcal{A}'[\{a'_1, \dots, a'_t\}]$. In particular, if $\mathcal{A}' \models \iota^{\mathcal{A}}(\overline{a}')$, then $\{a'_1, \dots, a'_t\}$ induces a substructure isomorphic to $\mathcal{A}$ in $\mathcal{A}'$.

Proof of Theorem 6.4.1. Let φ be any sentence in Σ_2. Therefore we can assume that φ is of the form $\varphi = \exists \overline{x}\, \forall \overline{y}\, \chi(\overline{x}, \overline{y})$ where $\overline{x} = (x_1, \dots, x_k)$ is a tuple of $k \in \mathbb{N}$ variables, $\overline{y} = (y_1, \dots, y_\ell)$ is a tuple of $\ell \in \mathbb{N}$ variables and $\chi(\overline{x}, \overline{y})$ is a quantifier-free formula. We can further assume that $\chi(\overline{x}, \overline{y})$ is in disjunctive normal form, and that

$$\varphi = \exists \overline{x}\, \forall \overline{y} \bigvee_{i \in I} \Big(\alpha^i(\overline{x}) \wedge \beta^i(\overline{y}) \wedge \mathrm{pos}^i(\overline{x}, \overline{y}) \wedge \mathrm{neg}^i(\overline{x}, \overline{y}) \Big), \tag{6.10}$$

where $\alpha^i(\overline{x})$ is a conjunction of literals only containing variables from $\overline{x}$, $\beta^i(\overline{y})$ is a conjunction of literals only containing variables in $\overline{y}$, $\mathrm{neg}^i(\overline{x}, \overline{y})$ is a conjunction of negated atomic formulas containing both variables from $\overline{x}$ and $\overline{y}$ and $\mathrm{pos}^i(\overline{x}, \overline{y})$ is a conjunction of atomic formulas containing both variables from $\overline{x}$ and $\overline{y}$. Here a literal is either an atomic formula or a negated atomic formula.

We now write the formula φ given in (6.10) as a disjunction over all possible structures in C_d the existentially quantified variables could enforce. Since the elements realising the existentially quantified variables will have a certain structure, it is natural to decompose the formula in this way.

Let $\mathfrak{M} \subseteq C_d$ be a set of models of φ, such that every model $\mathcal{A} \in C_d$ of φ contains an isomorphic copy of some $\mathcal{M} \in \mathfrak{M}$ as an induced substructure, and $\mathfrak{M}$ is minimal with this property.

Claim 1. Every $\mathcal{M} \in \mathfrak{M}$ has at most k elements.

Proof of Claim 1. Towards a contradiction assume there is $\mathcal{M} \in \mathfrak{M}$ with $|M| > k$. Since every structure in $\mathfrak{M}$ is a model of φ there must be a tuple $\overline{a} = (a_1, \dots, a_k) \in M^k$ such that $\mathcal{M} \models \forall \overline{y} \bigvee_{i \in I} \Big(\alpha^i(\overline{a}) \wedge \beta^i(\overline{y}) \wedge \mathrm{pos}^i(\overline{a}, \overline{y}) \wedge \mathrm{neg}^i(\overline{a}, \overline{y})\Big)$. This implies that for every tuple $\overline{b} \in M^\ell$ we have $\mathcal{M} \models \bigvee_{i \in I} \Big(\alpha^i(\overline{a}) \wedge \beta^i(\overline{b}) \wedge \mathrm{pos}^i(\overline{a}, \overline{b}) \wedge \mathrm{neg}^i(\overline{a}, \overline{b})\Big)$. Furthermore, since $\{a_1, \dots, a_k\}^\ell \subseteq M^\ell$ we have that $\mathcal{M}[\{a_1, \dots, a_k\}] \models \forall \overline{y} \bigvee_{i \in I} \Big(\alpha^i(\overline{a}) \wedge \beta^i(\overline{y}) \wedge \mathrm{pos}^i(\overline{a}, \overline{y}) \wedge \mathrm{neg}^i(\overline{a}, \overline{y})\Big)$. This means that $\mathcal{M}[\{a_1, \dots, a_k\}] \models \varphi$. Hence by definition, $\mathfrak{M}$ contains an induced substructure $\mathcal{M}'$ of $\mathcal{M}[\{a_1, \dots, a_k\}]$. Since every model of φ containing $\mathcal{M}$ as an induced substructure must also contain $\mathcal{M}'$ as an induced substructure $\mathfrak{M} \setminus \{\mathcal{M}\}$ is a strictly smaller set than $\mathfrak{M}$ with all desired properties. This contradicts the minimality of $\mathfrak{M}$. ■

Therefore $\mathfrak{M}$ is finite. For $\mathcal{M} \in \mathfrak{M}$ let

$$J_{\mathcal{M}} := \{j \in I \mid \mathcal{M} \models \alpha^j(\overline{m}) \text{ for some } \overline{m} \in M^\ell\} \subseteq I.$$

Claim 2. We have $\varphi \equiv_d \bigvee_{\mathcal{M} \in \mathfrak{M}} \Big(\exists \overline{x} \forall \overline{y} \Big[\iota^{\mathcal{M}}(\overline{x}) \wedge \bigvee_{j \in J_{\mathcal{M}}} \Big(\beta^j(\overline{y}) \wedge \mathrm{pos}^j(\overline{x}, \overline{y}) \wedge \mathrm{neg}^j(\overline{x}, \overline{y})\Big)\Big]\Big)$.

Proof of Claim 2. Let $\mathcal{A} \in C_d$ be a model of φ. Then there is a tuple $\overline{a} = (a_1, \dots, a_k) \in A^k$ such that $\mathcal{A} \models \forall y \chi(\overline{a}, \overline{y})$. Since $\{a_1, \dots, a_k\}^\ell \subseteq A^\ell$ this implies that $\mathcal{A}[\{a_1, \dots, a_k\}] \models \forall y \chi(\overline{a}, \overline{y})$ and hence $\mathcal{A}[\{a_1, \dots, a_k\}] \models \varphi$. In addition, we may assume that we picked $\overline{a}$ in such a way that for any tuple $\overline{a}' = (a'_1, \dots, a'_k) \in \{a_1, \dots, a_k\}^k$ with $\{a'_1, \dots, a'_k\} \subsetneq \{a_1, \dots, a_k\}$ we have that $\mathcal{A} \not\models \forall \overline{y} \chi(\overline{a}', \overline{y})$. (The reason is that if for some tuple $\overline{a}'$ this is not the case then we just replace $\overline{a}$ by $\overline{a}'$ and so on until this property holds). Hence $\mathcal{A}[\{a_1, \dots, a_k\}]$ cannot have a proper induced substructure in $\mathfrak{M}$, and it follows that there is $\mathcal{M} \in \mathfrak{M}$ such that $\mathcal{M} \cong \mathcal{A}[\{a_1, \dots, a_k\}]$. By choice of $J_{\mathcal{M}}$ we get $\mathcal{A} \models \forall \overline{y} \Big[\iota^{\mathcal{M}}(\overline{a}) \wedge \bigvee_{j \in J_{\mathcal{M}}} \Big(\beta^j(\overline{y}) \wedge \mathrm{pos}^j(\overline{a}, \overline{y}) \wedge \mathrm{neg}^j(\overline{a}, \overline{y})\Big)\Big]$ and hence

$$\mathcal{A} \models \bigvee_{\mathcal{M} \in \mathfrak{M}} \Big(\exists \overline{x} \forall \overline{y} \Big[\iota^{\mathcal{M}}(\overline{x}) \wedge \bigvee_{j \in J_{\mathcal{M}}} \Big(\beta^j(\overline{y}) \wedge \mathrm{pos}^j(\overline{x}, \overline{y}) \wedge \mathrm{neg}^j(\overline{x}, \overline{y})\Big)\Big]\Big).$$

To prove the other direction, we now let the structure $\mathcal{A} \in C_d$ be a model of the formula $\bigvee_{\mathcal{M} \in \mathfrak{M}} \Big(\exists \overline{x} \forall \overline{y} \Big[\iota^{\mathcal{M}}(\overline{x}) \wedge \bigvee_{j \in J_{\mathcal{M}}} \Big(\beta^j(\overline{y}) \wedge \mathrm{pos}^j(\overline{x}, \overline{y}) \wedge \mathrm{neg}^j(\overline{x}, \overline{y})\Big)\Big]\Big)$. Consequently there is $\mathcal{M} \in \mathfrak{M}$ and $\overline{a} \in A^k$ such that $\mathcal{A} \models \forall \overline{y} \Big[\iota^{\mathcal{M}}(\overline{a}) \wedge \bigvee_{j \in J_{\mathcal{M}}} \Big(\beta^j(\overline{y}) \wedge \mathrm{pos}^j(\overline{a}, \overline{y}) \wedge \mathrm{neg}^j(\overline{a}, \overline{y})\Big)\Big]$. By choice of $J_{\mathcal{M}}$ this implies $\mathcal{A} \models \forall \overline{y} \bigvee_{j \in J_{\mathcal{M}}} \Big(\alpha^j(\overline{a}) \wedge \beta^j(\overline{y}) \wedge \mathrm{pos}^j(\overline{a}, \overline{y}) \wedge \mathrm{neg}^j(\overline{a}, \overline{y})\Big)$ and hence $\mathcal{A} \models \varphi$. ■

Since the union of finitely many testable properties is testable by Corollary 3.5.2, it is sufficient to show that the property P_φ is testable, where φ is a sentence of the form

$$\varphi = \exists \overline{x} \forall \overline{y} \chi(\overline{x}, \overline{y}), \text{ where } \chi(\overline{x}, \overline{y}) = \Big[\iota^{\mathcal{M}}(\overline{x}) \wedge \bigvee_{j \in J_{\mathcal{M}}} \Big(\beta^j(\overline{y}) \wedge \mathrm{pos}^j(\overline{x}, \overline{y}) \wedge \mathrm{neg}^j(\overline{x}, \overline{y})\Big)\Big], \quad (6.11)$$

for some $\mathcal{M} \in \mathfrak{M}$. In the following, we will enforce that for every conjunctive clause of the big disjunction of χ, the universally quantified variables induce a specific substructure.

For $j \in J_{\mathcal{M}}$ let $\mathfrak{H}_j \subseteq C_d$ be a maximal set of pairwise non-isomorphic structures $\mathcal{H}$ such that $\mathcal{H} \models \beta^j(\overline{b})$ for some $\overline{b} = (b_1, \ldots, b_\ell) \in H^\ell$ with $\{b_1, \ldots, b_\ell\} = H$.

Claim 3. We have $\varphi \equiv_d \exists\overline{x}\forall\overline{y}\Big[\iota^{\mathcal{M}}(\overline{x}) \wedge \bigvee_{\substack{\mathcal{H}\in\mathfrak{H}_j,\\ j\in J_{\mathcal{M}}}} \Big(\iota^{\mathcal{H}}(\overline{y}) \wedge \mathrm{pos}^j(\overline{x},\overline{y}) \wedge \mathrm{neg}^j(\overline{x},\overline{y})\Big)\Big]$.

Proof of Claim 3. Let $\mathcal{A} \in C_d$ and $\overline{a} = (a_1, \ldots, a_k) \in A^k$. First assume that $\mathcal{A} \models \forall\overline{y}\chi(\overline{a},\overline{y})$. Hence for any tuple $\overline{b} \in A^\ell$ there is an index $j \in J_{\mathcal{M}}$ such that $\mathcal{A} \models \beta^j(\overline{b}) \wedge \mathrm{pos}^j(\overline{a},\overline{b}) \wedge \mathrm{neg}^j(\overline{a},\overline{b})$. Then $\mathcal{A} \models \beta^j(\overline{b})$ implies that $\mathcal{A}[\{b_1, \ldots, b_\ell\}] \cong \mathcal{H}$ for some $\mathcal{H} \in \mathfrak{H}_j$. Hence $\mathcal{A} \models \iota^{\mathcal{H}}(\overline{b})$ and $\mathcal{A} \models \Big[\iota^{\mathcal{M}}(\overline{a}) \wedge \bigvee_{\substack{\mathcal{H}\in\mathfrak{H}_j,\\ j\in J_{\mathcal{M}}}} \Big(\iota^{\mathcal{H}}(\overline{b}) \wedge \mathrm{pos}^j(\overline{a},\overline{b}) \wedge \mathrm{neg}^j(\overline{a},\overline{b})\Big)\Big]$.

For the other direction, we let $\mathcal{A} \models \forall\overline{y}\Big[\iota^{\mathcal{M}}(\overline{a}) \wedge \bigvee_{\substack{\mathcal{H}\in\mathfrak{H}_j,\\ j\in J_{\mathcal{M}}}} \Big(\iota^{\mathcal{H}}(\overline{y}) \wedge \mathrm{pos}^j(\overline{a},\overline{y}) \wedge \mathrm{neg}^j(\overline{a},\overline{y})\Big)\Big]$. Then for every tuple $\overline{b} \in A^\ell$ there is an index $j \in J_{\mathcal{M}}$ and $\mathcal{H} \in \mathfrak{H}_j$ such that $\mathcal{H} \models \iota^{\mathcal{H}}(\overline{b}) \wedge \mathrm{pos}^j(\overline{a},\overline{b}) \wedge \mathrm{neg}^j(\overline{a},\overline{b})$. Therefore $\mathcal{A}[\{b_1, \ldots, b_\ell\}] \cong \mathcal{H}$ and we know that $\mathcal{A} \models \beta^j(\overline{b})$. Therefore $\mathcal{A} \models \beta^j(\overline{b}) \wedge \mathrm{pos}^j(\overline{a},\overline{b}) \wedge \mathrm{neg}^j(\overline{a},\overline{b})$ and since this is true for any $\overline{b} \in A^\ell$ we get $\mathcal{A} \models \varphi$. ■

Thus, it suffices to assume that

$$\varphi = \exists\overline{x}\forall\overline{y}\chi(\overline{x},\overline{y}), \text{ where } \chi(\overline{x},\overline{y}) := \Big[\iota^{\mathcal{M}}(\overline{x}) \wedge \bigvee_{\substack{\mathcal{H}\in\mathfrak{H}_j,\\ j\in J_{\mathcal{M}}}} \Big(\iota^{\mathcal{H}}(\overline{y}) \wedge \mathrm{pos}^j(\overline{x},\overline{y}) \wedge \mathrm{neg}^j(\overline{x},\overline{y})\Big)\Big] \quad (6.12)$$

for some $\mathcal{M} \in \mathfrak{M}$.

Next we will define a universally quantified formula ψ and show that P_φ is indistinguishable from the property P_ψ. To do so we will need the two claims below. Intuitively, Claim 4 says that models of φ of bounded degree do not have many 'interactions' between existential and universal variables – only a constant number of tuples in relations combine both types of variables. Note that for a structure $\mathcal{A}$ and tuples $\overline{a} = (a_1, \ldots, a_k) \in A^k$, $\overline{b} = (b_1, \ldots, b_\ell) \in A^\ell$ the condition $\mathcal{A} \models \iota^{\mathcal{H}}(\overline{b}) \wedge \mathrm{pos}^j(\overline{a},\overline{b}) \wedge \mathrm{neg}^j(\overline{a},\overline{b})$ can force an element of $\overline{b}$ to be in a tuple (of a relation of $\mathcal{A}$) with an element of $\overline{a}$, even if $\mathrm{pos}^j(\overline{x},\overline{y})$ is an empty conjunction. For example, it may be the case that for some tuple $\overline{b}' \in \{b_1, \ldots, b_\ell\}^\ell$, every clause $\iota^{\mathcal{H}'}(\overline{y}) \wedge \mathrm{pos}^{j'}(\overline{x},\overline{y}) \wedge \mathrm{neg}^{j'}(\overline{x},\overline{y})$ for which $\mathcal{A} \models \iota^{\mathcal{H}'}(\overline{b}') \wedge \mathrm{pos}^{j'}(\overline{a},\overline{b}') \wedge \mathrm{neg}^{j'}(\overline{a},\overline{b}')$ forces a tuple to contain some element of $\overline{b}'$ and some element of $\overline{a}$. We will now define a set J to pick out the clauses that do not force a tuple to contain both at least one element from $\{a_1, \ldots, a_k\}$ and at least one element from $\{b_1, \ldots, b_\ell\} \setminus \{a_1, \ldots, a_k\}$. Note that we still allow elements from $\overline{b}$ to be amongst the elements in $\overline{a}$. In Claim 4 we show that for every $\mathcal{A} \in C_d$, $\overline{a} \in A^k$ for which $\mathcal{A} \models \forall\overline{y}\chi(\overline{a},\overline{y})$ there are a constant number of tuples $\overline{b} \in A^\ell$ that only satisfy clauses which force a tuple to contain both an element from $\{a_1, \ldots, a_k\}$ and from $\{b_1, \ldots, b_\ell\} \setminus \{a_1, \ldots, a_k\}$.

Let $j \in J_{\mathcal{M}}$, $\mathcal{H} \in \mathfrak{H}_j$ and $\overline{h} = (h_1, \ldots, h_\ell) \in H^\ell$ such that $\mathcal{H} \models \iota^{\mathcal{H}}(\overline{h})$. We define the set

$$P_{j,\mathcal{H}} := \big\{h_i \mid i \in \{1, \ldots, \ell\}, \mathrm{pos}^j(\overline{x},\overline{y}) \text{ does not contain } y_i = x_{i'} \text{ for any } i' \in \{1, \ldots, k\}\big\}.$$

Now we let $J \subseteq J_{\mathcal{M}} \times C_d$ be the set of pairs $(j, \mathcal{H})$, with $\mathcal{H} \in \mathfrak{H}_j$ such that the disjoint union $\mathcal{M} \sqcup \mathcal{H}[P_{j,\mathcal{H}}] \models \varphi$. J now precisely specifies the clauses that can be satisfied by a structure $\mathcal{A}$

and tuple $\overline{a} = (a_1, \ldots, a_k) \in A^k$ and $\overline{b} = (b_1, \ldots, b_\ell) \in A^\ell$ where $\mathcal{A}$ does not contain any tuples both containing elements from $\{a_1, \ldots, a_k\}$ and from $\{b_1, \ldots, b_\ell\} \setminus \{a_1, \ldots, a_k\}$.

Claim 4. Let $\mathcal{A} \in C_d$ and $\overline{a} = (a_1, \ldots, a_k) \in A^k$. If $\mathcal{A} \models \forall \overline{y}\, \chi(\overline{a}, \overline{y})$ then there are at most $k \cdot d$ tuples $\overline{b} \in A^\ell$ such that $\mathcal{A} \not\models \bigvee_{(j,\mathcal{H}) \in J} (\iota^{\mathcal{H}}(\overline{b}) \wedge \mathrm{pos}^j(\overline{a}, \overline{b}) \wedge \mathrm{neg}^j(\overline{a}, \overline{b}))$.

Proof of Claim 4. Assume $\mathcal{A} \models \forall \overline{y}\, \chi(\overline{a}, \overline{y})$. First observe that since $\mathcal{A} \models \forall \overline{y}\, \chi(\overline{a}, \overline{y})$, it holds that $\mathcal{A} \models \forall \overline{y} \bigvee_{\substack{\mathcal{H} \in \mathfrak{H}_j,\\ j \in J_{\mathcal{M}}}} \Big(\iota^{\mathcal{H}}(\overline{y}) \wedge \mathrm{pos}^j(\overline{a}, \overline{y}) \wedge \mathrm{neg}^j(\overline{a}, \overline{y})\Big)$ by Equation(6.12). We now let B be the set $B := \{\overline{b} \in A^\ell \mid \mathcal{A} \not\models \bigvee_{(j,\mathcal{H}) \in J} (\iota^{\mathcal{H}}(\overline{b}) \wedge \mathrm{pos}^j(\overline{a}, \overline{b}) \wedge \mathrm{neg}^j(\overline{a}, \overline{b}))\} \subseteq A^\ell$. Then every $\overline{b} \in B$ adds at least one to $\sum_{i=1}^k \deg_{\mathcal{A}}(a_i)$. Since $\mathcal{A} \in C_d$ implies that $\sum_{i=1}^k \deg_{\mathcal{A}}(a_i) \leq k \cdot d$ we get that $|B| \leq k \cdot d$. ■

Claim 5. Let ψ be a formula of the form $\psi = \forall \overline{z} \chi(\overline{z})$ where $\overline{z} = (z_1, \ldots, z_t)$ is a tuple of variables and $\chi(\overline{z})$ is a quantifier-free formula. Let $\mathcal{A} \in C_d$ with $|A| > d \cdot \mathrm{ar}(\sigma) \cdot t$ and let $b \in A$ be an arbitrary element. Let $\mathcal{A} \models \psi$ and let $\mathcal{A}'$ be obtained from $\mathcal{A}$ by 'isolating' b, i. e. by deleting all tuples containing b from $R^{\mathcal{A}}$ for every $R \in \sigma$. Then $\mathcal{A}' \models \psi$.

Proof of Claim 5. First note that $\mathcal{A}' \models \chi(\overline{a})$ for any tuple $\overline{a} = (a_1, \ldots, a_t) \in (A \setminus \{b\})^t$ as no tuple over the set of elements $\{a_1, \ldots, a_t\}$ has been deleted. Let $\overline{a} = (a_1, \ldots, a_t) \in A^t$ be a tuple containing b. Pick $b' \in A$ such that $\mathrm{dist}_{\mathcal{A}}(a_j, b') > 1$ for every $j \in \{1, \ldots, t\}$. Such an element exists as $|A| > d \cdot \mathrm{ar}(R) \cdot t$. Let $\overline{a}' = (a'_1, \ldots, a'_t)$ be the tuple obtained from $\overline{a}$ by replacing any occurrence of b by b'. Hence $a_j \mapsto a'_j$ defines an isomorphism from $\mathcal{A}'[\{a_1, \ldots, a_t\}]$ to $\mathcal{A}[\{a'_1, \ldots, a'_t\}]$ since b is an isolated element in $\mathcal{A}'[\{a_1, \ldots, a_t\}]$ and b' is an isolated element in $\mathcal{A}[\{a'_1, \ldots, a'_t\}]$. Since $\mathcal{A} \models \chi(\overline{a}')$, it follows that $\mathcal{A}' \models \chi(\overline{a})$. ■

Let $J' \subseteq J$ be the set of pairs $(j, \mathcal{H})$, with $\mathcal{H} \in \mathfrak{H}_j$, for which $\mathrm{pos}^j(\overline{x}, \overline{y})$ is the empty conjunction. J' contains $(j, \mathcal{H})$ for which we want to use $\iota^{\mathcal{H}}(\overline{y})$ to define the formula ψ.

Claim 6. The property P_φ with φ as in (6.12) is indistinguishable from the property P_ψ where $\psi := \forall \overline{y} \bigvee_{(j,\mathcal{H}) \in J'} \iota^{\mathcal{H}}(\overline{y})$.

Proof of Claim 6. Let $\epsilon > 0$ and $N(\epsilon) = N := \frac{k \cdot \ell^2 \cdot d \cdot \mathrm{ar}(R)}{\epsilon}$ and $\mathcal{A} \in C_d$ be any structure with $|A| > N$.

First assume that $\mathcal{A} \models \varphi$. The strategy is to isolate any element b by deleting all tuples containing b which is contained in a tuple $\overline{b} \in A^\ell$ such that $\mathcal{A} \not\models \bigvee_{(j,\mathcal{H}) \in J'} \iota^{\mathcal{H}}(\overline{b})$. This will result in a structure which is ϵ-close to $\mathcal{A}$ and a model of ψ.

Let $\overline{a} \in A^k$ be a tuple such that $\mathcal{A} \models \forall \overline{y} \chi(\overline{a}, \overline{y})$. Let $B \subseteq A^\ell$ be the set of tuples $\overline{b} \in A^\ell$ such that $\mathcal{A} \not\models \bigvee_{(j,\mathcal{H}) \in J} (\iota^{\mathcal{H}}(\overline{b}) \wedge \mathrm{pos}^j(\overline{a}, \overline{b}) \wedge \mathrm{neg}^j(\overline{a}, \overline{b}))$. Then $|B| \leq k \cdot d \cdot \mathrm{ar}(R)$ by Claim 4. Hence the structure $\mathcal{A}'$ obtained from $\mathcal{A}$ by deleting all tuples containing an element of

$$C := \{a_1, \ldots, a_k\} \cup \big\{b \in A \mid \text{ there is } (b_1, \ldots, b_\ell) \in B \text{ such that } b \in \{b_1, \ldots, b_\ell\}\big\}$$

is ϵ-close to $\mathcal{A}$. Since $\mathcal{A} \models \forall \overline{y} \chi(\overline{a}, \overline{y})$ implies $\mathcal{A} \models \forall \overline{y} \bigvee_{\substack{\mathcal{H} \in \mathfrak{H}_j, \\ j \in J_{\mathcal{M}}}} \iota^{\mathcal{H}}(\overline{y})$ by Claim 5 we know that $\mathcal{A}' \models \forall \overline{y} \bigvee_{\substack{\mathcal{H} \in \mathfrak{H}_j, \\ j \in J_{\mathcal{M}}}} \iota^{\mathcal{H}}(\overline{y})$. For any tuple $\overline{b} = (b_1, \ldots, b_\ell) \in (A \setminus C)^\ell$ we have by definition of J' that $\mathcal{A} \models \iota^{\mathcal{H}}(\overline{b})$ for some $(j, \mathcal{H}) \in J'$. Furthermore $\mathcal{A}[\{b_1, \ldots, b_\ell\}] = \mathcal{A}'[\{b_1, \ldots, b_\ell\}]$ and hence $\mathcal{A}' \models \bigvee_{(j,\mathcal{H}) \in J'} \iota^{\mathcal{H}}(\overline{b})$. Let $\overline{b} = (b_1, \ldots, b_\ell) \in A^\ell$ be any tuple containing t elements from C and let $c_1, \ldots, c_t \in C$ be those elements. Pick t elements $c'_1, \ldots, c'_t \in A \setminus C$ such that $\text{dist}_{\mathcal{A}}(a_i, c'_{i'}) > 1$, $\text{dist}_{\mathcal{A}}(c'_{i'}, b_i) > 1$ and $\text{dist}_{\mathcal{A}}(c'_{i'}, c'_i) > 1$ for suitable i, i'. This is possible as $|A| > (k+2\ell) \cdot d \cdot \text{ar}(R)$ which guarantees the existence of $k+2\ell$ elements of pairwise distance 1. Let $\overline{b}' = (b'_1, \ldots, b'_\ell)$ be the vector obtained from $\overline{b}$ by replacing c_i with c'_i. Since $\overline{b}' \in A^\ell$ there must be j', $\mathcal{H}' \in \mathfrak{H}_j$ such that $\mathcal{A} \models \iota^{\mathcal{H}'}(\overline{b}') \wedge \text{pos}^{j'}(\overline{a}, \overline{b}') \wedge \text{neg}^{j'}(\overline{a}, \overline{b}')$. By choice of $c'_1, \ldots, c'_t$ we have that $\text{pos}_{j'}(\overline{x}, \overline{y})$ must be the empty conjunction and hence $(j', \mathcal{H}') \in J'$. Since additionally $b_i \mapsto b'_i$ defines an isomorphism of $\mathcal{A}[\{b'_1, \ldots, b'_\ell\}]$ and $\mathcal{A}'[\{b_1, \ldots, b_\ell\}]$ this implies that $\mathcal{A}' \models \bigvee_{(j,\mathcal{H}) \in J'} \iota^{\mathcal{H}}(\overline{b})$ for all $\overline{b} \in A^\ell$ and hence $\mathcal{A}' \models \psi$.

Now we prove the other direction. Let $\mathcal{A} \models \psi$ with $|A| > N$. The idea here is to plant the structure $\mathcal{M}$ somewhere in $\mathcal{A}$. While this takes less then an ϵ fraction of edge modifications the resulting structure will be a model of φ.

Take any set $B \subseteq A$ of $|M|$ elements. Let $\mathcal{A}'$ be the structure obtained from $\mathcal{A}$ by deleting all edges incident to any element contained in B. Let $\mathcal{A}''$ be the structure obtained from $\mathcal{A}'$ by adding all tuples such that the structure induced by B is isomorphic to $\mathcal{M}$. This takes no more then $2\ell \cdot d \cdot \text{ar}(R) < \epsilon \cdot d \cdot |A|$ edge modifications. Let $\overline{a} \in B^k$ be such that $\mathcal{A} \models \iota^{\mathcal{M}}(\overline{a})$. By Claim 5 we get $\mathcal{A}' \models \psi$. Therefore pick any tuple $\overline{b} = (b_1, \ldots, b_\ell) \in (A \setminus B)^\ell$. Since by construction we have that all b_i's are of distance at least two from $\overline{a}$ we have that $\mathcal{A}'' \models \bigvee_{(j,\mathcal{H}) \in J'} (\iota^{\mathcal{H}}(\overline{b}) \wedge \text{neg}^j(\overline{a}, \overline{b}))$. By choice of $\mathcal{M}$ we also know that $\mathcal{A}'' \models \bigvee_{\substack{\mathcal{H} \in \mathfrak{H}_j, \\ j \in J_{\mathcal{M}}}} \left(\iota^{\mathcal{H}}(\overline{b}) \wedge \text{pos}^j(\overline{a}, \overline{b}) \wedge \text{neg}^j(\overline{a}, \overline{b})\right)$ for all $\overline{b} \in B^\ell$. Therefore pick $\overline{b} = (b_1, \ldots, b_\ell)$ containing both elements from B and from $A \setminus B$. Now pick a tuple $\overline{b}' = (b'_1, \ldots, b'_\ell) \in (A \setminus B)^\ell$ that equals $\overline{b}$ in all positions containing an element from $A \setminus B$. As noted before there is $(j, \mathcal{H}) \in J'$ such that $\mathcal{A}'' \models (\iota^{\mathcal{H}}(\overline{b}') \wedge \text{neg}^j(\overline{a}, \overline{b}'))$. By the definition of J' (and since $J' \subseteq J$) this means that $\mathcal{A}''[\{a_1, \ldots, a_k, b'_1 \ldots b'_\ell\}] \models \varphi$. Since $\overline{b} \in \{a_1, \ldots, a_k, b'_1 \ldots b'_\ell\}^\ell$ this implies $\mathcal{A}''[\{a_1, \ldots, a_k, b'_1 \ldots b'_\ell\}] \models \bigvee_{\substack{\mathcal{H} \in \mathfrak{H}_j, \\ j \in J_{\mathcal{M}}}} \left(\iota^{\mathcal{H}}(\overline{b}) \wedge \text{pos}^j(\overline{a}, \overline{b}) \wedge \text{neg}^j(\overline{a}, \overline{b})\right)$. Then $\mathcal{A}'' \models \bigvee_{\substack{\mathcal{H} \in \mathfrak{H}_j, \\ j \in J_{\mathcal{M}}}} \left(\iota^{\mathcal{H}}(\overline{b}) \wedge \text{pos}^j(\overline{a}, \overline{b}) \wedge \text{neg}^j(\overline{a}, \overline{b})\right)$ and hence $\mathcal{A}'' \models \varphi$. ■

Since $\psi \in \Pi_1$ we have that P_ψ is testable, and hence P_φ is testable by Claim 6. □

6.5 Summary

In this Chapter we studied testability of FO in the bounded degree model according to prefix classes of FO. We obtain a classification result of the form that every property defined by a sentence in Σ_2 is testable while there is a sentence in Π_2 which defines a non-testable property. We caution that we do not prove that every sentence in Π_2 defines a property which is not testable and in fact we show in Chapter 7 that this is not true. Our result gives a bounded degree equivalent to the study of FO definable properties in the dense model [6] and answers an open question from [2]. At the core of our classification result lies a construction of an FO sentence defining a class of expanders. We think that this construction is of independent interest.

Chapter 7

An alternative approach: testing properties of neighbourhoods

In this section we take an alternative approach to testability of properties definable in FO. While in Chapter 6 we identified the maximal prefix class of testable FO definable properties (i. e. Σ_2), in this section we are interested in identifying further fragments of FO that yield testable properties. Motivated by Hanf normal form, we study testability of negated Hanf sentences. Note that a Hanf sentence postulates the existence of a fixed number of vertices with a particular neighbourhood type. For large enough graph such neighbourhood types can be edited into any graph using at most a linear fraction of edge modifications. Therefore we can test whether a graph satisfies a given Hanf sentences by precisely determining the answer for small graphs and accepting any large graph. We can further prove testability for negated Hanf sentences under some restriction.

Furthermore, we study the property of neighbourhood regularity which is FO definable. This can be seen as a generalisation of degree regularity which is a testable property [69]. We prove testability for neighbourhood regularity for some restricted 1-types.

Here we should also remark that both these classes of properties (i. e. properties defined by negated Hanf sentences and neighbourhood regularity), even considering the restriction under which we can prove testability, are not in general definable by formulas in Σ_2. Hence the results obtained in this section are not covered by Theorem 6.4.1.

7.1 Neighbourhood freeness and neighbourhood regularity

In this section we only consider simple, undirected graphs. Let $d \in \mathbb{N}$ be a degree bound and $\mathcal{C}_d$ the class of simple graphs of bounded degree d. For any $r \in \mathbb{N}$ we let T_r be the set of all

r-types of bounded degree d.

Definition 7.1.1. Let $r \geq 1$ and let $\tau \in T_r$ be an r-type and let $\phi_\tau(x)$ be an FO formula expressing that x has r-type τ (see Lemma 2.4.6).

- We say that a graph $G \in \mathcal{C}_d$ is *τ-neighbourhood regular*, if $G \models \forall x \phi_\tau(x)$.
- We say that a graph $G \in \mathcal{C}_d$ is *τ-neighbourhood free*, if $G \models \neg\exists x \phi_\tau(x)$.
- If $\mathcal{F} \subseteq T_r$ we say that $G \in \mathcal{C}_d$ is $\mathcal{F}$-free, if G is τ-neighbourhood free for all $\tau \in \mathcal{F}$.

We prove the following theorems.

Theorem 7.1.2. *Let τ be an r-type, where $r \geq 1$. If $B \subseteq \mathcal{C}_{d'}$ for all $(B, b) \in \tau$ and $d' < d$ then τ-neighbourhood freeness is uniformly testable on the class $\mathcal{C}_d$ with constant running time.*

Theorem 7.1.3. *For every 1-type τ, τ-neighbourhood freeness is uniformly testable on the class $\mathcal{C}_d$ with constant running time.*

Theorem 7.1.4. *Let τ be a 1-type such that $B \setminus \{b\}$ is a union of disjoint cliques for all $(B, b) \in \tau$ then τ-neighbourhood regularity is uniformly testable on $\mathcal{C}_d$ in constant time.*

Recall that a *clique* in a graph G is a subset $W \subseteq V(G)$ such that $\{w, w'\} \in E(G)$ for every pair of distinct vertices $w, w' \in W$.

Example 7.1.5. Let us look at an example for a property whose testability follows from Theorem 7.1.4. For this let B be the graph

$$B = \Big(\{v_1, v_2, v_3, v_4\}, \big\{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_3, v_4\}\big\}\Big)$$

and τ be the 1-type for which $(B, v_1) \in \tau$. Since $B \setminus \{v_1\}$ is the disjoint union of a K_1 and a K_2, the property of being τ-neighbourhood regular is testable by Theorem 7.1.4. Furthermore, an example of a graph having this property can be constructed by taking a hexagonal grid, embedded on a torus and then taking the replacement product with a triangle (see Figure 7.1).

7.2 Prefix classes of neighbourhood regularity and neighbourhood freeness

First observe that both τ-neighbourhood freeness and τ-neighbourhood regularity can be defined by formulas in Π_2 for any neighbourhood type τ. This can be easily argued considering that $\neg\phi_{\tau'}(x) \in \Pi_2$ for any $\tau' \in T_r$ (see Lemma 2.4.6) and $\forall x \phi_\tau(x) \equiv \forall x \bigwedge_{\tau' \in T_r \setminus \{\tau\}} \neg\phi_{\tau'}$ and $\neg\exists x \phi_\tau(x) \equiv \forall x \neg\phi_\tau(x)$. Therefore the next Lemma shows that there exist neighbourhood properties that are in Π_2, but not in Σ_2.

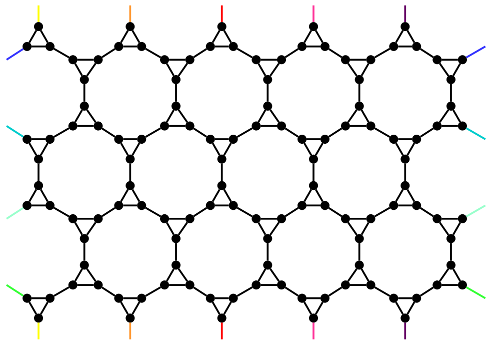


Figure 7.1: Example of a neighbourhood regular graph.

Lemma 7.2.1. *There exist* 1*-types* $\tau, \tau' \in T_r$ *such that neither* τ*-neighbourhood freeness nor* τ'*-neighbourhood regularity can be defined by a formula in* Σ_2*.*

Proof. For $n \in \mathbb{N}$, let C_n be the cycle on vertex set $[n] := \{0, 1, \dots, n-1\}$ and P_n the path on vertex set $[n]$. We first show the following claim.

Claim 1. Let $\varphi = \exists \overline{x} \forall \overline{y}\ \chi(\overline{x}, \overline{y})$ where $\overline{x} = (x_1, \dots, x_k)$, $\overline{y} = (y_1, \dots, y_\ell)$ are tuples of variables and $\chi(\overline{x}, \overline{y})$ is a quantifier-free formula. If $C_n \models \varphi$ then $P_{n-1} \models \varphi$ for any $n > k$.

Proof of Claim 1. Assume that on the contrary for some $n > k$, it holds that $C_n \models \varphi$, while $P_{n-1} \not\models \varphi$. Since $C_n \models \varphi$ there are k vertices $v_1, \dots, v_k$ in C_n such that $C_n \models \forall \overline{y}\ \chi((v_1, \dots, v_k), \overline{y})$. Since $n > k$, there exists at least one vertex $i \in [n]$ that is not amongst $v_1, \dots, v_k$. For $j \in \{1, \dots, k\}$ let $v'_j := (v_j + n - 1 - i) \bmod n$. Since $P_{n-1} \not\models \varphi$ and $v'_j \in [n-1]$, we have that $P_{n-1} \not\models \forall \overline{y}\ \chi((v'_1 \dots, v'_k), \overline{y})$. Hence there must be vertices $w'_1, \dots, w'_\ell$ in P_{n-1} such that $P_{n-1} \not\models \chi((v'_1, \dots, v'_k), (w'_1, \dots, w'_\ell))$. Now we let $w_j := (w'_j + i + 1) \bmod n$. Then $v_j \mapsto v'_j$ and $w_j \mapsto w'_j$ defines an isomorphism from $C_n[\{v_1, \dots, v_k, w_1, \dots, w_\ell\}]$ and $P_{n-1}[\{v'_1, \dots, v'_k, w'_1, \dots, w'_\ell\}]$. Hence $C_n \not\models \chi((v_1, \dots, v_k), (w_1, \dots, w_\ell))$ which contradicts that $C_n \models \varphi$. ■

Now we let τ be the 1-neighbourhood type saying that the center vertex x has exactly one neighbour. Let τ' be the 1-neighbourhood type saying that the center vertex has two non-adjacent neighbours. Since C_n is τ-neighbourhood free and τ'-neighbourhood regular, while P_{n-1} is neither, the statement of the lemma follows from Claim 1. □

Note that the above lemma implies that we cannot simply invoke the testers for testing Σ_2 properties from Theorem 6.4.1 to prove Theorem 7.1.2, Theorem 7.1.3 and Theorem 7.1.4.

7.3 Proving testability of neighbourhood freeness and neighbourhood regularity with restrictions

The testers for the properties of Theorem 7.1.2, Theorem 7.1.3 and Theorem 7.1.4 proceed in the same way. Furthermore, the correctness of these testers depends on some removal property. This removal property essentially says that a few "forbidden" neighbourhoods can be removed from a graph by a sub-linear number of edge modifications. We will first describe in general how such a tester works and provide the analysis of correctness assuming this removal property holds (Lemma 7.3.2). We will then prove so called removal lemmas (Lemma 7.3.4, Lemma 7.3.5, Lemma 7.3.6 and Claim 2) which prove that the removal property holds in all three cases. To describe the tester we use the following algorithms from [112] which estimates the neighbourhood-frequency vector of a graph. Recall that for fixed $r \in \mathbb{N}$ we denote the frequency vector of a graph $G \in \mathcal{C}_d$ defined in Definition 2.3.6 by $\mathrm{freq}_r(G)$.

Algorithm 2: $\textsc{EstimateFrequencies}_{r,s}$

Query access: $G \in \mathcal{C}_d$
Input : $n := |V(G)|$
Output : A vector $\widetilde{\mathrm{freq}}$

```
freq~ = (0, ..., 0);
Sample s Elements u_1, ..., u_s ∈ V(G) uniformly at random;
for i = 1, ..., s do
    Explore the r-neighbourhood of u_i;
    for j = 1, ..., t do
        if (N_r^G(u_i), u_i) ∈ τ_j then
            freq~_j = freq~_j + 1/s;
        end
    end
end
```

where $\tau_1, \dots, \tau_t$ is a list of all r-types of bounded degree d.

In the algorithm we mean by exploring the r-neighbourhood of a vertex $v \in V(G)$ that the algorithm performs a breadth-first search from v up to depth r using query access to G. For any $G \in \mathcal{C}_d$ we write $\textsc{EstimateFrequencies}_{r,s}(G)$ to express that we run the algorithm $\textsc{EstimateFrequencies}_{r,s}$ given query access to G with input $n = |V(G)|$. It is easy to observe that the query complexity of $\textsc{EstimateFrequencies}_{r,s}(G)$ depends only on r and s and is independent of $|V(G)|$. Furthermore, the following lemma [112, Lemma 5.1] states that the vector returned by $\textsc{EstimateFrequencies}_{r,s}(G)$ is a good approximation of $\mathrm{freq}_r(G)$.

Lemma 7.3.1 (Lemma 5.1 in [112]). *Let $r, s \in \mathbb{N}$ and $\lambda \in (0, 1]$. Let G be a graph of bounded* *degree d. If $s \geq \frac{t^2}{2\lambda^2} \cdot \ln(t+40)$ then* $\mathrm{EstimateFrequencies}_{r,s}(G)$ *returns a frequency vector* $\widetilde{\mathrm{freq}}$ *satisfying the following property with probability at least* $\frac{19}{20}$

$$\|\widetilde{\mathrm{freq}} - \mathrm{freq}_r(G)\|_1 \leq \lambda.$$

Now we can prove the following lemma which provides a framework that will be used in Theorem 7.1.2, Theorem 7.1.3 and Theorem 7.1.4.

Lemma 7.3.2. *Let $\mathcal{F} \subseteq T_r$ and let $\mathcal{P} \subseteq \mathcal{C}_d$ be the set of all graphs that are $\mathcal{F}$-free. Let $M \subseteq \mathbb{N}$ be a decidable set such that $G \in \mathcal{P}$ implies that $|V(G)| \notin M$. Let $f_M : \mathbb{N} \to \mathbb{N}$ be a function such that M can be decided in time f_M. Assume for every $\epsilon \in (0,1]$ there exist $\lambda := \lambda(\epsilon) \in (0,1]$ and $n_0 := n_0(r,\epsilon) \in \mathbb{N}$ such that every graph $G \in \mathcal{C}_d$ on $n \geq n_0$, $n \notin M$ vertices, which is ϵ-far from $\mathcal{P}$, contains more than λn elements v with $(\mathcal{N}_r^G(v), v) \in \tau \in \mathcal{F}$. $\mathcal{P}$ is uniformly testable on $\mathcal{C}_d$ in time $\mathcal{O}(f_M)$.*

Proof. Let $\epsilon \in (0,1]$ be fixed and let $\lambda := \lambda(\epsilon)$ and $n_0 := n_0(r,\epsilon) \in \mathbb{N}$. Furthermore, let $s = (t^2/\lambda^2)\ln(t+40)$. Consider the following probabilistic algorithm T, which is given query access to a graph $G \in \mathcal{C}_d$ and gets the number of vertices n as input. In the algorithm $\tau_1, \ldots \tau_t$ is a list of all r-types of bounded degree d.

Algorithm 3: Tester framework

Query access: $G \in \mathcal{C}_d$
Input : $n := |V(G)|$

Reject if $n \in M$;
if $n < n_0$ **then**
  Query the entire graph G and decide exactly if $G \in \mathcal{P}$;
else
  Run $\textsc{EstimateFrequencies}_{r,s}(G)$ to get $\widetilde{\mathrm{freq}}$ satisfying $\|\widetilde{\mathrm{freq}} - \mathrm{freq}_r(G)\|_1 \leq \lambda$ with probability at least 19/20;
  Reject G if $\sum_{\tau_i \in \mathcal{F}} \widetilde{\mathrm{freq}}_i > 0$;
  Accept otherwise;
end

Here by querying an entire graph G we mean that we make queries (v,i) for every $v \in V(G)$ and every $i \in \{1, \ldots, d\}$.

The query complexity of T is clearly constant, since s and n_0 are constant and the number of vertices in any r-neighbourhood is bounded by $(4d)^r$ for graphs in $\mathcal{C}_d$ by Lemma 2.3.3. The running time of the first step is $f_M(n)$ and for the other steps it is constant.

To prove that T is an ϵ-tester, first assume that $G \in \mathcal{P}$. Then $n \notin M$ and $(\mathcal{N}_r^G(v), v) \in \tau \notin \mathcal{F}$ for all vertices v . Hence $\sum_{\tau_i \in \mathcal{F}} \widetilde{\mathrm{freq}}_i = 0$ and T accepts G.

Now consider that G is ϵ-far from $\mathcal{P}$. If $n \in M$ then G is rejected in the first step. Hence let $n \notin M$ and assume $\|\widetilde{\text{freq}} - \text{freq}_r(G)\|_1 \leq \lambda$, which occurs with probability at least $19/20 \geq 2/3$. Then

$$\begin{aligned}
\sum_{\tau_i \in \mathcal{F}} \widetilde{\text{freq}}_i &= \sum_{\tau_i \in \mathcal{F}} (\text{freq}_r(G))_i - \sum_{\tau_i \in \mathcal{F}} \big((\text{freq}_r(G))_i - \widetilde{\text{freq}}_i\big) \\
&> \lambda - \Big| \sum_{\tau_i \in \mathcal{F}} \big((\text{freq}_r(G))_i - \widetilde{\text{freq}}_i\big) \Big| \\
&\geq \lambda - \sum_{\tau_i \in \mathcal{F}} \big| \big((\text{freq}_r(G))_i - \widetilde{\text{freq}}_i\big) \big| \\
&\geq 0,
\end{aligned}$$

where the first inequality holds by the assumption that in graphs that are ϵ-far from $\mathcal{P}$ there are more than λn vertices of r-type in $\mathcal{F}$ made in Lemma 7.3.2. Hence T rejects G. $\square$

To illustrate the use of the set M in Lemma 7.3.2 let us consider the following example.

Example 7.3.3. Let $\mathcal{P}$ be the property of being K_4-neighbourhood regular. Let G_m be the graph consisting of m disjoint copies of K_4 and one isolated vertex. First note that G_m contains $4m + 1$ vertices. Being K_4-regular implies that every vertex has degree 3. But because every graph contains an even number of vertices of odd degree, G_m cannot be made K_4-neighbourhood regular by edge modifications. Therefore G_m is ϵ-far from $\mathcal{P}$. But for $m \to \infty$ the probability of sampling the isolated vertex in G_m tends to 0 meaning that with high probability the tester with $M = \emptyset$ will accept G_m. We will show in Theorem 7.1.4 that $\mathcal{P}$ is testable if we set $M = N \setminus \{4m \mid m \in N\}$.

Lemma 7.3.4. *For $r \geq 1$, let τ be an r-type and $(B, b) \in \tau$. Let $\tilde{d} < d$, $d \neq 1$ be integers and assume that $\mathcal{N}^B_{r-1}(b)$ contains a vertex a with $\deg_B(a) = \tilde{d}$ and that $\deg_B(v) \neq \tilde{d} + 1$ for all vertices v in $\mathcal{N}^B_{r-1}(b)$. Let $\epsilon \in (0, 1]$ be fixed, $n_0 = 2d^2/\epsilon$ and $\lambda = \epsilon d/(14(4d)^{2r})$. Every graph $G \in \mathcal{C}_d$ on $n \geq n_0$ vertices which is ϵ-far from being τ-neighbourhood free contains more than λn vertices of r-type τ.*

Proof. We proceed by contraposition. Assume $G \in \mathcal{C}_d$ is a graph on $n \geq n_0$ vertices containing no more than λn vertices v of r-type τ.

Case $\tilde{d} = 0$, $d > 1$. Then we add one edge to every pair of vertices of degree 0. If there is only one vertex v of degree 0 left, we add an edge from v to any other vertex of degree $< d$. If there is no such vertex then there must be vertex u contained in two edges and we replace one edge $\{u, w\}$ by $\{v, w\}$. That way we obtain G' which is $2\lambda n \leq \epsilon dn$ close to G.

Case $\tilde{d} = 1$. We add edges between pairs of degree 1 vertices. If there are two left, connected by an edge, we delete that edge. If there is only one vertex v of degree 1 left, then there is another vertex u of odd degree. By removing an edge $\{u, w\}$ and adding $\{v, w\}$ we get that $\deg_G(v), \deg_G(w) > 1$. We obtain G' which is $2\lambda n \leq \epsilon dn$ close to G.

Case $\tilde{d} \geq 2$. Let us pick a set $\{v_1, \dots, v_k\}$ of $k \leq \lambda n$ vertices of degree $\tilde{d}$ such that for every vertex v of r-type τ there is an index $1 \leq i \leq k$ with $v_i \in N^G_{r-1}(v)$. We will distinguish the following two cases.

First assume that there are less than λn vertices of degree $\tilde{d}$, of pairwise distance greater than $2r$ and of distance greater than $2r$ from $\{v_1, \dots, v_k\}$. In this case there are less than $2\lambda n(4d)^{2r}$ vertices of degree $\tilde{d}$ in total. Let G' be a graph obtained from G by the following modifications. For every vertex w of degree $\tilde{d}$ we pick edges $\{w, w_1\}, \{w, w_2\}, \{v, v'\}, \{u, u'\}$ such that v, w, u have pairwise distance at least 3. We delete the edges $\{w, w_1\}, \{w, w_2\}, \{v, v'\}, \{u, u'\}$ and insert the edges $\{w_1, v\}, \{w_2, u\}, \{v', u'\}$, reducing the degree of v while maintaining the degrees of all other vertices. The resulting graph has no vertex of degree $\tilde{d}$. Note that if such edges do not exist at any point during the iteration the graph contains no more than $2d^3 \leq \epsilon dn$ edges, and we delete them all resulting in a graph with no vertex of degree $\tilde{d}$. In total we did no more than $7 \cdot 2\lambda n(4d)^{2r} \leq \epsilon dn$ edge modifications which implies that G' is ϵ-close to G. In addition, G' is τ-neighbourhood free, because a neighbourhood of type τ would imply having a vertex of degree $\tilde{d}$.

Now assume that there are at least λn vertices of degree $\tilde{d}$, of pairwise distance greater than $2r$ and of distance greater than $2r$ from $\{v_1, \dots, v_k\}$. Let $\{v'_1, \dots, v'_k\}$ be a set of vertices of degree $\tilde{d}$ such that $\mathrm{dist}_G(v_i, v'_j) > 2r$ for all $1 \leq i, j \leq k$ and $\mathrm{dist}(v'_i, v'_j) > 2r$ for all $1 \leq i < j \leq k$. Let G' be the graph obtained from G by inserting the edges $\{v_i, v'_i\}$. First note that this takes no more than $\lambda n \leq \epsilon dn$ edge modifications which implies that G is ϵ-close to G'. Further assume that v' is a vertex in G' of r-type τ. By choice of the set $\{v_1, \dots, v_k\}$ we altered the isomorphism type of each vertex of type τ in G. Therefore $\mathcal{N}^{G'}_r(v') \neq \mathcal{N}^G_r(v')$. It follows that $\mathcal{N}^{G'}_r(v')$ contains an inserted edges (v_i, v'_i). First we prove that either $\mathrm{dist}_{G'}(v', v_i) < r$ or $\mathrm{dist}_{G'}(v', v'_i) < r$. Assume towards a contradiction that this is not the case. Then there is a path in G' of the form $P = (v_i = w_{-r}, w_{-r+1}, \dots, w_{-1}, w_0 = v', w_1, \dots, w_{r-1}, w_r = v'_i)$ where $w_j \neq v_i$ and $w_j \neq v'_i$ for all $-r < j < r$. Let $-r \leq j < r$ be the largest index such that $w_j \in \{v_1, \dots, v_k, v'_1, \dots, v'_k\}$. This implies that the path $(w_j, \dots, w_r = v'_i)$ is a path in G of length $\leq 2r$, which contradicts the choice of $v_1, \dots, v_k, v'_1, \dots, v'_k$. Since $\deg_{G'}(v_i) = \deg_{G'}(v'_i) = \tilde{d}+1$, this implies that $\mathcal{N}^G_{r-1}(v')$ contains a vertex of degree $\tilde{d}+1$, which contradicts that v' has r-type τ. Hence G' is τ-neighbourhood free. □

Lemma 7.3.5. *For $r \geq 1$ let τ be an r-type of degree d and $(B, b) \in \tau$. Assume $\deg_B(v) = d$ for all vertices $v \in N^B_{r-1}(b)$. Let $\epsilon \in (0, 1]$ be fixed and let $\lambda = \epsilon$. Every graph $G \in \mathcal{C}_d$ on $n \geq 1$ vertices which is ϵ-far from being τ-neighbourhood free contains more than λn vertices of r-type τ.*

Proof. If $d = 0$ then the Lemma holds. Therefore we can assume that B is not just an isolated vertex. We proceed by contraposition. Assume $G \in \mathcal{C}_d$ is a graph on $n \geq 1$ vertices containing no more than λn vertices v of r-type τ. Let G' be the graph obtained from G by isolating all vertices v of r-type τ. First note that G' is ϵ-close to G since we did no more than $d\lambda n \leq \epsilon dn$

edge modifications. Now assume that v' is a vertex of r-type τ in G'. Since we isolated all vertices having r-type τ we know that $\mathcal{N}_r^{G'}(v') \neq \mathcal{N}_r^G(v')$. Therefore there must be a vertex v in $N_r^G(v')$ such that v has type τ, because otherwise the r-ball of v' could not witness any of the edge modifications. This means that there is a path $(v' = v_0, v_1, \dots, v_{k-1}, v_k = v)$ of length $k \leq r$ in G. Now pick the maximum index i such that $\mathrm{dist}_{G'}(v', v_i) < \infty$. First observe that because $v = v_k$ is isolated in G' we get that $i < k$ and therefore $\mathrm{dist}_{G'}(v', v_i) < r$. Since $\mathrm{dist}_{G'}(v', v_{i+1}) = \infty$ by construction and $\{v_i, v_{i+1}\} \in E(G)$, we get $\deg_{\mathcal{N}_r^{G'}(v')}(v_i) = \deg_{G'}(v_i) < \deg_G(v_i) \leq d$. Since $(\mathcal{N}_r^{G'}(v'), v') \in \tau$ this yields a contradiction to our previous assumption that all vertices in $N_{r-1}^B(b)$ have degree d. Hence the graph G' can not contain a vertex v' of r-type τ and is therefore τ-neighbourhood free. □

The next Lemma follows from Lemmas 7.3.5 and 7.3.4 since for radius $r = 1$ the $(r-1)$-ball contains only one vertex.

Lemma 7.3.6. *Let τ be a 1-type. Let $\epsilon \in (0, 1]$ be fixed, $n_0 = 2d^2/\epsilon$ and $\lambda = \epsilon d/(14(4d)^2)$. Every graph $G \in \mathcal{C}_d$ on $n \geq n_0$ vertices which is ϵ-far from being τ-neighbourhood free contains more than λn vertices of 1-type τ.* □

Proof of Theorem 7.1.2. Lemma 7.3.2 with $F = \{\tau\}$ and $M = \emptyset$ combined with Lemma 7.3.4 proves Theorem 7.1.2 in all cases apart from when $d = 1$. In case $d = 1$ we have $\tilde{d} = 0$. In this case we set $M := \{n \in \mathbb{N} \mid n \equiv 1 \bmod 2\}$ and get that for $\epsilon \in (0, 1]$ and $\lambda = \epsilon$ we have that every graph $G \in \mathcal{C}_d$ on $n \equiv 0 \bmod 2$ vertices which is ϵ-far from being τ-neighbourhood free contains more than λn vertices of r-type τ. This is the case as assuming the number of vertices of r-type τ is no more than λn we can add an edge between any pair of vertices of degree 0, obtaining a graph G' which is $\lambda n \leq \epsilon d n$ close to G. □

Theorem 7.1.3 follows from Lemma 7.3.2 and Lemma 7.3.6 where in Lemma 7.3.2 we use either $\mathcal{F} = \{\tau\}$ or $\mathcal{F} = \emptyset$ depending on whether τ has degree bounded by d.

Proof of Theorem 7.1.4. Let τ be a 1-type such that $B \setminus \{b\}$ is a union of disjoint cliques for all $(B, b) \in \tau$ as in the statement of the theorem. We define $\mathcal{P}$ to be the property of being τ-neighbourhood regular and let $\mathcal{K}^G$ be the set of maximal cliques in G, i. e. the set of all cliques in G which are not properly contained in another clique of G. Let us define the function $\mathrm{maxcl}^G : V(G) \times \mathbb{N} \to \mathbb{N}$ where $\mathrm{maxcl}^G(v, i) := |\{K \in \mathcal{K}^G \mid |K| = i, v \in K\}|$ is the number of maximal i-cliques containing v.

Claim 1. If $G \in \mathcal{P}$ then $\mathrm{maxcl}^B(b, i) \cdot n \equiv 0 \bmod i$.

Proof of Claim 1. First note that $G \in \mathcal{P}$ implies that $(\mathcal{N}_1^G(v), v) \in \tau$ for all $v \in V(G)$. Therefore $\mathrm{maxcl}^B(b, i) = \mathrm{maxcl}^G(v, i)$ for all $v \in V(G)$ and

$$\mathrm{maxcl}^B(b, i) \cdot n = \sum_{v \in V(G)} \mathrm{maxcl}^G(v, i) = |\{K \in \mathcal{K}^G \mid |K| = i\}| \cdot i \equiv 0 \bmod i.$$

■

Let $M := \{n \in \mathbb{N} \mid \text{there is } 1 \leq i \leq d \text{ such that } \operatorname{maxcl}^B(b,i) \cdot n \not\equiv 0 \bmod i\}$. Note that deciding whether $n \in M$ only requires standard arithmetic operations.

Claim 2. For $\epsilon \in (0,1]$ let $\lambda = \epsilon/(20d^6)$ and $n_0 = 20d^8$. Any graph $G \in \mathcal{C}_d$ on $n \geq n_0$, $n \notin M$ vertices, which is ϵ-far from $\mathcal{P}$, contains more than λn vertices v with 1-type τ.

Proof of Claim 2. We proceed by contraposition. Let $G \in \mathcal{C}_d$ be a graph on $n \geq n_0$, $n \notin M$ vertices and assume that G contains no more than λn vertices of 1-type τ. We will now describe an algorithmic procedure which takes less than ϵdn edge modifications and transforms G into a graph $G^{(4)} \in \mathcal{P}$, which will prove the claim.

Let $\tilde{E}^{(1)} := \{e \in E(G) \mid \text{there are distinct } K, K' \in \mathcal{K}^G, |K \cap K'| > 1, e \subseteq K\}$. Let $G^{(1)}$ be the graph $G^{(1)} = (V(G), E^{(1)})$, where $E^{(1)} = E(G) \setminus \tilde{E}^{(1)}$. First note that $G^{(1)}$ has no distinct maximal cliques K, K' with $|K \cap K'| > 1$. Furthermore

$$|\tilde{E}^{(1)}| \leq \binom{d}{2} \cdot |\{K \in \mathcal{K}^G \mid \text{exists } K' \in \mathcal{K}^G, |K \cap K'| > 1\}| \leq \frac{d^3 \lambda n}{2},$$

where the second inequality holds because every $K \in \mathcal{K}^G$ such that there is $K' \in \mathcal{K}^G$ with $|K \cap K'| > 1$ and $K \neq K'$ must contain one of the λn vertices v of 1-type τ and there are no more than $d\lambda n$ maximal cliques containing such a vertex. In addition, the removal of the edges in $\tilde{E}^{(1)}$ will affect no more than $d^4\lambda n$ vertices because there are no more than $d^3\lambda n$ vertices contained within an edge of $\tilde{E}^{(1)}$, each of their 1-neighbourhoods contains d vertices and any vertex, whose 1-neighbourhood is affected, must be of distance 1 to one of the vertices contained in an edge in $\tilde{E}^{(1)}$. Hence $G^{(1)}$ contains no more than $(d^4+1)\lambda n < 2d^4\lambda n$ vertices v of 1-type τ.

Note that in $G^{(1)}$ for all vertices v the graph $\mathcal{N}_1^{G^{(1)}}(v) \setminus \{v\}$ is a disjoint union of cliques but there might be $K \in \mathcal{K}^{G^{(1)}}$ such that $\operatorname{maxcl}^B(b, |K|) = 0$. We define the edge set $\tilde{E}^{(2)} := \{e \in E^{(1)} \mid \text{exists } K \in \mathcal{K}^{G^{(1)}}, e \subseteq K, \operatorname{maxcl}^B(b,|K|) = 0\}$ and let $G^{(2)}$ be the graph $G^{(2)} = (V(G), E^{(2)})$, where $E^{(2)} = E^{(1)} \setminus \tilde{E}^{(2)}$. Furthermore

$$|\tilde{E}^{(2)}| \leq d \cdot |\{v \mid \text{exists } K \in \mathcal{K}^{G^{(1)}}, v \in K, \operatorname{maxcl}^B(b,|K|) = 0\}| \leq d \cdot 2d^4\lambda n,$$

where the first inequality holds because every clique in $G^{(1)}$ has size $\leq d$ and the second because $(\mathcal{N}_1^{G^{(1)}}(v), v) \notin \tau$ for every $v \in \{v \mid \text{exists } K \in \mathcal{K}^{G^{(1)}}, v \in K, \operatorname{maxcl}^B(b,|K|) = 0\}$.

Note that $\operatorname{maxcl}^B(b,|K|) \neq 0$ for all $K \in \mathcal{K}^{G^{(2)}}$, but there might be $v \in V(G)$ and $i \leq d$ with $\operatorname{maxcl}^B(b,i) \neq \operatorname{maxcl}^{G^{(2)}}(v,i)$. Moreover, note that because $n \geq n_0$ there are at least $2d$ balls of radius 4 in $G^{(2)}$ which are completely disjoint from the 4-balls of any vertex v of 1-type τ. $G^{(3)}$ will also have this property. Let $G^{(3)} = (V(G), E^{(3)})$ be the graph obtained from $G^{(2)}$ by the following operations. For every pair v, v' such that there is $i \leq d$ with $\operatorname{maxcl}^B(b,i) > \operatorname{maxcl}^{G^{(2)}}(v,i)$ and $\operatorname{maxcl}^B(b,i) < \operatorname{maxcl}^{G^{(2)}}(v',i)$, let w be a vertex of type

τ which has at least distance 4 to v and to v'. Let $K' = \{v'_1, \dots, v'_{i-1}, v'\} \in \mathcal{K}^{G^{(2)}}$ and $K = \{v_1, \dots, v_{i-1}, w\} \in \mathcal{K}^{G^{(2)}}$. Delete the edges $\{\{v', v'_j\}, \{w, v_j\} \mid j \in [i-1]\}$ and add the edges $\{\{v, v_j\}, \{w, v'_j\} \mid j \in [i-1]\}$. Note that the vertices $v_1, \dots, v_{i-1}, v'_1, \dots, v'_{i-1}, w$ are still contained in the same number of cliques as before, while v is contained in one additional i-clique and v' is contained in one less.

Note that in $G^{(3)}$ either $\mathrm{maxcl}^B(b,i) \geq \mathrm{maxcl}^{G^{(3)}}(v,i)$ for all vertices $v \in V(G)$ or $\mathrm{maxcl}^B(b,i) \leq \mathrm{maxcl}^{G^{(3)}}(v,i)$ for all v for every $i \in [d]$. Let $G^{(4)}$ be the graph obtained from $G^{(3)}$ by the following operations. For every i such that there is a vertex $v \in V(G)$ with $\mathrm{maxcl}^B(b,i) < \mathrm{maxcl}^{G^{(3)}}(v,i)$, we pick i not necessarily distinct vertices $v_1, \dots, v_i$ with $i \cdot \mathrm{maxcl}^B(b,i) < \sum_{v \in \{v_1, \dots, v_i\}} \mathrm{maxcl}^{G^{(3)}}(v,i)$. Note that these choices are possible because $\sum_{v \in V(G)} \mathrm{maxcl}^{G^{(3)}}(v,i) \equiv 0 \bmod i$ and $\mathrm{maxcl}^B(b,i) \cdot n \equiv 0 \bmod i$ by assumption $n \notin M$ and hence we have $\sum_{v \in V(G)} (\mathrm{maxcl}^{G^{(3)}}(v,i) - \mathrm{maxcl}^B(b,i)) \equiv 0 \bmod i$. Let $K_1, \dots, K_i \in \mathcal{K}^{G^{(3)}}$ be distinct cliques such that $v_j \in K_j$ for every $1 \leq j \leq i$. Let $K = \{w_1, \dots, w_i\} \in \mathcal{K}^{G^{(3)}}$ such that the distance between any pair v_j, w_k is at least 4. Remove the set of edges $\{\{w_j, w_k\}, \{v_j, v\} \mid v \in K_j, j, k \in [i]\}$ and add the set of edges $\{\{w_j, v\} \mid v \in K_j, j \in [i]\}$. Note that this reduces the number of maximal i-cliques $v_1, \dots, v_i$ are in by one, while leaving the number of cliques $w_1, \dots, w_i$ are in the same. Similarly, for every i such that there is a vertex v with $\mathrm{maxcl}^B(b,i) > \mathrm{maxcl}^{G^{(3)}}(v,i)$ we pick i not necessarily distinct vertices $v_1, \dots, v_i$ with $i \cdot \mathrm{maxcl}^B(b,i) > \sum_{v \in \{v_1, \dots, v_i\}} \mathrm{maxcl}^{G^{(3)}}(v,i)$. Let $w_1, \dots, w_i$ be vertices with $\mathrm{maxcl}^B(b,i) = \mathrm{maxcl}^{G^{(3)}}(w_j,i)$ such that $w_1, \dots, w_i$ are of distance at least 4 from every v_j, $1 \leq j \leq i$, and $w_1, \dots, w_i$ are pairwise of distance at least 4. Let $K_j \in \mathcal{K}^{G^{(3)}}$ with $w_j \in K_j$ for $1 \leq j \leq i$. Remove the set of edges $\{\{w_j, w\} \mid w \in K_j, 1 \leq j \leq i\}$ and add the set of edges $\{\{v_j, w\}\{w_j, w_k\} \mid w \in K_j, j, k \in [i]\}$. Note that this adds one to the number of i-cliques $v_1, \dots, v_i$ are in, while leaving the number of i-cliques $w_1, \dots, w_i$ are in the same.

By construction $G^{(4)} \in \mathcal{P}$. The number of edge modifications in total is $|E^{(1)}| + |E^{(2)}|$ plus the number of modifications it takes to transform $G^{(2)}$ into $G^{(4)}$. First note that

$$\sum_{i=2}^{d} \sum_{v \in V(G)} |\mathrm{maxcl}^B(b,i) - \mathrm{maxcl}^{G^{(2)}}(v,i)| \leq 2d \cdot 2d^4 \lambda n$$

since every of the at most $2d^4\lambda n$ vertices v in $G^{(2)}$ of 1-type τ can contribute at most $2d$ to the sum above. Since transforming $G^{(2)}$ into $G^{(4)}$ we proceed greedily, meaning we reduce the number $\sum_{i=3}^{d} \sum_{v \in V(G)} |\mathrm{maxcl}^B(b,i) - \mathrm{maxcl}^{G^{(2)}}(v,i)|$ by at least one in every step, and every such reduction takes a maximum of $4d^2$ edge modifications in total we need less than

$$|E^{(1)}| + |E^{(2)}| + 4d^2 \cdot 2d \cdot 2d^4 \lambda n \leq 20 d^7 \lambda n = \epsilon d n$$

edge modifications. ■

Let $\mathcal{F} := \{\tau' \mid \tau \text{ is a 1-type }, \tau \neq \tau'\}$. Note that $|\mathcal{F}| \leq |T_r| < \infty$, where equality occurs when $B \notin \mathcal{C}_d$. Then Claim 2 combined with Lemma 7.3.2 for M and $\mathcal{F}$ defined as above proves the Theorem. □

7.4 Summary

In this section we have shown that there are certain fragments of FO which yield testable properties. More precisely we consider the fragment of FO sentences expressing that a certain r-neighbourhood can not appear in a graph and prove testability under a mild assumption on the degrees. Moreover, we consider the fragment of FO sentences expressing that every vertex has to have the same r-neighbourhood and show testability for radius $r = 1$ in some special cases. The properties considered in this section are interesting as they are natural extensions of properties that are known to be testable (i. e. properties defined by a Hanf sentence which are trivially testable, degree regularity [69]). Furthermore, there are sentences defining neighbourhood regularity and neighbourhood freeness which are contained in $\Pi_2 \setminus \Sigma_2$. This implies that there can not be a dichotomy of the form that an FO sentence defines a testable property if and only if it is contained in Σ_2. We believe that the study of the properties in this section enhances our understanding of testability of FO definable properties.

Chapter 8

Comparing locality notions and answering an open question

Since Gaifman's locality Theorem [63], it is known that FO can only express local properties, for some notion of locality. And hence in Chapter 6 we proved that locality, as prescribed in Gaifman's locality Theorem, is not sufficient for property testing in the bounded degree model. Since a constant query property tester can only explore a graph locally, locality plays a central role in property testing. Indeed, considering how property testers explore graphs has led to a different notion of locality. With the purpose of characterising properties which have a one-sided error POT, Goldreich and Ron in [76] defined local properties as generalised subgraph freeness properties. We refer to this notion of locality as GSF-locality. In [76] Goldreich and Ron show that the graph properties that allow constant query proximity oblivious testing in the bounded degree model are precisely the properties that are GSF-local and satisfy the non-propagation condition. Whether the non-propagating condition is necessary is formulated as an open question in [76]. We answer this question negatively by proving the following Theorem.

Theorem 8.0.1. *There exists a GSF-local property that is not testable in the bounded degree graph model. Thus, not all GSF-local properties are non-propagating.*

The notion of GSF-locality has also been used by Ito, Khoury and Newman in their recent work [89] which classifies which monotone properties and which hereditary properties are testable by a one-sided error constant query property tester in bounded degree digraphs (in both the unidirectional and the bidirectional model). In monotone GSF-local properties and in hereditary GSF-local properties the non-propagating condition is always satisfied and hence the classification given in [89] is not effected by our result. The authors further prove that every property which is testable by a one-sided error constant query property tester for bounded degree digraphs is close to being a GSF-local property. Whether the converse is true is asked as an open question in [89]. Indeed, if this question could be answered positively we could characterise properties which are testable by a one-sided error constant query property tester. Our result shows that this is already not possible in the bounded degree model which can be seen as a special case of both bounded degree digraph models considered in [89].

Proof outline To prove Theorem 8.0.1 we make use of the FO definable property $P_{\textcircled{z}}$ from Section 6. Hanf's Theorem [83] implies that we can understand locality of FO as prescribing upper and lower bounds for the occurrence of certain local neighbourhood (isomorphism) types.

On the other hand, a GSF-local property as defined in [76] refers to properties which do not contain embeddings of some constant size marked graphs. Here the markings of graphs signify how the embedded graph interacts with the rest of the graph. This is motivated by the fact that a property tester does not just obtain an induced subgraph of the input graph but also obtains the information how this induced subgraph is connected to the rest of the graph. However, forbidding the embedding of such a marked graph is intuitively similar to excluding certain neighbourhood types, or in other words, limiting the number of occurrences of certain types.

Building upon the above observations, we establish a formal connection between FO properties and GSF-local properties. We first encode the possible bounds on occurrences of local neighbourhood types into what we call *neighbourhood profiles*, and characterise FO definable properties of bounded degree relational structures as finite unions of properties defined by neighbourhood profiles (Lemma 8.2.3). We then show that every FO formula defined by a non-trivial finite union of properties which in turn is defined by so-called *0-profiles*, i. e. the prescribed lower bounds are all 0, is GSF-local (Theorem 8.2.5). Given the fundamental role of local properties in graph theory, graph limits [102], we believe this new connection is of independent interest.

For technical reasons, we make use of the property $P_{\textcircled{z}}$ of *relational structures* instead of directly using the non-testable graph property from Section 6.3. We further prove that a variant of the relational structure property $P_{\textcircled{z}}$, which we denote by $P'_{\textcircled{z}}$, can be defined by 0-profiles (Lemma 8.3.2). Finally, we construct a non-testable *graph* property $\mathcal{P}_{\text{graph}}$ by a *local* reduction from the σ-structure property $P'_{\textcircled{z}}$ (Lemma 8.3.3). In the reduction we maintain being definable by 0-profiles which proves GSF-locality of the graph property $\mathcal{P}_{\text{graph}}$ (Lemma 8.3.4).

8.1 Generalised subgraph freeness

Now we present the formal definition of generalised subgraph freeness, GSF-local properties and the notion of non-propagation, which were introduced in [76].

Definition 8.1.1 (Generalized subgraph freeness (GSF))**.** A *marked* graph is a graph with each vertex marked as either *'full'* or *'semifull'* or *'partial'*. An *embedding* of a marked graph F into a graph G is an injective map $f : V(F) \to V(G)$ such that for every $v \in V(F)$ the following three conditions hold.

– If v is marked 'full', then $N_1^G(f(v)) = f(N_1^F(v))$.

– If v is marked 'semifull', then $N_1^G(f(v)) \cap f(V(F)) = f(N_1^F(v))$.

– If v is marked 'partial', then $N_1^G(f(v)) \supseteq f(N_1^F(v))$.

The graph G is called *F-free* if there is no embedding of F into G. For a set of marked graphs $\mathcal{F}$, a graph G is called *$\mathcal{F}$-free* if it is F-free for every $F \in \mathcal{F}$.

Based on the above definition of generalised subgraph freeness, we can define GSF-local properties.

Definition 8.1.2 (GSF-local properties)**.** Let $\mathcal{P} = \bigcup_{n\in\mathbb{N}} \mathcal{P}_n$ be a graph property where $\mathcal{P}_n = \{G \in \mathcal{P} \mid |V(G)| = n\}$ and $\overline{\mathcal{F}} = (\mathcal{F}_n)_{n\in\mathbb{N}}$ a sequence of sets of marked graphs. $\mathcal{P}$ is called *$\overline{\mathcal{F}}$-local* if there exists an integer s such that for every n the following conditions hold.

– $\mathcal{F}_n$ is a set of marked graphs, each of size at most s.

– $\mathcal{P}_n$ equals the set of n-vertex graphs that are $\mathcal{F}_n$-free.

$\mathcal{P}$ is called *GSF-local* if there is a sequence $\overline{\mathcal{F}} = (\mathcal{F}_n)_{n\in\mathbb{N}}$ of sets of marked graphs such that $\mathcal{P}$ is $\overline{\mathcal{F}}$-local.

The following notion of non-propagating condition of a sequence of sets of marked graphs was introduced to study constant query POTs in [76].

Definition 8.1.3 (Non-propagating)**.** Let $\overline{\mathcal{F}} = (\mathcal{F}_n)_{n\in\mathbb{N}}$ be a sequence of sets of marked graphs.

– For a graph G, a subset $B \subseteq V(G)$ *covers* $\mathcal{F}_n$ in G if for every marked graph $F \in \mathcal{F}_n$ and every embedding of F in G, at least one vertex of F is mapped to a vertex in B.

– The sequence $\overline{\mathcal{F}}$ is *non-propagating* if there exists a (monotonically non-decreasing) function $\tau : (0, 1] \to (0, 1]$ such that the following two conditions hold.

  - For every $\epsilon > 0$ there exists $\beta > 0$ such that $\tau(\beta) < \epsilon$.
  - For every graph G and every $B \subseteq V(G)$ such that B covers $\mathcal{F}_n$ in G, either G is $\tau\left(\frac{|B|}{n}\right)$-close to being $\mathcal{F}_n$-free or there are no graphs with n vertices that are $\mathcal{F}_n$-free.

  A GSF-local property $\mathcal{P}$ is *non-propagating* if there exists a non-propagating sequence $\overline{\mathcal{F}}$ such that $\mathcal{P}$ is $\overline{\mathcal{F}}$-local.

In the above definition, the set B can be viewed as the set involving necessary modifications for repairing a graph G that does not satisfy the property $\mathcal{P}$ that is $\overline{\mathcal{F}}$-local, and the second condition says we do not need to modify G "much beyond" B. In particular, it implies we can repair G without triggering a global "chain reaction". Note that there are sequences $\overline{\mathcal{F}} = (\mathcal{F}_n)_{n\in\mathbb{N}}$ that are propagating as mentioned in [76].

Goldreich and Ron gave the following characterisation for the proximity-oblivious testable properties in the bounded degree model of graphs.

Theorem 8.1.4 (Theorem 5.5 in [76])**.** *A graph property $\mathcal{P}$ has a constant query POT if and only if $\mathcal{P}$ is GSF-local and non-propagating.*

The following open question was raised in [76].

Question 8.1.5 (Are all GSF-local properties non-propagating?)**.** Is it the case that for every GSF-local property $\mathcal{P} = \bigcup_{n\in\mathbb{N}} \mathcal{P}_n$, there is a sequence $\overline{\mathcal{F}} = (\mathcal{F}_n)_{n\in\mathbb{N}}$ that is non-propagating and $\mathcal{P}$ is $\overline{\mathcal{F}}$-local?

8.2 Relating different notions of locality

In this section we define properties by prescribing upper and lower bounds on the number of occurrence of neighbourhood types. These bounds are given by *neighbourhood profiles* which we will define formally below. We use these properties to give a natural characterisation of FO properties of bounded degree structures in Lemma 8.2.3, which is a straightforward consequence of Hanf's Theorem (Theorem 2.4.7). We use this characterisation to establish links between FO definability and GSF-locality. This connection is the key ingredient in the proof of our main theorem.

Let σ be a signature and $d \in \mathbb{N}$. For every $r \in \mathbb{N}$ we assume we fixed an ordering of all r-types of bounded degree d. We further associate with each σ-structure $\mathcal{A} \in C_d$ its *r-histogram vector* $\mathrm{hist}_r(\mathcal{A})$ defined in Definition 2.3.5 with respect to the chosen ordering of r-types. We let

$$\mathfrak{I} := \{[k, l], [k, \infty) \mid k \le l \in \mathbb{N}\}$$

be the set of all closed or half-closed, infinite intervals with natural lower/upper bounds.

Definition 8.2.1. Let $r \in \mathbb{N}$ and t be the number of r-types of bounded degree d.

1. An *r-neighbourhood profile* of degree d is a function $\rho : \{1, \dots, t\} \to \mathfrak{I}$.
2. For a structure $\mathcal{A} \in C_d$, we say $\mathcal{A}$ *obeys* ρ, denoted by $\mathcal{A} \sim \rho$, if

   $$(\mathrm{hist}_r(\mathcal{A}))_i \in \rho(i) \text{ for all } i \in \{1, \dots, t\}.$$

   Let P_ρ be the set of structures $\mathcal{A}$ that obey ρ, i.e., $P_\rho = \{\mathcal{A} \in C_d \mid \mathcal{A} \sim \rho\}$.
3. We say that a property P is *defined by a finite union of neighbourhood profiles* if there is $k \in \mathbb{N}$ such that $P = \bigcup_{1\le i\le k} P_{\rho_i}$ where ρ_i is an r_i-neighbourhood profile and $r_i \in \mathbb{N}$ for every $i \in \{1, \dots, k\}$.

Let us consider the following example in which we find a representation by neighbourhood profiles for an FO-property.

Figure 8.1: One types of bounded degree 2, where the centres are marked in green.

Example 8.2.2. Let us consider the following FO-sentence.

$$\varphi := \forall x \forall y \neg E(x,y) \vee \forall x \exists y_1 \exists y_2 \Big(y_1 \neq y_2 \wedge E(x,y_1) \wedge E(x,y_2) \\ \wedge \forall z (z \neq y_1 \wedge z \neq y_2) \to \neg E(x,z) \Big).$$

The property P_φ defined by the sentence φ is the property containing all edgeless graphs and all graphs that are disjoint unions of cycles.

For degree bound 2 all 1-types are listed in Figure 8.1. Let $\rho_1 : \{1,\dots,4\} \to \mathfrak{I}$ be the neighbourhood profile defined by $\rho_1(1) = [0,\infty)$ and $\rho_1(i) = [0,0]$ for $i \in \{2,3,4\}$. Furthermore, let $\rho_2 : \{1,\dots,4\} \to \mathfrak{I}$ be the neighbourhood profile defined by $\rho_2(i) = [0,\infty)$ for $i \in \{3,4\}$ and $\rho_2(j) = [0,0]$ for $j \in \{1,2\}$. It is easy to observe that the properties P_φ and $P_{\rho_1} \cup P_{\rho_2}$ are equal.

Indeed representing FO-properties by neighbourhood profiles works in general. The following lemma shows that bounded degree FO properties can be equivalently defined as finite unions of properties defined by neighbourhood profiles. Here the technicalities that arise are due to Hanf normal form not requiring the locality-radius of all Hanf-sentences to be the same.

Lemma 8.2.3. *For every non-empty property $P \subseteq C_d$, P is FO definable on C_d if and only if P can be obtained as a finite union of properties defined by neighbourhood profiles.*

Proof. For the first direction assume φ is an FO-sentence. Then by Hanf's Theorem (Theorem 2.4.7) there is a sentence ψ in Hanf normal form such that $P_\varphi = P_\psi$.

We will first convert ψ into a sentence in Hanf normal form where every Hanf sentence appearing has the same locality radius. Let $r \in \mathbb{N}$ be the maximum locality radius appearing in ψ, and let $\varphi_\tau^{\geq m} := \exists^{\geq m} x \phi_\tau(x)$ be a Hanf sentence, where τ is an r'-type for some $r' < r$. Let $\tau_1, \dots, \tau_k$ be a list of all r-types of bounded degree d for which $(\mathcal{N}_{r'}^{\mathcal{B}}(b), b) \in \tau$ for $(\mathcal{B}, b) \in \tau_i$, for every $1 \leq i \leq k$. Let Π be the set of all partitions of m into k parts. Let

$$\tilde{\varphi}_\tau^{\geq m} := \bigvee_{(m_1,\dots,m_k)\in\Pi} \bigwedge_{i=1}^{k} \exists^{\geq m_i} x \phi_{\tau_i}(x).$$

Claim 1. $\varphi_\tau^{\geq m}$ is d-equivalent to $\tilde{\varphi}_\tau^{\geq m}$.

Proof of Claim 1. Assume that $\mathcal{A} \in C_d$ satisfies $\varphi_\tau^{\geq m}$, and assume that $a_1, \dots, a_m$ are m distinct elements with $(\mathcal{N}_{r'}^{\mathcal{A}}(a_j), a_j) \in \tau$, for every $1 \leq j \leq m$. Let $\tilde{\tau}_j$ be the r-type for which

$(\mathcal{N}_r^{\mathcal{A}}(a_j), a_j) \in \tilde{\tau}_j$. By choice of $\tau_1, \dots, \tau_k$, we get that there are indices $i_1, \dots, i_m$ such that $\tilde{\tau}_j = \tau_{i_j}$. For $i \in \{1, \dots, k\}$ let $m_i = |\{j \in \{1, \dots, m\} \mid i_j = i\}|$. Hence $\mathcal{A} \models \bigwedge_{i=1}^{k} \exists^{\geq m_i} x \phi_{\tau_i}(x)$ and since additionally $(m_1, \dots, m_k) \in \Pi$ this implies $\mathcal{A} \models \tilde{\varphi}_{\tau}^{\geq m}$.

On the other hand, let $\mathcal{A} \in C_d$ satisfy $\tilde{\varphi}_{\tau}^{\geq m}$, and let $(m_1, \dots, m_k) \in \Pi$ be a partition of m such that $\mathcal{A} \models \bigwedge_{i=1}^{k} \exists^{\geq m_i} x \phi_{\tau_i}(x)$. For every $1 \leq i \leq k$, let $a_1^i, \dots, a_{m_i}^i$ be m_i distinct elements such that $(\mathcal{N}_r^{\mathcal{A}}(a_j^i), a_j^i) \in \tau_i$, for every $1 \leq j \leq m_i$. By choice of $\tau_1, \dots, \tau_k$, we get that $(\mathcal{N}_{r'}^{\mathcal{A}}(a_j^i), a_j^i) \in \tau$, for every pair $1 \leq i \leq k$, $1 \leq j \leq m_i$. But since $m_1 + \dots + m_k = m$ this implies that $\mathcal{A} \models \varphi_{\tau}^{\geq m}$. This proves that $\varphi_{\tau}^{\geq m}$ and $\tilde{\varphi}_{\tau}^{\geq m}$ are d-equivalent. ■

Let ψ' be the formula in which every Hanf-sentence $\varphi_{\tau}^{\geq m}$ for which τ is an r'-type for some $r' < r$ gets replaced by $\tilde{\varphi}_{\tau}^{\geq m}$. By a simple inductive argument using Claim 1, we get that ψ is d-equivalent to ψ', and hence $P_{\varphi} = P_{\psi} = P_{\psi'}$. Furthermore since $\tilde{\varphi}_{\tau}^{\geq m}$ is a Boolean combination of Hanf-sentences for every $\varphi_{\tau}^{\geq m}$, and any Boolean combination of Boolean combinations is a Boolean combination itself, ψ' is in Hanf normal form. Furthermore, every Hanf-sentence appearing in ψ' has locality radius r by construction.

Since any Boolean combination can be converted into disjunctive normal form (Lemma 2.4.3), we can assume that ψ' is a disjunction of sentences ξ of the form

$$\xi = \bigwedge_{j=1}^{k} \exists^{\geq m_j} x \phi_{\tau_j}(x) \wedge \bigwedge_{j=k+1}^{\ell} \neg \exists^{\geq m_j + 1} x \phi_{\tau_j}(x),$$

where $\ell \in \mathbb{N}_{\geq 1}$, $1 \leq k \leq \ell$, $m_i \in \mathbb{N}_{\geq 1}$ and τ_i is an r-type for every $1 \leq i \leq \ell$. We can further assume that every sentence in the disjunction ψ' is satisfiable by some $\mathcal{A} \in C_d$, as any sentence with no bounded degree d model can be removed from ψ'.

Let $\tilde{\tau}_1, \dots, \tilde{\tau}_t$ be a list of all r-types of bounded degree d in the order we fixed. Let $k_i := \max(\{m_j \mid 1 \leq j \leq k, \tau_j = \tilde{\tau}_i\} \cup \{0\})$ and $\ell_i := \min(\{m_j \mid k+1 \leq j \leq \ell, \tau_j = \tilde{\tau}_i\} \cup \{\infty\})$ for every $i \in \{1, \dots, t\}$. Since ξ has at least one bounded degree model $k_i \leq \ell_i$ for every $i \in \{1, \dots, t\}$. Let $\rho : \{1, \dots, t\} \to \mathfrak{I}$ be the neighbourhood profile defined by $\rho(i) := [k_i, \ell_i]$ if $\ell_i < \infty$ and $\rho(i) := [k_i, \ell_i)$ otherwise. Then by construction, we get that $P_{\rho} = P_{\xi}$. Since ψ' is a disjunction of formulas, each of which defines a property which can be defined by some neighbourhood profile, we get that $P_{\psi'}$ must be a finite union of properties defined by some neighbourhood profile.

On the other hand, for every r-neighbourhood profile ρ of degree d, $\tau_1, \dots, \tau_t$ a list of all r-types of bounded degree d in the order fixed and the formula

$$\varphi_{\rho} := \bigwedge_{\substack{i \in \{1,\dots,t\}, \\ \rho(i) = [k_i, \ell_i]}} \left(\exists^{\geq k_i} x \phi_{\tau_i}(x) \wedge \neg \exists^{\geq \ell_i + 1} x \phi_{\tau_i}(x) \right) \wedge \bigwedge_{\substack{i \in \{1,\dots,t\}, \\ \rho(i) = [k_i, \infty)}} \exists^{\geq k_i} x \phi_{\tau_i}(x)$$

it clearly holds that $P_{\rho} = P_{\varphi_{\rho}}$. Hence every finite union of properties defined by neighbourhood profiles can be defined by the disjunction of the formulas φ_{ρ} of all ρ in the finite union. □

8.2.1 Relating FO properties to GSF-local properties

We now prove that FO properties which arise as unions of neighbourhood profiles of a particularly simple form are GSF-local. For this let

$$\mathfrak{I}_0 := \{[0,\infty), [0,k] \mid k \in \mathbb{N}\} \subseteq \mathfrak{I}.$$

We call any neighbourhood profile ρ with codomain $\mathfrak{I}_0$ a 0-*profile*, as all lower bounds for the occurrence of types are 0.

Observation 8.2.4. Let ρ be a 0-profile, $r \in \mathbb{N}$ and t be the number of all r-types of bounded degree d. If two structures $\mathcal{A}, \mathcal{A}' \in C_d$ satisfy $(\mathrm{hist}_r(\mathcal{A}))_i \leq (\mathrm{hist}_r(\mathcal{A}'))_i$ for every $i \in \{1, \dots, t\}$ and $\mathcal{A}' \sim \rho$, then $\mathcal{A} \sim \rho$.
In particular, the existence of an r-type cannot be expressed by a 0-profile.

While we need the concept of a 0-profile for relational structures in general the following theorem can only be stated for graphs as the concept of GSF-locality is not defined for structures. Hence for the following theorem we only consider graphs. Let $d \in \mathbb{N}$ and $\mathcal{C}_d$ be the class of all graphs of bounded degree d.

Theorem 8.2.5. *Every finite union of properties defined by* 0*-profiles is GSF-local.*

Proof. We prove this in two parts (Claim 1 and Claim 2). We first argue that every property P_ρ defined by some 0-profile $\rho : \{1, \dots, t\} \to \mathfrak{I}_0$ is GSF-local, where $\tau_1, \dots, \tau_t$ denotes a list of all r-types of bounded degree d and $r \in \mathbb{N}$ is fixed. For this it is important to note that we can express a forbidden r-type τ by a forbidden generalised subgraph. For $(B, b) \in \tau$, the set of all graphs with no vertex of neighbourhood type τ is the set of all B-free graphs where every vertex in $V(B)$ of distance less than r to b is marked 'full' and every vertex in $V(B)$ of distance r to b is marked 'semifull'. Since a profile of the form $\rho : \{1, \dots, t\} \to \mathfrak{I}_0$ can express that some neighbourhood type τ can appear at most k times for some fixed $k \in \mathbb{N}$, we need to forbid all marked graphs in which type τ appears $k+1$ times. We will formalise this in the following claim.

Claim 1. For the r-neighbourhood profile $\rho : \{1, \dots, t\} \to \mathfrak{I}_0$, there is a finite set $\mathcal{F}$ of marked graphs such that P_ρ is exactly the property of $\mathcal{F}$-free graphs.

Proof of Claim 1. Assume τ is an r-type and $k \in \mathbb{N}_{>0}$. Then we say that a marked graph F is a *k-realisation* of τ if F has the following properties.

- There are k distinct vertices $v_1, \dots, v_k$ in F such that $(\mathcal{N}_r^F(v_i), v_i) \in \tau$ for every $i = 1, \dots, k$.
- Every vertex v in F has distance less or equal to r to at least one vertex v_i.
- Every vertex v in F of distance less than k to at least one v_i is marked as 'full'.

– Every vertex v in F of distance greater or equal to k to every v_i is marked as 'semifull'.

We denote by $S^k(\tau)$ the set of all k-realisations of τ.

Now we can define the set $\mathcal{F}$ of forbidden subgraphs to be

$$\mathcal{F} := \bigcup_{\substack{k\in\mathbb{N},\, 1\leq i\leq t \\ \rho(i)=[0,k]}} S^{k+1}(\tau_i).$$

Let $\mathcal{P}$ be the property of all $\mathcal{F}$-free graphs. We first prove that the property $\mathcal{P}$ is contained in P_ρ. Towards a contradiction assume that $G \in \mathcal{C}_d$ is $\mathcal{F}$-free but not contained in P_ρ. As G is not contained in P_ρ there must be an index $i \in \{1, \dots, t\}$ such that $(\text{hist}_r(G))_i \notin \rho(i)$. Since $\rho(i) \in \mathfrak{I}_0$ there is $k \in \mathbb{N}$ such that $\rho(i) = [0, k]$ and hence $(\text{hist}_r(G))_i > k$. Hence there must be $k+1$ vertices $v_1, \dots, v_{k+1}$ in G such that $(\mathcal{N}_r^G(v_i), v_i) \in \tau_i$. We define the marked graph F to be the subgraph of G induced by the r-neighbourhoods of $v_1, \dots, v_{k+1}$, i. e. $G[\cup_{1\leq i\leq k+1} N_r^G(v_i)]$, in which every vertex of distance less than k to at least one of the v_i is marked as 'full' and every other vertex is marked as 'semifull'. Then F is by definition a $(k+1)$-realisation of τ_i and hence $F \in \mathcal{F}$. We now argue that F can be embedded into G. Since F is an induced subgraph of G the identity map gives us a natural embedding $f : F \to G$. Let v be any vertex marked 'full' in F. By construction of F, there is $i \in \{1, \dots, k+1\}$ such that $f(v)$ is of distance less than r to v_i in G. But then $N_1^G(f(v))$ is a subset of $N_r^G(v_i)$. As F without the marking is the subgraph of G induced by $\cup_{1\leq i\leq k+1} N_r^G(v_i)$ this implies that $f(N_1^F(v)) = N_1^G(f(v))$. Furthermore, assume v is a vertex marked 'semifull' in F. Then $f(N_1^F(v)) = N_1^G(f(v)) \cap f(V(F))$ holds as F without the markings is an induced subgraph of G. This proves that G is not F-free by Definition 8.1.1. This is a contradiction to our assumption that G is $\mathcal{F}$-free and $F \in \mathcal{F}$.

Similarly, we can show that $P_\rho \subseteq \mathcal{P}$ by assuming $G \in \mathcal{C}_d$ is in P_ρ but not $\mathcal{F}$-free, and showing that the embedding of any graph of $\mathcal{F}$ into G yields an amount of vertices of a certain type contradicting containment in P_ρ. ■

Next we prove that classes defined by excluding finitely many marked graphs are closed under finite unions. For two marked graphs F_1, F_2 consider the union of the class of all $\{F_1\}$-free graphs with the class of all $\{F_2\}$-free graphs. Every graph in this union excludes F_1 as a generalised subgraph or excludes F_2 as a generalised subgraph. Hence we have to forbid any marked graph whose presence in a graph G as a generalised subgraph would imply that both F_1 and F_2 are generalised subgraphs of G. We formalise this in the following claim.

Claim 2. Let $\mathcal{F}_1, \mathcal{F}_2$ be two finite sets of marked graphs. For $i \in \{1, 2\}$, let $\mathcal{P}_i$ be the property of $\mathcal{F}_i$-free graphs. Then there is a set $\mathcal{F}$ of generalised subgraphs such that $\mathcal{P}_1 \cup \mathcal{P}_2$ is the property of $\mathcal{F}$-free graphs.

Proof of Claim 2. We say that a marked graph F is a (not necessarily disjoint) *union of marked graphs* F_1, F_2 if

– there is an embedding f_i of F_i into the graph F without its markings as in Definition 8.1.1 for every $i \in \{1, 2\}$.

– for every vertex v in F there is $i \in \{1, 2\}$ and a vertex w in F_i such that $f_i(w) = v$.

– every vertex v in F is marked 'full', if there is $i \in \{1, 2\}$ and a 'full' vertex w in F_i such that $f_i(w) = v$.

– every vertex v in F is marked 'semifull', if there is $i \in \{1, 2\}$ and a 'semifull' vertex w in F_i such that $f_i(w) = v$ and $f_i(u) \neq v$ for every $i \in \{1, 2\}$ and every 'full' vertex u.

– every vertex v in F is marked 'partial' if $f_i(u) \neq v$ for every $i \in \{1, 2\}$ and every 'full' or 'semifull' vertex u.

We define $S(F_1, F_2)$ to be the set of all possible (not necessarily disjoint) unions of F_1, F_2. We can now define the set $\mathcal{F}$ to be

$$\mathcal{F} := \bigcup_{F_1 \in \mathcal{F}_1, F_2 \in \mathcal{F}_2} S(F_1, F_2).$$

Let $\mathcal{P}$ be the property of all $\mathcal{F}$-free graphs. Now we prove $\mathcal{P} \subseteq \mathcal{P}_1 \cup \mathcal{P}_2$. Towards a contradiction assume G is $\mathcal{F}$-free but G is in neither $\mathcal{P}_1$ nor in $\mathcal{P}_2$. Then for every $i \in \{1, 2\}$ there is a graph $F_i \in \mathcal{F}_i$ such that G is not F_i-free. It is easy to see that there is a union $F_\cup$ of F_1 and F_2 such that G is not $F_\cup$-free, which contradicts that G is $\mathcal{F}$-free.

Conversely, in order to prove $\mathcal{P}_1 \cup \mathcal{P}_2 \subseteq \mathcal{P}$, if G is $\mathcal{F}_i$ free for some $i \in \{1, 2\}$ then G must be $\mathcal{F}$-free by construction of $\mathcal{F}$. ■

Combining the two claims above proves the Theorem 8.2.5. □

Further discussion of the relation between FO definablility and GSF-locality First let us remark that it is neither true that every FO definable property is GSF-local, nor that every GSF-local property is FO definable.

Example 8.2.6. The property of bounded degree graphs containing a triangle is FO definable but not GSF-local.

Indeed, the existence of a fixed number of vertices of certain neighbourhood types can be expressed in FO, while in general, this cannot be expressed by forbidding generalised subgraphs. If a formula has a 0-profile (and hence does not require the existence of any types) then the property defined by that formula is GSF-local, as shown in Theorem 8.2.5.

Example 8.2.7. The class of all bounded degree graphs with an even number of vertices is GSF-local but not FO definable.

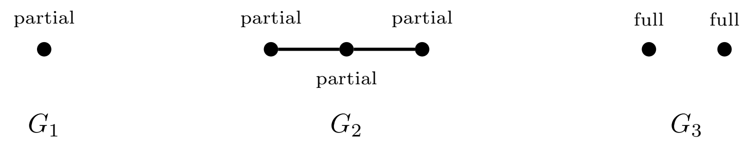


Figure 8.2: Marked graphs for Example 8.2.8.

Furthermore, if a GSF-local property $\mathcal{P}$ is not definable in FO then the sequence used to define $\mathcal{P}$ must be non-stabilising. More precisely, if a property $\mathcal{P}$ is $\overline{\mathcal{F}}$-local for a sequence of marked graphs $\overline{\mathcal{F}} = (\mathcal{F}_n)_{n\in\mathbb{N}}$ and there is $n_0 \in \mathbb{N}$ such that $\mathcal{F}_n = \mathcal{F}_{n+1}$ for every $n \geq n_0$ then $\mathcal{P}$ is FO definable. That is as up to n_0 we can express in FO exactly which graphs are in $\mathcal{P}$ and we can express in FO that every graph with at least n_0 vertices is $\mathcal{F}_n$-free.

Let us remark that Theorem 8.2.5 combined with Lemma 8.2.3 proves that every finite union of properties definable by 0-profiles is both FO definable and GSF-local. Hence it is natural to ask whether the intersection of FO definable properties and GSF-local properties is precisely the set of finite unions of properties definable by 0-profiles. However, this is not the case. The following example shows that there are properties which are both FO definable and GSF-local but cannot be expressed by 0-profiles.

Example 8.2.8. Let $d \geq 2$ and let $B_1 := (\{v\}, \{\})$, $B_2 = (\{v, w\}, \{\{v, w\}\})$ be two graphs. We further let τ_1, τ_2 be the 1-types of degree d such that $(B_1, v) \in \tau_1$ and $(B_2, v) \in \tau_2$. Consider the property $\mathcal{P}$ defined by the following FO-formula

$$\varphi := \neg\exists x(x = x) \vee \exists^{=1}x\Big(\varphi_{\tau_1}(x) \wedge \forall y\big(x \neq y \to \varphi_{\tau_2}(y)\big)\Big).$$

$\mathcal{P}$ contains, besides the empty graph, unions of an arbitrary amount of disjoint edges and one isolated vertex. To define a sequence of forbidden subgraphs we let G_1, G_2, G_3 be the marked graphs in Figure 8.2. Let $\mathcal{F}_{\text{even}} := \{G_1\}$ and $\mathcal{F}_{\text{odd}} := \{G_2, G_3\}$ and let $\overline{\mathcal{F}} = (\mathcal{F}_n)_{n\in\mathbb{N}}$ where $\mathcal{F}_i = \mathcal{F}_{\text{even}}$ if i is even and $\mathcal{F}_i = \mathcal{F}_{\text{odd}}$ if i is odd. Note that every graph on more than one vertex with an odd number of vertices which is $\mathcal{F}_{\text{odd}}$-free must contain a vertex of neighbourhood type τ_1, and that the set of $\mathcal{F}_{\text{even}}$-free graphs contains only the empty graph. Hence $\mathcal{P}$ is $\overline{\mathcal{F}}$-local. Now assume towards a contradiction that $\mathcal{P} = \bigcup_{1\leq i\leq k} \mathcal{P}_{\rho_i}$ for 0-profiles ρ_i. Let G_m be the graph consisting of m disjoint edges and one isolated vertex and H_m the graph consisting of m disjoint edges. Since $G_m \in \mathcal{P}$ there is $i \in \{1, \ldots, k\}$ such that $G_m \sim \rho_i$. By choice of G_m and H_m we have $0 \leq (\text{hist}_r(H_m))_j \leq (\text{hist}_r(G_m))_j \in \rho_i(j)$ for every $j \in \{1, \ldots, t\}$, where t is the number of 1-types of bounded degree d. Since additionally $\rho_i(j) \in \mathfrak{I}_0$ this implies that

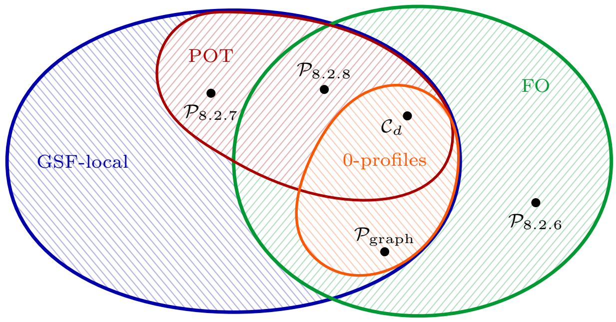


Figure 8.3: Overview of all relevant classes of properties. Here $\mathcal{P}_i$ refers to the property from Example i, $\mathcal{C}_d$ refers to the property of all graphs of bounded degree d and $\mathcal{P}_{\mathrm{graph}}$ is the property defined in Section 8.3.2.

$(\mathrm{hist}_r(H_m))_j \in \rho_i(j)$. But then $H_m \sim \rho_i$ which yields a contradiction as $H_m \notin \mathcal{P}$. Hence $\mathcal{P}$ can not be defined as a finite union of 0-profiles.

Figure 8.3 gives a schematic overview of all classes of properties discussed here and their relationship.

8.3 GSF-locality is not sufficient for proximity-oblivious testing

In this section we prove Theorem 8.0.1. We start by describing a property of relational structures, similar to a property in Section 6, which is not testable. We then show that the property can be expressed by a union of 0-profiles. This will be used later to show that a certain graph property, which we obtain from the relational structure property by a local reduction, is GSF-local using Theorem 8.2.5.

Let σ be the signature from Section 6.1.2 Equation 6.1, $d \in \mathbb{N}$ as in Section 6.1.3 and $P_{Ⓩ}$ be the property of σ-structures of bounded degree d from Section 6.2.

8.3.1 Characterisation by neighbourhood profiles

Our aim in this section is to prove that a minor variation of the property $P_{Ⓩ}$ of relational structures can be written as a finite union of properties defined by 0-profiles of radius 2. As the existence of a certain vertex cannot be expressed with a 0-profile (see Observation 8.2.4) and $\varphi_{Ⓩ}$ demands the existence of a certain vertex (the root vertex), the property $P_{Ⓩ}$ cannot be expressed in terms of 0-profiles. However we define a slight variation of the formula $\varphi_{Ⓩ}$ which, as we will see later, can be expressed by 0-profiles. Let

$$\varphi'_{Ⓩ} := \varphi'_{\mathrm{tree}} \wedge \varphi_{\mathrm{rotationMap}} \wedge \varphi_{\mathrm{base}} \wedge \varphi_{\mathrm{recursion}},$$

where we obtain φ'_{tree} from φ_{tree} by replacing the subformula $\exists^{=1}x\varphi_{\text{root}}(x)$ by $\exists^{\leq 1}x\varphi_{\text{root}}(x)$, where $\varphi_{\text{root}}(x)$, φ_{tree}, $\varphi_{\text{rotationMap}}$, φ_{base} and $\varphi_{\text{recursion}}$ are the formulas given in Section 6.1.2 Equations 6.2, 6.3, 6.4, 6.5 and 6.6. We define the property

$$P'_{Ⓩ} := \{\mathcal{A} \in C_d \mid \mathcal{A} \models \varphi'_{Ⓩ}\}.$$

We denote the empty structure by $\mathcal{A}_\emptyset$ (i. e. $A_\emptyset = \emptyset$).

Lemma 8.3.1. *The properties* $P'_{Ⓩ}$ *and* $P_{Ⓩ} \cup \{\mathcal{A}_\emptyset\}$ *are equal.*

Proof. We first prove that $P'_{Ⓩ} \subseteq P_{Ⓩ} \cup \{\mathcal{A}_\emptyset\}$. Consider the formula $\tilde{\varphi}_{Ⓩ}$ which is obtained from $\varphi_{Ⓩ}$ by removing the subformula $\exists^{=1}x\varphi_{\text{root}}(x)$. We use the following observation which is proved by a simple analysis of the formula.

Claim 1. Satisfying $\tilde{\varphi}_{Ⓩ}$ is closed under disjoint unions on C_d.

Proof of Claim 1. Let $\mathcal{A}, \mathcal{A}' \in C_d$ such that $\mathcal{A} \models \tilde{\varphi}_{Ⓩ}$ and $\mathcal{A}' \models \tilde{\varphi}_{Ⓩ}$. Our aim is to prove that $\mathcal{A} \sqcup \mathcal{A}' \models \tilde{\varphi}_{Ⓩ}$ where $\mathcal{A} \sqcup \mathcal{A}'$ denotes the disjoint union of $\mathcal{A}$ and $\mathcal{A}'$. For this we essentially prove that for any two elements $a \in A$ and $b \in A'$ the formula $\tilde{\varphi}_{Ⓩ}$ does not require a tuple containing a and b.

Let us define the following subformulas of φ_{tree} (see Section 6.1.2 Equation 6.3)

$$\varphi := \forall x\Big(\big(\varphi_{\text{root}}(x) \wedge R(x,x)\big) \vee \big(\exists^{=1}yF(y,x) \wedge \neg\exists yR(x,y) \wedge \neg\exists yR(y,x)\big)\Big),$$

$$\psi(x) := \neg\exists yF(x,y) \wedge \bigwedge_{k\in([D]^2)^2} L_k(x,x) \wedge \forall y\Big(y \neq x \to \bigwedge_{k\in([D]^2)^2} \neg L_k(x,y) \wedge \bigwedge_{k\in([D]^2)^2} \neg L_k(y,x)\Big) \text{ and}$$

$$\chi(x) := \neg\exists y \bigvee_{k\in([D]^2)^2} \big(L_k(x,y) \vee L_k(y,x)\big)\wedge$$

$$\bigwedge_{k\in([D]^2)^2} \exists y_k\Big(x \neq y_k \wedge F_k(x,y_k) \wedge (\bigwedge_{k'\in([D]^2)^2, k'\neq k} \neg F_{k'}(x,y_k)) \wedge \forall y(y \neq y_k \to \neg F_k(x,y))\Big).$$

Then $\tilde{\varphi}_{Ⓩ} := \varphi \wedge \forall x(\psi(x) \vee \chi(x)) \wedge \varphi_{\text{rotationMap}} \wedge \varphi_{\text{base}} \wedge \varphi_{\text{recursion}}$. Hence it is sufficient to prove that $\mathcal{A} \sqcup \mathcal{A}' \models \varphi$, $\mathcal{A} \sqcup \mathcal{A}' \models \forall x(\psi(x) \vee \chi(x))$, $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{rotationMap}}$, $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{base}}$ and $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{recursion}}$.

We first argue that $\mathcal{A} \sqcup \mathcal{A}' \models \varphi$. Let $a \in A \sqcup A'$ be arbitrary and assume without loss of generality that $a \in A$. Assume that $\mathcal{A} \sqcup \mathcal{A}' \not\models \varphi_{\text{root}}(a) \wedge R(a,a)$. Since $\varphi_{\text{root}}(x) := \forall y \neg F(y,x)$ this implies that $\mathcal{A} \not\models \varphi_{\text{root}}(a) \wedge R(a,a)$. Furthermore, since $\mathcal{A} \models \varphi$ we get that $\mathcal{A} \models \exists^{=1}yF(y,a) \wedge \neg\exists yR(a,y) \wedge \neg\exists yR(y,a)$. Hence there is an element $b \in A$ such that $(b,a) \in F^{\mathcal{A}}$. Furthermore, for every $b' \in A$, $b' \neq b$ we have $(b',a) \notin F^{\mathcal{A}}$, $(a,b') \notin R^{\mathcal{A}}$ and $(b',a) \notin R^{\mathcal{A}}$. But because a cannot be in a tuple with any element in A' we get that

$\mathcal{A} \sqcup \mathcal{A}' \models \exists^{=1} y F(y,a) \wedge \neg\exists y R(a,y) \wedge \neg\exists y R(y,a)$. Hence $\mathcal{A} \sqcup \mathcal{A}' \models \varphi$.

Next we prove that $\mathcal{A} \sqcup \mathcal{A}' \models \forall x(\psi(x) \vee \chi(x))$. Let $a \in A \sqcup A'$ be arbitrary and assume without loss of generality that $a \in A$. First assume that $(a,b) \notin F^{\mathcal{A}\sqcup\mathcal{A}'}$ for every $b \in A \sqcup A'$. Since $\mathcal{A}$ is an induced substructure of $\mathcal{A} \sqcup \mathcal{A}'$ this means that $\mathcal{A} \models \neg\exists y F(a,y)$. But then $\mathcal{A} \not\models \bigwedge_{k\in([D]^2)^2} \exists y_k\Big(a \neq y_k \wedge F_k(a,y_k)\Big)$ which implies $\mathcal{A} \not\models \chi(a)$. Since $\mathcal{A} \models \forall x(\psi(x) \vee \chi(x))$ this implies that $\mathcal{A} \models \psi(a)$. Hence for every $k \in ([D]^2)^2$ we have $(a,a) \in L_k^{\mathcal{A}}$ and for every $b \in A$, $b \neq a$ we have $(a,b),(b,a) \notin F_k^{\mathcal{A}}$. Since there are no tuples containing both elements from $\mathcal{A}$ and $\mathcal{A}'$ this directly implies that $\mathcal{A} \sqcup \mathcal{A}' \models \psi(a)$.

On the other hand, assume that there is $b \in A \sqcup A'$ such that $(a,b) \in F^{\mathcal{A}\sqcup\mathcal{A}'}$. Since we are considering the disjoint union of $\mathcal{A}$ and $\mathcal{A}'$ this implies that b must be an element from $\mathcal{A}$. Hence $\mathcal{A} \not\models \psi(a)$. Since $\mathcal{A} \models \forall x(\psi(x) \vee \chi(x))$ this implies that $\mathcal{A} \models \chi(a)$. Then for every $k \in ([D]^2)^2$ there is an element $b \in A$ such that $(a,b) \in F_k^{\mathcal{A}}$, $(a,b) \notin F_{k'}^{\mathcal{A}}$ for every $k' \in ([D]^2)^2$, $k' \neq k$ and $(a,b') \notin F_k^{\mathcal{A}}$ for every $b' \in A$, $b' \neq b$. But since in $\mathcal{A} \sqcup \mathcal{A}'$ there are no tuples containing both elements from $\mathcal{A}$ and $\mathcal{A}'$ this implies that $\mathcal{A} \sqcup \mathcal{A}' \models \chi(a)$. In conclusion we proved that $\mathcal{A} \sqcup \mathcal{A}' \models \forall x(\psi(x) \vee \chi(x))$.

We now prove $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{rotationMap}}$. Hence assume $a,b \in A \sqcup A'$ are arbitrary elements. First consider the case that a,b are either both from A or both from A'. In this case, if for some $i,j \in [D]^2$ we have that $(a,b) \in E_{i,j}^{\mathcal{A}\sqcup\mathcal{A}'}$ then $(b,a) \in E_{j,i}^{\mathcal{A}\sqcup\mathcal{A}'}$ because $\mathcal{A} \models \varphi_{\text{rotationMap}}$ and $\mathcal{A}' \models \varphi_{\text{rotationMap}}$. Now consider the case that $|\{a,b\} \cap A| = 1$. Then $(a,b) \notin E_{i,j}^{\mathcal{A}\sqcup\mathcal{A}'}$ and $(b,a) \notin E_{j,i}^{\mathcal{A}\sqcup\mathcal{A}'}$ and hence we get $\mathcal{A} \sqcup \mathcal{A}' \models \bigwedge_{i,j\in[D]^2}(E_{i,j}(a,b) \to E_{j,i}(b,a))$. Therefore $\mathcal{A} \sqcup \mathcal{A}' \models \forall x \forall y\Big(\bigwedge_{i,j\in[D]^2}(E_{i,j}(x,y) \to E_{j,i}(y,x))\Big)$.

Now consider an arbitrary element $a \in A \sqcup A'$ and any $i \in [D]^2$. Without loss of generality assume $a \in A$. Since $\mathcal{A} \models \varphi_{\text{rotationMap}}$ there must be an index $j \in [D]^2$ and an element $b \in A$ such that $(a,b) \in E_{i,j}^{\mathcal{A}}$. Furthermore, for every $b' \in A$, $b' \neq b$ we have $(a,b') \notin E_{i,j}^{\mathcal{A}}$ and for every $j' \in [D]^2$, $j' \neq j$ and every $\tilde{b} \in A$ we have $(a,\tilde{b}) \notin E_{i,j'}^{\mathcal{A}}$. But since $a \in A$ it also holds that $(a,b') \notin E_{i,j'}^{\mathcal{A}}$ for every $b' \in A'$ and every $j' \in [D]^2$. Hence $\mathcal{A} \sqcup \mathcal{A}' \models \bigvee_{j\in[D]^2}\big(\exists^{=1} y E_{i,j}(a,y) \wedge \bigwedge_{\substack{j'\in[D]^2\\ j'\neq j}} \neg\exists y E_{i,j'}(a,y)\big)$. This concludes the proof of $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{rotationMap}}$.

We now prove $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{base}}$. Assume $a \in A \sqcup A'$ is an arbitrary element such that $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{root}}(a)$. Without loss of generality assume $a \in A$. Since $\varphi_{\text{root}}(x) := \forall y \neg F(y,x)$ and $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{root}}(a)$ we get that $\mathcal{A} \models \varphi_{\text{root}}(a)$. Since $\mathcal{A} \models \varphi_{\text{base}}$ this means that for every $i,j \in [D]^2$ we have $(a,a) \in E_{i,j}^{\mathcal{A}}$ and $(a,b),(b,a) \notin E_{i,j}^{\mathcal{A}}$ for every $b \in A$, $b \neq a$. Since further $(a,b),(b,a) \notin E_{i,j}^{\mathcal{A}\sqcup\mathcal{A}'}$ for every $b \in A'$ this implies that $\mathcal{A} \sqcup \mathcal{A}' \models \bigwedge_{i,j\in[D]^2}\Big(E_{i,j}(a,a) \wedge \forall y\Big(a \neq y \to \big(\neg E_{i,j}(a,y) \wedge \neg E_{i,j}(y,a)\big)\Big)\Big)$. Furthermore, since $\mathcal{A} \models \varphi_{\text{base}}$ and $\mathcal{A} \models \varphi_{\text{root}}(a)$ for every $k,k' \in ([D]^2)^2, i,i' \in [D]^2$ for which $\mathrm{ROT}_{H^2}(k,i) = (k',i')$ there are $b,b' \in A$ such that

$(a,b) \in F_k^{\mathcal{A}}$, $(a,b') \in F_{k'}^{\mathcal{A}}$ and $(b,b') \in E_{i,i'}^{\mathcal{A}}$. Since $\mathcal{A}$ is a substructure of $\mathcal{A} \sqcup \mathcal{A}'$ this proves that $\mathcal{A} \sqcup \mathcal{A} \models \varphi_{\text{base}}$.

Finally we prove $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{recursion}}$. Hence assume $a, c \in A \sqcup A'$ are arbitrary elements. Assume $\mathcal{A} \sqcup \mathcal{A}' \not\models \neg\exists y F(a,y) \wedge \neg\exists y F(c,y)$ and assume without loss of generality that there is $\tilde{a} \in A \sqcup A'$ such that $(a,\tilde{a}) \in F^{\mathcal{A}\sqcup\mathcal{A}'}$. Since there are no tuples containing both elements from $\mathcal{A}$ and $\mathcal{A}'$ we get that $a, \tilde{a}$ are from the same structure. Without loss of generality assume $a, \tilde{a} \in A$. Assume that for indices $k_1', k_2' \in [D]^2, \ell_1', \ell_2' \in [D]^2$ and some element $b \in A \sqcup A'$ we have $(a,b) \in E_{k_1',\ell_1'}^{\mathcal{A}\sqcup\mathcal{A}'}$ and $(b,c) \in E_{k_2',\ell_2'}^{\mathcal{A}\sqcup\mathcal{A}'}$. As b also has to be in A and $\mathcal{A} \models \varphi_{\text{recursion}}$ this implies that for every $i, j, i', j' \in [D], k, \ell \in ([D]^2)^2$ for which $\text{ROT}_H(k,i) = ((k_1', k_2'), i')$ and $\text{ROT}_H((\ell_2', \ell_1'), j) = (\ell, j')$ there are elements $a', c' \in A \sqcup A'$ such that $(a,a') \in F_k^{\mathcal{A}\sqcup\mathcal{A}'}$, $(c,c') \in F_\ell^{\mathcal{A}\sqcup\mathcal{A}'}$ and $(a',c') \in E_{(i,j),(j',i')}^{\mathcal{A}\sqcup\mathcal{A}'}$. Hence $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{\text{recursion}}$. ■

Since $\mathcal{A}_\emptyset \in P_{Ⓩ} \cup \{\mathcal{A}_\emptyset\}$ it is sufficient to consider only non-empty structures in the following. Therefore assume that there exists $\mathcal{A} \in C_d$ with $A \neq \emptyset$ such that $\mathcal{A} \models \varphi'_{Ⓩ}$ and $\mathcal{A}$ contains no element a for which $\mathcal{A} \models \varphi_{\text{root}}(a)$. Let $\mathcal{A}' \in C_d$ be any model of $\varphi_{Ⓩ}$ with $A \cap A' = \emptyset$. Then $\mathcal{A} \sqcup \mathcal{A}' \models \tilde{\varphi}_{Ⓩ}$ by Claim 1. Furthermore, $\mathcal{A} \sqcup \mathcal{A}' \models \exists^{=1} x \varphi_{\text{root}}(x)$, which implies $\mathcal{A} \sqcup \mathcal{A}' \models \varphi_{Ⓩ}$. By construction, the Gaifman graph $G((\mathcal{A} \sqcup \mathcal{A}')|_F)$ of the structure $\mathcal{A}|_F := (A, (F_k^{\mathcal{A}})_{k\in([D]^2)^2})$ has more than one connected component as both $A \neq \emptyset$ and $A' \neq \emptyset$ and $\mathcal{A} \sqcup \mathcal{A}'$ is a disjoint union of $\mathcal{A}$ and $\mathcal{A}'$. Hence we obtain a contradiction to Lemma 6.1.14. Therefore every non-empty structure satisfying $\varphi'_{Ⓩ}$ must satisfy $\exists^{=1} x \varphi_{\text{root}}(x)$, and hence also $\varphi_{Ⓩ}$.

Conversely, if $\mathcal{A} \in C_d$ is a model of $\varphi_{Ⓩ}$ then $\mathcal{A} \models \exists^{=1} x \varphi_{\text{root}}(x)$. This implies directly that $\mathcal{A} \models \exists^{\leq 1} x \varphi_{\text{root}}(x)$ and hence $\mathcal{A} \models \varphi'_{Ⓩ}$. Furthermore, $\mathcal{A}_\emptyset \in P'_{Ⓩ}$ as $\mathcal{A}_\emptyset \models \exists^{\leq 1} x \varphi_{\text{root}}(x)$ and $\mathcal{A}_\emptyset \models \tilde{\varphi}_{Ⓩ}$ as $\tilde{\varphi}_{Ⓩ}$ is a conjunction of formulas of the form $\forall x \psi(x)$ for some formula $\psi(x)$. Hence $P_{Ⓩ} \cup \{\mathcal{A}_\emptyset\} \subseteq P'_{Ⓩ}$. □

We now define the 0-profiles which express the property $P'_{Ⓩ}$. For all σ-structures in $P_{Ⓩ}$ (all σ-structure in $P'_{Ⓩ}$ but $\mathcal{A}_\emptyset$) it is crucial that they are allowed to contain precisely one root element. Hence the neighbourhood profile describing $P'_{Ⓩ}$ must restrict the number of occurrences of the 2-type of the root element. But since in $P_{Ⓩ}$, the root elements in different structures may have different 2-types, we partition $P_{Ⓩ}$ into parts $P_1, \ldots, P_m$ by the 2-type of the root element. Note that the number m of parts is constant as there are a constant number of 2-types in total. For each of these parts we then define a neighbourhood profile ρ_k such that $P_k \cup \{\mathcal{A}_\emptyset\} = P_{\rho_k}$. We would like to remark here that the roots of all but one structure in $P_{Ⓩ}$ actually have the same 2-type. Hence the partition only contains two parts and one of the two parts only contains one structure. We now define the parts and corresponding profiles formally.

Let $\tau_1, \ldots, \tau_t$ be a list of all 2-types of bounded degree d. Assume without loss of generality that the 2-types $\tau_1, \ldots, \tau_t$ are ordered in such a way that for $(\mathcal{B}, b) \in \tau_k$, it holds that $\mathcal{B} \models$

$\varphi_{\text{root}}(b)$ if and only if $k \in \{1, \dots, m\}$ for some $m \le t$. For $k \in \{1, \dots, m\}$, let

$$P_k := \{\mathcal{A} \in P_{\textcircled{z}} \mid \text{ there is } a \in A \text{ such that } (\mathcal{N}_2^{\mathcal{A}}(a), a) \in \tau_k\}.$$

Since every $\mathcal{A} \in P_{\textcircled{z}}$ satisfies $\exists^{=1} x \varphi_{\text{root}}(x)$ we get that

$$P'_{\textcircled{z}} = \bigcup_{1 \le k \le m} P_k \cup \{\mathcal{A}_\emptyset\}$$

and this union is disjoint. Furthermore, for $k \in \{1, \dots, m\}$, let $I_k \subseteq \{1, \dots, t\}$ be the set of indices j such that there is a structure $\mathcal{A} \in P_k$ and $a \in A$ with $(\mathcal{N}_2^{\mathcal{A}}(a), a) \in \tau_j$. For every $k \in \{1, \dots, m\}$ we define the 2-neighbourhood profile $\rho_k : \{1, \dots, t\} \to \mathfrak{I}_0$ by

$$\rho_k(i) := \begin{cases} [0,1] & \text{if } i = k, \\ [0,\infty) & \text{if } i \in I_k \setminus \{k\}, \\ [0,0] & \text{otherwise.} \end{cases}$$

To prove that these 0-profiles of radius 2 define the property $P'_{\textcircled{z}}$, the crucial observation is that for every element a of some structure in C_d, the FO-formula $\varphi'_{\textcircled{z}}$ only talks about elements of distance at most 2 to a (i. e. $\varphi'_{\textcircled{z}}$ is 2-local). Hence the 2-histogram vector of a structure already captures whether the structure satisfies $\varphi'_{\textcircled{z}}$. We will now formally prove this.

Lemma 8.3.2. *It holds that* $P'_{\textcircled{z}} = \bigcup_{1 \le k \le m} P_{\rho_k}$.

Proof. We first prove that $P'_{\textcircled{z}} \subseteq \bigcup_{1 \le k \le m} P_{\rho_k}$. First note that trivially $\mathcal{A}_\emptyset \in \bigcup_{1 \le k \le m} P_{\rho_k}$. Now assume $\mathcal{A} \in P_{\textcircled{z}}$. This implies that there is $k \in \{1, \dots, m\}$ such that $\mathcal{A} \in P_k$. By construction we have that for every $a \in \mathcal{A}$, there is $i \in I_k$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \in \tau_i$. Furthermore, since $\mathcal{A} \models \varphi_{\textcircled{z}}$, we have that $\mathcal{A} \models \exists^{=1} x \varphi_{\text{root}}(x)$, and that there can be at most one $a \in A$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \in \tau_k$. Therefore $\mathcal{A} \in P_{\rho_k}$.

To prove $\bigcup_{1 \le k \le m} P_{\rho_k} \subseteq P'_{\textcircled{z}}$, we prove that every structure in $\bigcup_{1 \le k \le m} P_{\rho_k}$ must satisfy $\varphi'_{\textcircled{z}}$. We will prove that every $\mathcal{A} \in \bigcup_{1 \le k \le m} P_{\rho_k}$ satisfies $\varphi'_{\textcircled{z}}$ in the following four claims. Note that $\mathcal{A}_\emptyset \models \varphi'_{\textcircled{z}}$ by Lemma 8.3.1 and hence we exclude $\mathcal{A}_\emptyset$ in the following.

Claim 1. Every structure $\mathcal{A} \in \bigcup_{1 \le k \le m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$ satisfies φ'_{tree}.

Proof of Claim 1. Let $\mathcal{A} \in \bigcup_{1 \le k \le m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$. Then there is $k \in \{1, \dots, m\}$ such that $\mathcal{A} \in P_{\rho_k}$.

By definition (see Section 6.1.2 Equation 6.3), $\varphi'_{\text{tree}} := \exists^{\le 1} x \varphi_{\text{root}}(x) \wedge \varphi \wedge \forall x(\psi(x) \vee \chi(x))$, where

$$\varphi := \forall x\Big(\big(\varphi_{\text{root}}(x) \wedge R(x,x)\big) \vee \big(\exists^{=1} y F(y,x) \wedge \neg\exists y R(x,y) \wedge \neg\exists y R(y,x)\big)\Big),$$

$$\psi(x) := \neg\exists y F(x,y) \wedge \bigwedge_{k\in([D]^2)^2} L_k(x,x) \wedge \forall y\Big(y \neq x \to \bigwedge_{k\in([D]^2)^2} \neg L_k(x,y) \wedge \bigwedge_{k\in([D]^2)^2} \neg L_k(y,x)\Big) \text{ and}$$

$$\chi(x) := \neg\exists y \bigvee_{k\in([D]^2)^2} \big(L_k(x,y) \vee L_k(y,x)\big) \wedge$$

$$\bigwedge_{k\in([D]^2)^2} \exists y_k\Big(x \neq y_k \wedge F_k(x,y_k) \wedge \big(\bigwedge_{k'\in([D]^2)^2, k'\neq k} \neg F_{k'}(x,y_k)\big) \wedge \forall y(y \neq y_k \to \neg F_k(x,y))\Big).$$

Thus, it is sufficient to prove that $\mathcal{A} \models \exists^{\leq 1} x\varphi_{\text{root}}(x)$, $\mathcal{A} \models \varphi$ and $\mathcal{A} \models \forall x(\psi(x) \vee \chi(x))$.

To prove $\mathcal{A} \models \exists^{\leq 1} x\varphi_{\text{root}}(x)$ we note that by construction of ρ_k we have $\mathcal{A} \not\models \varphi_{\text{root}}(a)$ for any $a \in A$ for which $(\mathcal{N}_2^{\mathcal{A}}(a), a) \notin \tau_k$. Since ρ_k restricts the number of occurrences of elements of neighbourhood type τ_k to at most one, this proves that there is at most one $a \in A$ with $\mathcal{A} \models \varphi_{\text{tree}}(a)$ and hence $\mathcal{A} \models \exists^{\leq 1} x\varphi_{\text{root}}(x)$.

To prove $\mathcal{A} \models \varphi$, let $a \in A$ be an arbitrary element. Since $\mathcal{A} \in P_{\rho_k}$, there is an $i \in I_k$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \in \tau_i$. But then by definition, there exist $\tilde{\mathcal{A}} \models \varphi_{Z}$ and $\tilde{a} \in \tilde{A}$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \cong (\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Assume f is an isomorphism from $(\mathcal{N}_2^{\mathcal{A}}(a), a)$ to $(\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. First consider the case that $\mathcal{A} \models \varphi_{\text{root}}(a) := \forall y \neg F(y,a)$. Assume there is $\tilde{b} \in \tilde{A}$ such that $(\tilde{b}, \tilde{a}) \in F^{\tilde{\mathcal{A}}}$. Since $\tilde{b} \in N_2^{\tilde{\mathcal{A}}}(\tilde{a})$, there must be an element $b \in N_2^{\mathcal{A}}(a)$ such that $f(b) = \tilde{b}$. Since f is an isomorphism mapping a to $\tilde{a}$, this implies $(b,a) \in F^{\mathcal{A}}$, which contradicts $\mathcal{A} \models \varphi_{\text{root}}(a)$. Hence $\tilde{\mathcal{A}} \models \varphi_{\text{root}}(\tilde{a})$. Since $\tilde{\mathcal{A}} \models \varphi'_{\text{tree}}$, it holds that $\tilde{\mathcal{A}} \models \varphi$, which means that $(\tilde{a}, \tilde{a}) \in R^{\tilde{\mathcal{A}}}$. But since f is an isomorphism mapping a onto $\tilde{a}$, this implies $(a,a) \in R^{\mathcal{A}}$. Now consider the case that $\mathcal{A} \not\models \varphi_{\text{root}}(a)$. Then there is $b \in A$ with $(b,a) \in F^{\mathcal{A}}$. Since f is an isomorphism, this implies $(f(b), \tilde{a}) \in F^{\tilde{\mathcal{A}}}$. Hence $\tilde{\mathcal{A}} \models \exists^{=1} y F(y,\tilde{a}) \wedge \neg\exists y R(\tilde{a},y) \wedge \neg\exists y R(y,\tilde{a})$, as $\tilde{\mathcal{A}} \models \varphi$. Now assume that there is $b' \neq b$ such that $(b',a) \in F^{\mathcal{A}}$. Then $f(b) \neq f(b')$ and $(f(b),\tilde{a}), (f(b'),\tilde{a}) \in F^{\tilde{\mathcal{A}}}$. Since this contradicts $\tilde{\mathcal{A}} \models \exists^{=1} y F(y,\tilde{a})$ we have $\mathcal{A} \models \exists^{=1} y F(y,a)$. Furthermore, assume that there is $b' \in A$ such that either $(a,b') \in R^{\mathcal{A}}$ or $(b',a) \in R^{\mathcal{A}}$. Then either $(\tilde{a}, f(b')) \in R^{\tilde{\mathcal{A}}'}$ or $(f(b'), \tilde{a}) \in R^{\tilde{\mathcal{A}}}$, which contradicts $\tilde{\mathcal{A}} \models \neg\exists y R(\tilde{a},y) \wedge \neg\exists y R(y,\tilde{a})$. Therefore $\mathcal{A} \models \neg\exists y R(a,y) \wedge \neg\exists y R(y,a)$ which completes the proof of $\mathcal{A} \models \varphi$.

We prove $\mathcal{A} \models \forall x(\psi(x) \vee \chi(x))$ by considering the two cases $\mathcal{A} \models \neg\exists y F(a,y)$ and $\mathcal{A} \models \exists y F(a,y)$ for each element $a \in A$. For this, let $a \in A$ be any element. By the construction of ρ_k there is $\tilde{\mathcal{A}} \models \varphi_{Z}$ and $\tilde{a} \in \tilde{A}$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \cong (\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Let f be an isomorphism from $(\mathcal{N}_2^{\mathcal{A}}(a), a)$ to $(\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. First consider the case that $\mathcal{A} \models \neg\exists y F(a,y)$. If there was $\tilde{b} \in \tilde{A}$ with $(\tilde{a}, \tilde{b}) \in F^{\tilde{\mathcal{A}}}$ then $(a, f^{-1}(\tilde{b})) \in F^{\mathcal{A}}$ contradicting our assumption. Hence $\tilde{\mathcal{A}} \models \neg\exists y F(\tilde{a},y)$ which implies that $\tilde{\mathcal{A}} \not\models \chi(\tilde{a})$. But since $\tilde{\mathcal{A}} \models \varphi_{Z}$, it holds that $\tilde{\mathcal{A}} \models \forall x(\psi(x) \vee \chi(x))$, which implies that $\tilde{\mathcal{A}} \models \psi(\tilde{a})$. Hence $(\tilde{a}, \tilde{a}) \in L_k^{\tilde{\mathcal{A}}}$ for every $k \in ([D]^2)^2$. Since f is an isomorphism and $f(a) = \tilde{a}$, it holds that $(a,a) \in L_k^{\mathcal{A}}$ for every $k \in ([D]^2)^2$, and hence $\mathcal{A} \models \bigwedge_{k\in([D]^2)^2} L_k(a,a)$. Furthermore, assume that there is $b \in A$, $b \neq a$ and $k \in ([D]^2)^2$ such that either $(a,b) \in L_k^{\mathcal{A}}$ or $(b,a) \in L_k^{\mathcal{A}}$. Since f is an isomorphism this implies that either $(\tilde{a}, f(b)) \in L_k^{\tilde{\mathcal{A}}}$ or $(f(b), \tilde{a}) \in L_k^{\tilde{\mathcal{A}}}$ which contradicts $\tilde{\mathcal{A}} \models \chi(\tilde{a})$. Hence $\mathcal{A} \models \forall y\Big(y \neq a \rightarrow \bigwedge_{k\in([D]^2)^2} \neg L_k(a,y) \wedge \bigwedge_{k\in([D]^2)^2} \neg L_k(y,a)\Big)$ proving that $\mathcal{A} \models \psi(a)$.

Now consider the case that there is an element $b \in A$ such that $(a,b) \in F^{\mathcal{A}}$. Since this implies that $(\tilde{a}, f(b)) \in F^{\tilde{\mathcal{A}}}$, we get that $\tilde{\mathcal{A}} \not\models \psi(\tilde{a})$, and hence $\tilde{\mathcal{A}} \models \chi(\tilde{a})$. Now assume that there is a $b \in A$ and $k \in ([D]^2)^2$ such that either $(a,b) \in L_k^{\mathcal{A}}$ or $(b,a) \in L_k^{\mathcal{A}}$. But then either $(\tilde{a}, f(b)) \in L_k^{\tilde{\mathcal{A}}}$ or $(f(b), \tilde{a}) \in L_k^{\tilde{\mathcal{A}}}$, which contradicts $\tilde{\mathcal{A}} \models \chi(\tilde{a})$. Hence $\mathcal{A} \models \neg\exists y \bigvee_{k\in([D]^2)^2} \big(L_k(a,y) \vee L_k(y,a)\big)$. For each $k \in ([D]^2)^2$, let $\tilde{b}_k \in \tilde{A}$ be an element such that $\tilde{\mathcal{A}} \models \tilde{a} \neq \tilde{b}_k \wedge F_k(\tilde{a}, \tilde{b}_k) \wedge (\bigwedge_{k'\in([D]^2)^2, k'\neq k} \neg F_{k'}(\tilde{a}, \tilde{b}_k)) \wedge \forall y(y \neq \tilde{b}_k \rightarrow \neg F_k(\tilde{a}, y))$. Since f is an isomorphism, this implies that $a \neq b_k := f^{-1}(\tilde{b}_k)$, $(a, b_k) \in F_k^{\mathcal{A}}$ and $(a, b_k) \notin F_{k'}^{\mathcal{A}}$, for each $k' \in ([D]^2)^2, k' \neq k$. Furthermore, assume there is $b \in A$, $b \neq b_k$ such that $(a,b) \in F_k^{\mathcal{A}}$. Since f is an isomorphism, this implies $f(b) \neq f(b_k) = \tilde{b}_k$ and $(\tilde{a}, f(b)) \in F_k^{\tilde{\mathcal{A}}}$, which contradicts $\tilde{\mathcal{A}} \models \forall y(y \neq \tilde{b}_k \rightarrow \neg F_k(\tilde{a}, y))$. Hence $\mathcal{A} \models \forall y(y \neq b_k \rightarrow \neg F_k(a,y))$ and therefore concluding that $\mathcal{A} \models \chi(a)$. This proves that in either case $\mathcal{A} \models \psi(a) \vee \chi(a)$ and therefore $\mathcal{A} \models \forall x(\psi(x) \vee \chi(x))$. ■

Claim 2. Every structure $\mathcal{A} \in \bigcup_{1\leq k\leq m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$ satisfies $\varphi_{\text{rotationMap}}$.

Proof of Claim 2. Let $\mathcal{A} \in \bigcup_{1\leq k\leq m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$. Then there is a $k \in \{1, \dots, m\}$ such that $\mathcal{A} \in P_{\rho_k}$.

By definition (Section 6.1.2 Equation 6.4), $\varphi_{\text{rotationMap}} = \varphi \wedge \psi$, where

$$\varphi := \forall x \forall y\Big(\bigwedge_{i,j\in[D]^2} (E_{i,j}(x,y) \rightarrow E_{j,i}(y,x))\Big) \text{ and}$$
$$\psi := \forall x\Big(\bigwedge_{i\in[D]^2}\Big(\bigvee_{j\in[D]^2}\big(\exists^{=1} y E_{i,j}(x,y) \wedge \bigwedge_{\substack{j'\in[D]^2\\ j'\neq j}} \neg\exists y E_{i,j'}(x,y)\big)\Big)\Big).$$

Thus, it is sufficient to prove that $\mathcal{A} \models \varphi$ and $\mathcal{A} \models \psi$.

To prove $\mathcal{A} \models \varphi$, assume towards a contradiction that there are $a, b \in A$ such that for some pair $i, j \in [D]^2$, we have that $(a,b) \in E_{i,j}^{\mathcal{A}}$, but $(b,a) \notin E_{j,i}^{\mathcal{A}}$. By construction of P_{ρ_k}, there is a structure $\tilde{\mathcal{A}} \models \varphi_{Ⓩ}$ and $\tilde{a} \in \tilde{A}$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \cong (\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Assume f is an isomorphism from $(\mathcal{N}_2^{\mathcal{A}}(a), a)$ to $(\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Note that $f(b)$ is defined since b is in the 2-neighbourhood of a. Furthermore since f is an isomorphism, $(a,b) \in E_{i,j}^{\mathcal{A}}$ implies $(\tilde{a}, f(b)) \in E_{i,j}^{\tilde{\mathcal{A}}}$, and $(b,a) \notin E_{j,i}^{\mathcal{A}}$ implies $(f(b), \tilde{a}) \notin E_{j,i}^{\tilde{\mathcal{A}}}$. Hence $\tilde{\mathcal{A}} \not\models \varphi$, which contradicts $\tilde{\mathcal{A}} \models \varphi_{\text{rotationMap}}$.

To prove $\mathcal{A} \models \psi$, assume towards a contradiction that there is an $a \in A$ and $i \in [D]^2$ such that $\mathcal{A} \not\models \exists^{=1} y E_{i,j}(a,y) \wedge \bigwedge_{\substack{j'\in[D]^2\\ j'\neq j}} \neg\exists y E_{i,j'}(a,y)$ for every $j \in [D]^2$. We know that there is a structure $\tilde{\mathcal{A}} \models \varphi_{Ⓩ}$ and $\tilde{a} \in \tilde{A}$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \cong (\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Let f be an isomorphism from $(\mathcal{N}_2^{\mathcal{A}}(a), a)$ to $(\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Since $\tilde{\mathcal{A}} \models \psi$, there must be $j \in [D]^2$ such that $\tilde{\mathcal{A}} \models \exists^{=1} y E_{i,j}(\tilde{a}, y) \wedge \bigwedge_{\substack{j'\in[D]^2\\ j'\neq j}} \neg\exists y E_{i,j'}(\tilde{a}, y)$. Hence there must be $\tilde{b} \in \tilde{A}$ such that

$(\tilde{a},\tilde{b}) \in E_{i,j}^{\tilde{\mathcal{A}}}$, which implies that $(a, f^{-1}(\tilde{b})) \in E_{i,j}^{\mathcal{A}}$. Since we assumed that $\mathcal{A} \not\models \exists^{=1} y E_{i,j}(a,y) \wedge \bigwedge_{\substack{j' \in [D]^2 \\ j' \neq j}} \neg\exists y E_{i,j'}(a,y)$, there must be either $b \neq f^{-1}(\tilde{b})$ with $(a,b) \in E_{i,j}^{\mathcal{A}}$, or there must be $j' \in [D]^2$, $j' \neq j$ and $b' \in A$ such that $(a,b') \in E_{i,j'}^{\mathcal{A}}$. In the first case $(\tilde{a}, f(b)) \in E_{i,j}^{\tilde{\mathcal{A}}}$, since f is an isomorphism. But then $\tilde{\mathcal{A}} \not\models \exists^{=1} y E_{i,j}(\tilde{a},y)$, which is a contradiction. In the second case, we get that $(\tilde{a}, f(b')) \in E_{i,j'}^{\tilde{\mathcal{A}}}$. But then $\tilde{\mathcal{A}} \not\models \bigwedge_{\substack{j' \in [D]^2 \\ j' \neq j}} \neg\exists y E_{i,j'}(\tilde{a},y)$, which is a contradiction. Hence $\mathcal{A} \models \varphi \wedge \psi$. ∎

Claim 3. Every structure $\mathcal{A} \in \bigcup_{1 \leq k \leq m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$ satisfies φ_{base}.

Proof of Claim 3. Let $\mathcal{A} \in \bigcup_{1 \leq k \leq m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$. Then there is a $k \in \{1,\dots,m\}$ such that $\mathcal{A} \in P_{\rho_k}$.

By definition (Section 6.1.2 Equation 6.5), $\varphi_{\text{base}} := \forall x \big(\varphi_{\text{root}}(x) \to (\varphi(x) \wedge \psi(x))\big)$, where

$$\varphi(x) := \bigwedge_{i,j \in [D]^2} \Big(E_{i,j}(x,x) \wedge \forall y \Big(x \neq y \to \big(\neg E_{i,j}(x,y) \wedge \neg E_{i,j}(y,x)\big)\Big)\Big) \text{ and}$$

$$\psi(x) := \bigwedge_{\substack{\mathrm{ROT}_{H^2}(k,i) = (k',i') \\ k,k' \in ([D]^2)^2 \\ i,i' \in [D]^2}} \exists y \exists y' \big(F_k(x,y) \wedge F_{k'}(x,y') \wedge E_{i,i'}(y,y')\big).$$

Thus, it is sufficient to prove that $\mathcal{A} \models \varphi(a)$ and $\mathcal{A} \models \psi(a)$ for every $a \in A$ for which $\mathcal{A} \models \varphi_{\text{root}}(a)$. Therefore assume $a \in A$ is any element such that $\mathcal{A} \models \varphi_{\text{root}}(a)$. Because $\mathcal{A} \in P_{\rho_k}$ there is an $i \in I_k$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \in \tau_i$. Then by definition there is a structure $\tilde{\mathcal{A}} \models \varphi_{Z}$ and $\tilde{a} \in \tilde{A}$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \cong (\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Let f be an isomorphism from $(\mathcal{N}_2^{\mathcal{A}}(a), a)$ to $(\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Assume that there is an element $\tilde{b} \in \tilde{A}$ such that $(\tilde{b}, \tilde{a}) \in F^{\tilde{\mathcal{A}}}$. Since f is an isomorphism and $\tilde{b} \in N_2^{\tilde{\mathcal{A}}}(\tilde{a})$ we get that $(f^{-1}(\tilde{b}), a) \in F^{\mathcal{A}}$ which contradicts that $\mathcal{A} \models \varphi_{\text{root}}(a)$ as $\varphi_{\text{root}}(x) := \forall y \neg F(y,x)$. Hence there is no element $\tilde{b} \in \tilde{A}$ such that $(\tilde{b}, \tilde{a}) \in F^{\tilde{\mathcal{A}}}$ which implies that $\tilde{\mathcal{A}} \models \varphi_{\text{root}}(\tilde{a})$. But since $\tilde{\mathcal{A}} \models \varphi_{Z}$ we have that $\tilde{\mathcal{A}} \models \varphi_{\text{base}}$ and hence $\tilde{\mathcal{A}} \models \varphi(\tilde{a})$ and $\tilde{\mathcal{A}} \models \psi(\tilde{a})$.

To prove $\mathcal{A} \models \varphi(a)$ first observe that $(a,a) \in E_{i,j}^{\mathcal{A}}$ for every $i,j \in [D]^2$ since $\tilde{\mathcal{A}} \models \varphi(\tilde{a})$ implies that $(\tilde{a},\tilde{a}) \in E_{i,j}^{\tilde{\mathcal{A}}}$ for every $i,j \in [D]^2$ and f is an isomorphism mapping a onto $\tilde{a}$. Assume that there is an element $b \in A$, $b \neq a$ and indices $i,j \in [D]^2$ such that either $(a,b) \in E_{i,j}^{\mathcal{A}}$ or $(b,a) \in E_{i,j}^{\mathcal{A}}$. Since $b \in N_2^{\mathcal{A}}(a)$ and f is an isomorphism we get that $f(b) \neq f(a) = \tilde{a}$ and either $(\tilde{a}, f(b)) \in E_{i,j}^{\tilde{\mathcal{A}}}$ or $(f(b), \tilde{a}) \in E_{i,j}^{\tilde{\mathcal{A}}}$. But this contradicts $\tilde{\mathcal{A}} \models \varphi(\tilde{a})$ and hence $\mathcal{A} \models \varphi(a)$.

We now prove $\mathcal{A} \models \psi(a)$. Let $k,k' \in ([D]^2)^2$ and $i,i' \in [D]^2$ such that $\mathrm{ROT}_{H^2}(k,i) = (k',i')$. Since $\tilde{\mathcal{A}} \models \psi(\tilde{a})$ there must be elements $\tilde{b}, \tilde{b}' \in \tilde{A}$ such that $(\tilde{a},\tilde{b}) \in F_k^{\tilde{\mathcal{A}}}$, $(\tilde{a},\tilde{b}') \in F_{k'}^{\tilde{\mathcal{A}}}$ and $(\tilde{b},\tilde{b}') \in E_{i,i'}^{\tilde{\mathcal{A}}}$. But since $\tilde{b},\tilde{b}' \in N_2^{\tilde{\mathcal{A}}}(\tilde{a})$ we get that $f^{-1}(\tilde{b})$ and $f^{-1}(\tilde{b}')$ are defined and since f is an isomorphism we get that $(a, f^{-1}(\tilde{b})) \in F_k^{\mathcal{A}}$, $(a, f^{-1}(\tilde{b}')) \in F_{k'}^{\mathcal{A}}$ and $(f^{-1}(\tilde{b}), f^{-1}(\tilde{b}')) \in E_{i,i'}^{\mathcal{A}}$. Hence $\mathcal{A} \models \exists y \exists y' \big(F_k(a,y) \wedge F_{k'}(a,y') \wedge E_{i,i'}(y,y')\big)$ for any $k,k' \in ([D]^2)^2$ and $i,i' \in [D]^2$ such that $\mathrm{ROT}_{H^2}(k,i) = (k',i')$ which implies that $\mathcal{A} \models \psi(a)$. ∎

Claim 4. Every structure $\mathcal{A} \in \bigcup_{1\leq k\leq m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$ satisfies $\varphi_{\text{recursion}}$.

Proof of Claim 4. Let $\mathcal{A} \in \bigcup_{1\leq k\leq m} P_{\rho_k} \setminus \{\mathcal{A}_\emptyset\}$. Then there is a $k \in \{1,\dots,m\}$ such that $\mathcal{A} \in P_{\rho_k}$.

By definition (Section 6.1.2 Equation 6.6), $\varphi_{\text{recursion}} := \forall x \forall z \big(\varphi(x,z) \vee \psi(x,z)\big)$, where

$$\varphi(x,z) := \neg\exists y F(x,y) \wedge \neg\exists y F(z,y) \text{ and}$$

$$\psi(x,z) := \bigwedge_{\substack{k_1',k_2'\in[D]^2\\ \ell_1',\ell_2'\in[D]^2}} \Bigg(\exists y\big[E_{k_1',\ell_1'}(x,y) \wedge E_{k_2',\ell_2'}(y,z)\big] \to \bigwedge_{\substack{i,j,i',j'\in[D],k,\ell\in([D]^2)^2\\ \mathrm{ROT}_H(k,i)=((k_1',k_2'),i')\\ \mathrm{ROT}_H((\ell_2',\ell_1'),j)=(\ell,j')}} \exists x'\exists z'\big[F_k(x,x') \wedge F_\ell(z,z') \wedge E_{(i,j),(j',i')}(x',z')\big]\Bigg).$$

Let $a,c \in A$. Assume first that there is $b \in A$ with $(a,b) \in F^{\mathcal{A}}$. Hence $\mathcal{A} \not\models \varphi(a,c)$. Since $\varphi_{\text{recursion}} := \forall x\forall z\big(\varphi(x,z) \vee \psi(x,z)\big)$ we aim to prove $\mathcal{A} \models \psi(a,c)$. By construction of ρ_k, there is an $i \in I_k$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \in \tau_i$. Therefore there is a structure $\tilde{\mathcal{A}} \models \varphi_{\text{Z}}$ and $\tilde{a} \in \tilde{A}$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), a) \cong (\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Let f be an isomorphism from $(\mathcal{N}_2^{\mathcal{A}}(a), a)$ to $(\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{a}), \tilde{a})$. Since $b \in N_2^{\mathcal{A}}(a)$, we get that $f(b)$ is defined. Since f is an isomorphism mapping a onto $\tilde{a}$, we have that $(a,b) \in F^{\mathcal{A}}$ implies that $(\tilde{a}, f(b)) \in F^{\tilde{\mathcal{A}}}$. Hence $\tilde{\mathcal{A}} \not\models \varphi(\tilde{a},\tilde{c})$, for every $\tilde{c} \in \tilde{A}$. But since $\tilde{\mathcal{A}} \models \varphi_{\text{recursion}}$, as $\tilde{\mathcal{A}} \models \varphi_{\text{Z}}$, this shows that $\tilde{\mathcal{A}} \models \psi(\tilde{a},\tilde{c})$ for every $\tilde{c} \in \tilde{A}$.

Let $k_1', k_2' \in [D]^2$ and $\ell_1', \ell_2' \in [D]^2$ be indices such that there is $b' \in A$ with $(a,b') \in E^{\mathcal{A}}_{k_1',\ell_1'}$ and $(b',c) \in E^{\mathcal{A}}_{k_2',\ell_2'}$. Since $b', c \in N_2^{\mathcal{A}}(a)$, by assumption we get that $f(b')$ and $f(c)$ are defined. Furthermore, $(a,b') \in E^{\mathcal{A}}_{k_1',\ell_1'}$ and $(b',c) \in E^{\mathcal{A}}_{k_2',\ell_2'}$ imply that $(\tilde{a}, f(b')) \in E^{\tilde{\mathcal{A}}}_{k_1',\ell_1'}$ and $(f(b'), f(c)) \in E^{\tilde{\mathcal{A}}}_{k_2',\ell_2'}$, since f is an isomorphism mapping a onto $\tilde{a}$. We proved in the previous paragraph that $\tilde{\mathcal{A}} \models \psi(\tilde{a}, f(c))$. Hence we can conclude that for all indices $i,j,i',j' \in [D]$, $k,\ell \in ([D]^2)^2$ for which $\mathrm{ROT}_H(k,i) = ((k_1',k_2'),i')$ and $\mathrm{ROT}_H((\ell_2',\ell_1'),j) = (\ell,j')$, there are elements $\tilde{a}', \tilde{c}' \in \tilde{A}$ such that $(\tilde{a},\tilde{a}') \in F_k^{\tilde{\mathcal{A}}}$, $(f(c),\tilde{c}') \in F_\ell^{\tilde{\mathcal{A}}}$, and $(\tilde{a}',\tilde{c}') \in E^{\tilde{\mathcal{A}}}_{(i,j),(j',i')}$. Since $\tilde{a}', \tilde{c}' \in N_2^{\tilde{\mathcal{A}}}(\tilde{a})$, we get that $a' := f^{-1}(\tilde{a}')$ and $c' := f^{-1}(\tilde{c}')$ are defined. Furthermore, we get that $(a,a') \in F_k^{\mathcal{A}}$, $(c,c') \in F_\ell^{\mathcal{A}}$ and $(a',c') \in E^{\mathcal{A}}_{(i,j),(j',i')}$. This proves that $\mathcal{A} \models \psi(a,c)$.

In the case that there is $b \in A$ with $(c,b) \in F^{\mathcal{A}}$, we can prove similarly that $\mathcal{A} \models \psi(a,c)$, by considering that there exist $\tilde{\mathcal{A}} \models \varphi_{\text{Z}}$ and $\tilde{c} \in \tilde{A}$ such that $(\mathcal{N}_2^{\mathcal{A}}(a), c) \cong (\mathcal{N}_2^{\tilde{\mathcal{A}}}(\tilde{c}), \tilde{c})$ by construction of ρ_k. Finally if there is no $b \in A$ such that $(a,b) \in F^{\mathcal{A}}$ or $(c,b) \in F^{\mathcal{A}}$ then $\mathcal{A} \models \varphi(a,c)$. Since this covers every case we get that $\mathcal{A} \models \varphi_{\text{recursion}}$. ■

Assume $\mathcal{A} \in \bigcup_{1\leq k\leq m} P_{\rho_k}$. As proved in Claims 1, 2, 3 and 4 this implies that $\mathcal{A} \models \varphi'_{\text{tree}}$, $\mathcal{A} \models \varphi_{\text{rotationMap}}$, $\mathcal{A} \models \varphi_{\text{base}}$ and $\mathcal{A} \models \varphi_{\text{recursion}}$. Since φ'_{Z} is a conjunction of these formulas, we get $\mathcal{A} \models \varphi'_{\text{Z}}$ and hence $\mathcal{A} \in P'_{\text{Z}}$. □

8.3.2 A local reduction from relational structures to graphs

In this section we construct a property $\mathcal{P}_{\text{graph}}$ from the property $P'_{\text{Ⓩ}}$. We obtain this graph property as $f(P'_{\text{Ⓩ}})$ by defining a map $f : C_d \to \mathcal{C}_d$. To define f we introduce a distinct arrow-graph gadget for every relation in σ (i. e. for every edge colour). The map f then replaces every tuple in a certain relation (every coloured edge) by the respective arrow-graph gadget. We further prove that this replacement operation defines a local reduction f from $P'_{\text{Ⓩ}}$ to $\mathcal{P}_{\text{graph}}$. Recall that a local reduction is a function maintaining distance that can be simulated locally by queries. Since by Lemma 3.6.3 local reductions preserve testability, we use the local reduction from $P'_{\text{Ⓩ}}$ to $\mathcal{P}_{\text{graph}}$ to obtain non-testability of the property $\mathcal{P}_{\text{graph}}$ from the non-testability of $P'_{\text{Ⓩ}}$ which easily follows from the non-testability of $P_{\text{Ⓩ}}$. We will now define f formally.

Let ℓ be the number of relations (the number of edge colours) in σ. We first introduce the different types of arrow-graph gadgets we need to define the local reduction. For $1 \leq k \leq \ell$, we let H_k be the graph with vertex set $V(H_k) := \{a_1, \dots, a_{2\ell+2}, b_1, b_2\}$ and edge set

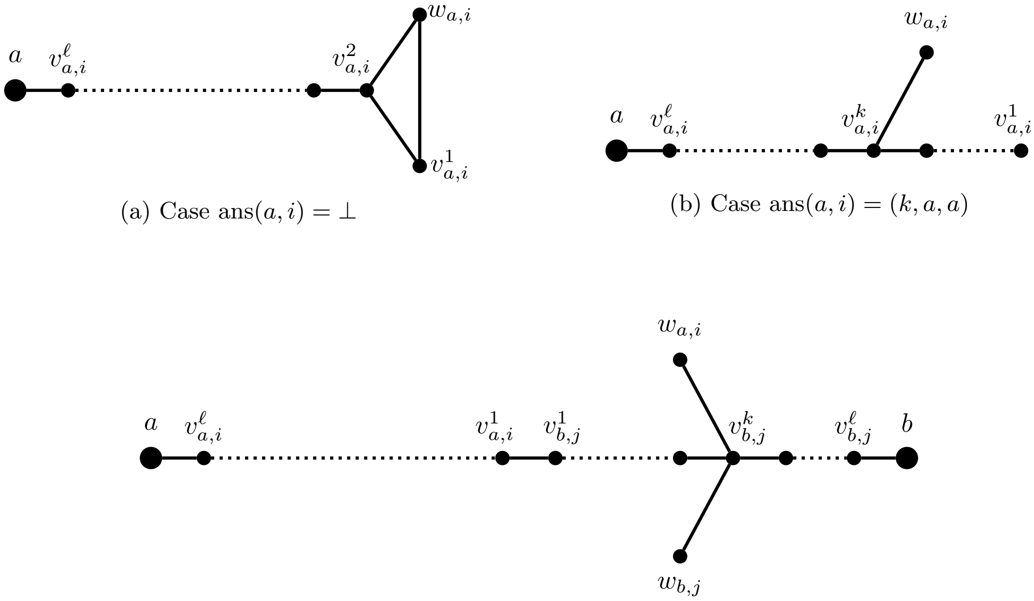
$$E(H_k) := \{\{a_i, a_{i+1}\} \mid 1 \leq i \leq 2\ell + 1\} \cup \{\{a_{\ell+1+k}, b_j\} \mid j \in \{1, 2\}\}.$$

We call H_k a *k-arrow.* For any graph G and vertices $v, w \in V(G)$, we say that there is a k-arrow from v to w, denoted $v \xrightarrow{k} w$, if there are $2\ell + 2$ vertices $v_2, \dots, v_{2\ell+1}, w_1, w_2 \in V(G)$ and an isomorphism $g : H_k \to \mathcal{N}_1^G(v_2, \dots, v_{2\ell+1}, w_1, w_2)$ such that $g(a_1) = v$ and $g(a_{2\ell+2}) = w$. We now define a second arrow gadget. For $1 \leq k \leq \ell$, we let L_k be the graph with vertex set $V(L_k) := \{a_1, \dots, a_{\ell+1}, b\}$ and edge set $E(L_k) := \{\{a_i, a_{i+1}\} \mid 1 \leq i \leq \ell\} \cup \{\{a_k, b\}\}$. We call L_k a *k-loop.* For any graph G and vertex $v \in V(G)$, we say that there is a k-loop at v, denoted $v \xrightarrow{k} v$, if there are $\ell + 1$ vertices $v_1, \dots, v_\ell, w \in V(G)$ and an isomorphism $g : L_k \to \mathcal{N}_1^G(v_1, \dots, v_\ell, w)$ such that $g(a_{\ell+1}) = v$. Finally we let $H_\perp$ be the graph with vertex set $V(H_\perp) := \{a_1, \dots, a_{\ell+1}, b\}$ and edge set $E(H_\perp) := \{\{a_i, a_{i+1}\} \mid 1 \leq i \leq \ell\} \cup \{\{a_i, b\} \mid i \in \{1, 2\}\}$. We call $H_\perp$ a *non-arrow.* For any graph G and vertex $v \in V(G)$, we say that there is a non-arrow at v, denoted $v \not\to$, if there are $\ell + 1$ vertices $v_1, \dots, v_\ell, w \in V(G)$ and an isomorphism $g : H_\perp \to N_1^G(v_1, \dots, v_\ell, w)$ such that $g(a_{\ell+1}) = v$.

We now define a function $f : C_d \to \mathcal{C}_d$ by $f(\mathcal{A}) := G_\mathcal{A}$, where $G_\mathcal{A}$ is the graph on vertex set $V(G_\mathcal{A}) := A \cup \{v^k_{a,i}, w_{a,i} \mid 1 \leq i \leq d, a \in A, 1 \leq k \leq \ell\}$ and edge set

$$\begin{aligned} E(G_\mathcal{A}) := & \Big\{\{a, v^\ell_{a,i}\} \mid a \in A, 1 \leq i \leq d\Big\} \cup \Big\{\{v^k_{a,i}, v^{k+1}_{a,i}\} \mid 1 \leq k \leq \ell - 1, a \in A, 1 \leq i \leq d\Big\} \\ & \cup \Big\{\{v^k_{b,j}, w_{b,j}\}, \{v^k_{b,j}, w_{a,i}\}, \{v^\ell_{a,i}, v^\ell_{b,j}\} \mid a \neq b, \text{ans}(a, i) = \text{ans}(b, j) = (k, a, b)\Big\} \\ & \cup \Big\{\{v^k_{a,i}, w_{a,i}\} \mid \text{ans}(a, i) = (k, a, a)\Big\} \cup \Big\{\{v^1_{a,i}, w_{a,i}\}, \{v^2_{a,i}, w_{a,i}\} \mid \text{ans}(a, i) = \perp\Big\}, \end{aligned}$$

where $\text{ans}(a, i) = (k, a, b)$ denotes that the i-th tuple of a is (a, b) and is in the k-th relation. Hence $G_\mathcal{A}$ is defined in such a way that if (a, b) is a tuple in the k-th relation of σ in $\mathcal{A}$, then $a \xrightarrow{k} b$ in $G_\mathcal{A}$, and a has a non-arrow for every i satisfying that $\text{ans}(a, i) = \perp$ for every k. For



(a) Case $\mathrm{ans}(a,i) = \perp$

(b) Case $\mathrm{ans}(a,i) = (k,a,a)$

(c) Case $\mathrm{ans}(a,i) = \mathrm{ans}(b,j) = (k,a,b)$

Figure 8.4: Different types of arrows in $G_\mathcal{A}$.

illustration see Figure 8.4.

Now we define property $\mathcal{P}_{\text{graph}} := \{f(\mathcal{A}) \mid \mathcal{A} \in P'_{Z}\} \subseteq \mathcal{C}_d$.

Lemma 8.3.3. *The map f is a local reduction from P'_{Z} to $\mathcal{P}_{\text{graph}}$.*

Proof. First note that for any $\mathcal{A} \in P'_{Z}$, we have that $f(\mathcal{A}) \in \mathcal{P}_{\text{graph}}$ by definition and hence property (LR4) of local reductions follows. Furthermore, properties (LR1) and (LR2) are trivially true.

Now let $c := 2d + 2d^2\ell$. We prove that if $\mathcal{A} \in C_d$ is ϵ-far from P'_{Z} then $f(\mathcal{A})$ is $\frac{\epsilon}{c}$-far from $\mathcal{P}_{\text{graph}}$ by contraposition. Therefore assume that $f(\mathcal{A}) =: G_\mathcal{A}$ is not $\frac{\epsilon}{c}$-far from $\mathcal{P}_{\text{graph}}$ for some $\mathcal{A} \in C_d$. Then there is a set $E \subseteq \{e \subseteq V(G_\mathcal{A}) \mid |e| = 2\}$ of size at most $\frac{\epsilon d |V(G_\mathcal{A})|}{c}$, and a graph $G \in \mathcal{P}_{\text{graph}}$ such that G is obtained from $G_\mathcal{A}$ by modifying the tuples in E. By definition of $\mathcal{P}_{\text{graph}}$, there is a structure $\mathcal{A}_G \in P'_{Z}$ such that $f(\mathcal{A}_G) = G$. First note that $|A_G| = |A|$, as $(1 + d\ell)|A| = |V(G_\mathcal{A})| = |V(G)| = (1 + d\ell)|A_G|$. Hence there must be a set R of tuples that need to be modified to make $\mathcal{A}$ isomorphic to $\mathcal{A}_G$. First note that R cannot contain a tuple (a,b) where $\{a, v^k_{a,i}, w_{a,i}, b, v^k_{b,i}, w_{b,i} \mid 1 \le i \le d, 1 \le k \le \ell\} \cap e = \emptyset$ for every $e \in E$. This is because if (a,b) is a tuple in $\mathcal{A}$, then $a \xrightarrow{k} b$ for some k in $G_\mathcal{A}$. But since $\{a, v^k_{a,i}, w_{a,i}, b, v^k_{b,i}, w_{b,i} \mid 1 \le i \le d, 1 \le k \le \ell\} \cap e = \emptyset$ for every $e \in E$, we have that $a \xrightarrow{k} b$ in G. But then (a,b) must be a tuple in $\mathcal{A}_G$, and hence (a,b) cannot be in R. The same argument works when assuming that (a,b) is a tuple in $\mathcal{A}_G$. Since for every $e \in E$, there are at most $2d$

tuples (a, b) such that $\{a, v^k_{a,i}, w_{a,i}, b, v^k_{b,i}, w_{b,i} \mid 1 \leq i \leq d, 1 \leq k \leq \ell\} \cap e \neq \emptyset$, we get that

$$|R| \leq \frac{2d^2\epsilon \cdot |V(G_{\mathcal{A}})|}{c} = \frac{2(1+d\ell)\epsilon d^2 \cdot |A|}{c} = \epsilon d \cdot |A|.$$

Hence $\mathcal{A}$ is not ϵ-far to being in $P'_{Ⓩ}$. This implies property (LR5) of local reductions.

Let $t := d + 1$. Let $\mathcal{A} \in C_d$ and $G_{\mathcal{A}} := f(\mathcal{A})$. Note that any $a \in A$ is adjacent in $G_{\mathcal{A}}$ to $v^\ell_{a,i}$, for every $1 \leq i \leq d$. Hence any neighbour query in $G_{\mathcal{A}}$ to a can be answered without querying $\mathcal{A}$. Assume $v \in \{v^k_{a,i}, w_{a,i} \mid 1 \leq k \leq \ell\}$ for some $a \in A$ and some $1 \leq i \leq d$. Then we can determine all neighbours of v by querying (a, i) and further if $\text{ans}(a, i) \neq \perp$ and $\text{ans}(a, i) = (k, a, b)$, then we need to query (b, j) for every $1 \leq j \leq d$. Hence we can determine the answer to any query to $G_{\mathcal{A}}$ by making t queries to $\mathcal{A}$ which implies property (LR3) of local reductions. This proves that f is a local reduction from $P'_{Ⓩ}$ to $\mathcal{P}_{\text{graph}}$. □

We remark that $\mathcal{P}_{\text{graph}}$ is a simpler version of the simple graph property defined in Section 6.3 where extra care had to be taken to define degree-regular graphs.

8.3.3 The graph property is GSF-local

Let $\mathcal{P}_{\text{graph}}$ be the graph property as defined in Section 8.3.2. We now show that $\mathcal{P}_{\text{graph}}$ is GSF-local.

Lemma 8.3.4. *The graph property $\mathcal{P}_{\text{graph}}$ is GSF-local.*

Proof. For this we will prove that $\mathcal{P}_{\text{graph}}$ is equal to a finite union of properties defined by 0-profiles, and then use Theorem 8.2.5 to prove that $\mathcal{P}_{\text{graph}}$ is GSF-local. We define the 0-profiles for $\mathcal{P}_{\text{graph}}$ in a very similar way to the relational structure case, and then use the description of $P'_{Ⓩ}$ by 0-profiles shown in Lemma 8.3.2. To this end let $\tau_1, \dots, \tau_t$ be a list of all 2-types of σ-structures of bounded degree d and $\hat{\tau}_1, \dots, \hat{\tau}_s$ be a list of all $(4\ell + 2)$-types of graphs of bounded degree d. Assume that the $(4\ell + 2)$-types $\hat{\tau}_1, \dots, \hat{\tau}_s$ are ordered in such a way that $(\mathcal{N}^{f(\mathcal{B})}_{4\ell+2}(b), b) \in \hat{\tau}_k$, for every $k \in \{1, \dots, m\}$ and $(\mathcal{B}, b) \in \tau_k$, where m is the number of parts of the partition of $P_{Ⓩ}$ defined in Subsection 8.3.1. Recall that P_k is a part in the partition of $P_{Ⓩ}$ defined in Section 8.3.1 for every $k \in \{1, \dots, m\}$. For $k \in \{1, \dots, m\}$, let $\hat{I}_k$ be the set of indices i such that there is $\mathcal{A} \in P_k$, and $v \in V(f(\mathcal{A}))$ for which $(\mathcal{N}^{f(\mathcal{A})}_{4\ell+2}(v), v) \in \hat{\tau}_i$. Let $\hat{\rho_k} : \{1, \dots, s\} \to \mathfrak{I}_0$ be defined by

$$\hat{\rho_k}(i) := \begin{cases} [0,1] & \text{if } i = k, \\ [0,\infty) & \text{if } i \in \hat{I}_k \setminus \{k\}, \\ [0,0] & \text{otherwise.} \end{cases}$$

Claim 1. It holds that $\mathcal{P}_{\text{graph}} = \bigcup_{1 \leq k \leq m} P_{\hat{\rho}_k}$.

Proof of Claim 1. First we prove $\mathcal{P}_{\text{graph}} \subseteq \bigcup_{1 \le k \le m} P_{\hat{\rho}_k}$. Assume $G \in \mathcal{P}_{\text{graph}}$ and let $\mathcal{A} \in P'_{Ⓩ}$ be a structure such that $G = f(\mathcal{A})$. If $\mathcal{A} = \mathcal{A}_\emptyset$ then clearly $G \in \bigcup_{1 \le k \le m} P_{\hat{\rho}_k}$. Hence assume $\mathcal{A} \neq \mathcal{A}_\emptyset$. Then $\mathcal{A} \in P_k$ for some $k \in \{1, \ldots, m\}$. By the construction of $\hat{I}_k$ we know that for every $v \in V(G)$ we have $(\mathcal{N}^G_{4\ell+2}(v), v) \in \hat{\tau}_i$ for some $i \in \hat{I}_k$. Furthermore, since $\mathcal{A} \in P_k$ there is at most one $a \in A$ with $(\mathcal{N}^{\mathcal{A}}_2(a), a) \in \tau_k$. This implies directly that there can be at most one vertex $v \in V(G)$ with $(\mathcal{N}^G_{4\ell+2}(v), v) \in \hat{\tau}_k$ and hence $G \in P_{\hat{\rho}}$.

Now we prove that $\bigcup_{1 \le k \le m} P_{\hat{\rho}_k} \subseteq \mathcal{P}_{\text{graph}}$. Let $G \in \bigcup_{1 \le k \le m} P_{\hat{\rho}_k}$ and let $k \in \{1, \ldots, m\}$ be an index such that $G \in P_{\hat{\rho}_k}$.

First note that every model of $\varphi_{Ⓩ}$ is d regular for some large d. Then for any $\mathcal{A} \models \varphi_{Ⓩ}$, every vertex in $f(\mathcal{A})$ has either degree less or equal to 4 or degree d . Since every structure in $P'_{Ⓩ}$ apart from the empty structure $\mathcal{A}_\emptyset$ is a model of $\varphi_{Ⓩ}$, this implies that every vertex in any graph $G' \in \mathcal{P}_{\text{graph}}$ has degree ≤ 4 or degree d. Since for every i for which $\hat{\rho}(i) \neq [0, 0]$, there is a graph $G' \in \mathcal{P}_{\text{graph}}$ and $v \in V(G')$ such that $(\mathcal{N}^{G'}_{4\ell+2}(v), v) \in \hat{\tau}_i$, we get that every vertex in G has to have degree less or equal to 4 or degree d. Using this argument further, we get that every vertex $v \in V(G)$ of degree less or equal 4 has to be contained in the $(\ell+1)$-neighbourhood of a vertex of degree d, and that the $(2\ell+1)$-neighbourhood of every vertex $v \in V(G)$ of degree d is the union of k-arrows, k-loops and non-arrows which are disjoint apart from their endpoints. Hence there is a σ-structure $\mathcal{A}$ such that $f(\mathcal{A}) \cong G$. Let g be an isomorphism from $f(\mathcal{A})$ to G.

Now we argue that $\mathcal{A} \in P_{\rho_k}$. First assume that there are two elements a, b in $\mathcal{A}$ with $(\mathcal{N}^{\mathcal{A}}_2(a), a) \in \tau_k$ and $(\mathcal{N}^{\mathcal{A}}_2(b), b) \in \tau_k$. By definition, we get that $(\mathcal{N}^{f(\mathcal{A})}_{4\ell+2}(a), a) \in \hat{\tau}_k$ and $(\mathcal{N}^{f(\mathcal{A})}_{4\ell+2}(b), b) \in \hat{\tau}_k$. Since g is an isomorphism, the restriction of g to $N^{f(\mathcal{A})}_{4\ell+2}(a)$ must be an isomorphism from $\mathcal{N}^{f(\mathcal{A})}_{4\ell+2}(a)$ to $\mathcal{N}^G_{4\ell+2}(g(a))$, and hence $(\mathcal{N}^G_{4\ell+2}(g(a)), g(a)) \cong (\mathcal{N}^{f(\mathcal{A})}_{4\ell+2}(a), a) \in \hat{\tau}_k$. But the same holds for the $(4\ell+2)$-ball of $g(b)$, and hence we contradict the assumption that $G \in P_{\hat{\rho}_k}$ since $\hat{\rho}_k(k) = [0, 1]$. Let us further assume that there is an $a \in A$ such that $(\mathcal{N}^{\mathcal{A}}_2(a), a) \in \tau_i$ for some $i \notin I_k$. Let j be the index such that $(\mathcal{N}^{f(\mathcal{A})}_{4\ell+2}(a), a) \in \hat{\tau}_j$. Additionally note that a must have degree d in $f(\mathcal{A})$ by construction of f. As g is an isomorphism, we get that $(\mathcal{N}^G_{4\ell+2}(g(a)), g(a)) \in \hat{\tau}_j$, and $g(a)$ has degree d. But then by construction of $\hat{\rho}_k$, there must be $G' \in \mathcal{P}_{\text{graph}}$ and a vertex $v \in V(G')$ of degree d such that $(\mathcal{N}^{G'}_{4\ell+2}(v), v) \in \hat{\tau}_j$. By construction of $\mathcal{P}_{\text{graph}}$, there is a structure $\mathcal{A} \in P'_{Ⓩ}$ such that $f(\mathcal{A}') = G'$. Since v has degree d, it must be an element in $\mathcal{A}'$. Furthermore $(\mathcal{N}^{\mathcal{A}'}_2(v), v) \in \tau_i$ by choice of i and j. Hence $\mathcal{A}' \notin P_{\rho_k}$. But this contradicts Lemma 8.3.2.

Hence we have shown that $\mathcal{A} \in P_{\rho_k}$. Then by Lemma 8.3.2 $\mathcal{A} \in P'_{Ⓩ}$, and by construction $G \in \mathcal{P}_{\text{graph}}$. ∎

Since by Claim 1 we can express $\mathcal{P}_{\text{graph}}$ as a finite union of properties each defined by a 0-profile, Theorem 8.2.5 implies that $\mathcal{P}_{\text{graph}}$ is GSF-local. □

8.3.4 Putting everything together

Now we prove the main theorem of this section.

Proof of Theorem 8.0.1. Let the property $P'_{\textcircled{z}}$ of relational structures be as defined above. Note that $P'_{\textcircled{z}}$ is not testable, as $P_{\textcircled{z}}$ is not testable by Theorem 6.2.1 and $P'_{\textcircled{z}}$ only differs from $P_{\textcircled{z}}$ by the empty structure. By Lemma 8.3.3 and Lemma 3.6.3, the graph property $\mathcal{P}_{\text{graph}}$ that is locally reduced from $P'_{\textcircled{z}}$ is not testable. Lemma 8.3.4 shows that $\mathcal{P}_{\text{graph}}$ is also a GSF-local property. Hence there exists a GSF-local property of bounded degree graphs which is not testable. Furthermore, since having a POT implies being testable (Theorem 3.4.3), this proves that there is a GSF-local property which has no POT. By Theorem 8.1.4 this implies that not all GSF-local properties are non-propagating. □

8.4 Summary

In this chapter we have utilised the construction of an FO definable property, which is not testable, to prove that generalised subgraph freeness properties are not in general testable. This entails that it is not in general possible to modify a graph containing a few copies of forbidden marked graphs without causing a chain reaction of necessary edge modifications. This answers an open question from [76] and provides a missing piece of understanding in the characterisation of properties which allow constant query one-sided error POT's given in [76]. Our result also proves that the characterisation of which monotone properties and which hereditary properties have one-sided error constant query testers can not be extended to non-monotone, non-hereditary properties, which was asked as an open question in [89]. We believe that our result will aid our understanding of which properties are testable and may be useful for the goal of characterising which properties are testable in the bounded degree model.

Chapter 9

On testability of NP-hard problems and construction of hard instances for property testing

Since the seminal work of Cook [27] and Karp [92] the study of NP-hard problems plays a central role in Computer Science. Besides identifying such problems, a great deal of research has been conducted into developing exact efficient algorithms for NP-hard problems by restricting the set of inputs in some way and into finding approximation algorithms for NP-hard problems. In most known cases NP-hard problems appear to be hard in the setting of property testing as well. Several problems which remain NP-hard on graphs of bounded degree have query complexity $\Theta(n)$, such as 3-SAT, 3-colourability [22], Hamiltonicity [70, 129], Independent Set [70], 3-edge-colourability, 3-dimensional matching [129]. On the other hand, there are some NP-hard problems which are constant query testable, e. g. Hamiltonicity on bounded degree planar graphs for which NP-hardness was shown in [67] and, as planar graphs are hyperfinite, testability follows from the result of Newman and Sohler [112] which states that every property can be tested for the class of hyperfinite graph of bounded degree. There is a variety of problems which remain NP-hard on bounded degree planar graphs (see e. g. [65]) which therefore yield examples of testable NP-hard problems.

In this chapter we study the complexity of property testing of NP-hard problems with the aim of investigating what constitutes hardness in the context of property testing. Besides showing lower-bounds for two NP-hard problems (i. e. dominating set and treewidth) we also investigate hardness for property testing by providing a deterministic construction of hard-instances for a known hard problem for property testing, i. e. Hamiltonicity. We believe that this construction advances our understanding of property testing complexity.

Overview of results and techniques In this Section we show three property testing lower bounds for three different NP-hard problems using three different techniques. We want to remark here that graph property testing does not allow an input parameter. We solve this here by considering properties were the parameter is determined by some function of n – the number of vertices. To extend property testing in such a way as to allow an input parameter there should be some consideration of an appropriate extension of the notion of ϵ-farness.

In Section 9.1 we show that testing whether a graph has a dominating set containing at most a quarter of all vertices takes at least a linear amount of queries. We show this via a local reduction from 3-SAT, for which a linear lower-bound on the query complexity has been shown in [22]. Showing hardness via a local reduction has been used in this context in [22,70,129]. Furthermore, the local reduction presented in this section is a slight variation of a straightforward polynomial time reduction from 3-SAT. We adjust this reduction in such a way as to benefit simplicity of proving that the reduction is local. To our knowledge the result presented in this Section was not previously known. Note that for fixed parameter k, considering k-dominating set on bounded degree graphs is of no interest as any large enough graph can not contain such a dominating set.

In Section 9.2 we give an explicit deterministic construction of a class of graphs of bounded degree that are locally Hamiltonian, but (globally) far from being Hamiltonian. By locally Hamiltonian we mean that every subgraph induced by the neighbourhood of a small vertex set appears in some Hamiltonian graph. More precisely, we obtain graphs which differ in $\Theta(n)$ edges from any Hamiltonian graph, but non-Hamiltonicity cannot be detected in the neighbourhood of $o(n)$ vertices. Our class of graphs yields a class of hard instances for one-sided error property testers with linear query complexity. It is known that any property tester (even with two-sided error) requires a linear number of queries to test Hamiltonicity [70,129]. This is proved via a randomised construction of hard instances. We hope that studying hard instances will further our understanding of the complexity of property testing. A similar approach was taken in [22] for 3-colourability, where graphs, which are far from being 3-colourable but locally look 3-colourable, are implicitly obtained using a reduction from the constraint satisfaction problem (CSP). An explicit construction of a CSP, which is far from being satisfiable but every sublinear subset of constraints is satisfiable, is given. To our knowledge this is the only other known deterministic construction of a similar kind.

In Section 9.3 we discuss the testability of treewidth. First observe that for fixed parameter k, testing the property $\mathcal{P}^{\mathrm{tw}}_{\leq k}$ of bounded degree graphs with treewidth at most k can be done with a constant number of queries. This is due to $\mathcal{P}^{\mathrm{tw}}_{\leq k}$ being minor-closed which implies testability [18]. We show that for every sub-linear, super-constant function $f \in o(n) \cap \omega(n)$ we can not test whether a given bounded degree graph G on n vertices has treewidth at most $f(n)$. Hence we can for example not test the property of all graphs whose treewidth is logarithmic in the number of vertices of the graph. We use a theorem of Grohe and Marx [80] that shows that expanders have linear treewidth and conclude non-testability by using a similar argument as in the proof of Theorem 6.2.1. We argue that any graph with linear treewidth is far from having treewidth $f(n)$ and combine the theorem by Alon introduced in Section 5.2 [102, Proposition 19.10] and the theorem by Adler and Harwath introduced in Section 5.1 [2, Theorem 19] to prove non-testability.

9.1 Lower bound for testing dominating set with two-sided error

In this section we prove that testing dominating set size takes at least a linear amount of queries. We first briefly introduce the dominating set problem. Let G be a graph. For any subset $D \subseteq V(G)$ and any vertex $v \in V(G)$ we say that D *dominates* v if $v \in D$ or there is $w \in D$ such that $\{v, w\} \in E(G)$. A subset $D \subseteq V(G)$ is called a *dominating set* of G if D dominates every vertex $v \in V(G)$. The minimum size of a dominating set of G is called the domination number of G. Let $d \in \mathbb{N}$ and $\mathcal{P}_d^{\mathrm{DS}} \subseteq \mathcal{C}_d$ be the property of all graphs G of bounded degree d with domination number no more than $\frac{|V(G)|}{4}$.

We show that $\mathcal{P}_d^{\mathrm{DS}}$ is hard to test for degree $d = 5$ using a local reduction from 3-SAT. The reduction we use is similar to a straightforward approach of reducing 3-SAT to dominating set. We will first introduce the model for testing 3-SAT and then we will give the reduction.

9.1.1 Property testing model for bounded degree 3-SAT

We use the model for testing whether a 3-CNF formula is satisfiable or not as in [22].

Here we consider propositional formulas i. e. quantifier-free formulas over the language σ consisting of one unary relation T. We denote the set of variables of a propositional formula φ by $\textsc{Var}(\varphi)$. We say that a propositional formula φ is satisfiable if there is an assignment $\alpha : \textsc{Var}(\varphi) \to \{0, 1\}$ of the variables of φ such that the σ-structure $\mathcal{A} = (\{0, 1\}, T^{\mathcal{A}})$, where $T^{\mathcal{A}} = \{(1)\}$, satisfies φ under the assignment α. We remind the reader that a propositional formula φ over the set of variables $\textsc{Var}(\varphi)$ is in 3-CNF if it has the form

$$\varphi = \bigwedge_{i=1}^{m} \left(\ell_i^1 \vee \ell_i^2 \vee \ell_i^3 \right),$$

where $\ell_i^j \in \{x, \neg x \mid x \in \textsc{Var}(\varphi)\}$ for $i \in \{1, \ldots, m\}$, $j \in \{1, 2, 3\}$. We call ℓ_i^j a literal and $\ell_i^1 \vee \ell_i^2 \vee \ell_i^3$ a clause of the CNF. Note that we can assume that every clause has precisely three literals, as $\ell \equiv \ell \vee \ell$ for any literal ℓ. Additionally we assume that all clauses are pairwise distinct.

Let $d \in \mathbb{N}$. We say that a 3-CNF formula has bounded degree d if every variable is contained in at most d clauses, where variable x is contained in a clause if either x or $\neg x$ is a literal of the clause. Note that deciding satisfiability of a bounded degree 3-CNF remains NP-hard for degree bound 3 as we can rename each appearance of a variable x by a different auxiliary variable and then force all auxiliary variables to have to be assigned the same value. Let CNF_d be the set of all 3-CNFs of bounded degree d and let $\mathcal{P}^d_{\mathrm{SAT}} \subseteq \mathrm{CNF}_d$ be the set of all satisfiable 3-CNFs of bounded degree d.

We encode a 3-CNF φ as a membership list M_φ which contains for each variable $x \in \textsc{Var}(\varphi)$ and each $1 \leq j \leq d$ the j-th clause x is contained in or $\perp$ if x is contained in fewer than j clauses. Furthermore a property tester is given $n = |\textsc{Var}(\varphi)|$ as auxiliary information and we allow queries from $Q_\varphi := [n] \times [d]$ and the property tester obtains the following answer

$$\mathrm{ans}_\varphi(i,j) := \begin{cases} \ell^1_k \vee \ell^2_k \vee \ell^3_k & \text{if } \ell^1_k \vee \ell^2_k \vee \ell^3_k \text{ is the } j\text{-th clause of } x_i, \\ \perp & \text{if } x_i \text{ is contained in less than } j \text{ clauses.} \end{cases}$$

Distance in this model is defined in terms of clause addition or removal. This leads to the following definition.

Definition 9.1.1 (Distance to satisfiability)**.** We say that a 3-CNF φ of bounded degree d with $|\textsc{Var}(\varphi)| = n$ is ϵ-far from being satisfiable if any formula obtained from φ by removing any subset of at most $\frac{\epsilon d n}{3}$ clauses is not satisfiable.

We use the following hardness result for property testing $\mathcal{P}^d_{\mathrm{SAT}}$.

Lemma 9.1.2 (Lemma 20 in [22])**.** *For every $\epsilon \in (0, \frac{1}{8})$ there are constants $d \in \mathbb{N}$, $c > 0$ such that every algorithm that distinguishes satisfiable instances of CNF_d with n variables from instances that are ϵ-far from satisfiable must have query complexity at least cn.*

9.1.2 Local reduction from 3-SAT to dominating set

In this section we show the following theorem using a local reduction from 3-SAT.

Theorem 9.1.3. *The property $\mathcal{P}^{\mathrm{DS}}_d$ is not testable with $o(n)$ queries on the class $\mathcal{C}_d$ for $d \geq 5$.*

Proof. We prove this theorem by defining a local reduction from $\mathcal{P}^{d'}_{\mathrm{SAT}}$ to $\mathcal{P}^{\mathrm{DS}}_d$, where d' is the degree bound from Lemma 9.1.2. The theorem then follows from Lemma 9.1.2 and Lemma 3.6.3.

We define $f : \mathrm{CNF}_{d'} \to \mathcal{C}_d$ as follows. For a CNF $\varphi = \bigwedge_{i=1}^m \left(\ell^1_i \vee \ell^2_i \vee \ell^3_i\right) \in \mathrm{CNF}_{d'}$ with

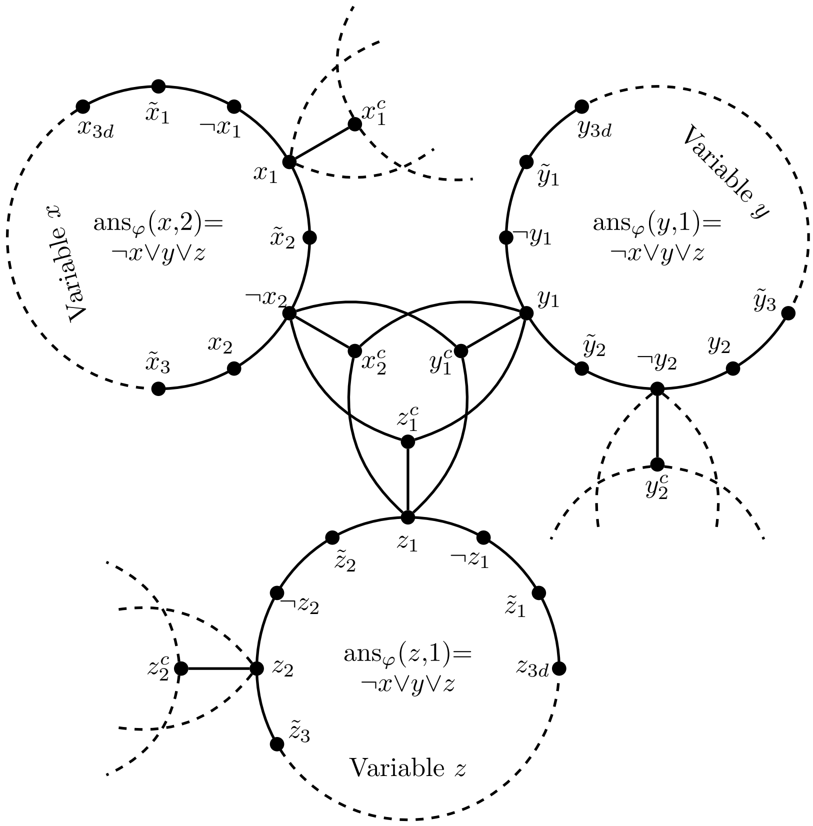


Figure 9.1: Illustration of the dominating set reduction.

$|\text{VAR}(\varphi)| = n$ we let $f(\varphi) = G_\varphi$, where

$$
\begin{aligned}
V(G_\varphi) :=& \Big\{ x_i, \neg x_i, \tilde{x}_i, x_i^c \;\Big|\; x \in \text{VAR}(\varphi), i \in \{1, \dots, d'\} \Big\} \text{ and} \\
E(G_\varphi) :=& \Big\{ \{\tilde{x}_i, \neg x_i\}, \{\neg x_i, x_i\}, \{x_i, \tilde{x}_j\} \;\Big|\; i \in \{1, \dots, d'\}, j = i+1 \text{ if } i < 3d', j = 1 \text{ otherwise} \Big\} \\
& \cup \Big\{ \{\ell_i, y_j^c\} \;\Big|\; \ell \in \{x, \neg x\}, i, j \in \{1, \dots, d'\}, \text{ans}_\varphi(x,i) = \text{ans}_\varphi(y,j) \Big\} \\
& \cup \Big\{ \{x_i, y_i^c\}, \{\neg x_i, y_i^c\} \;\Big|\; i \in \{1, \dots, d'\}, \text{ans}_\varphi(x,i) = \bot \Big\}.
\end{aligned}
$$

Hence G_φ consists of a cycle of length $3d'$ and d' extra vertices, which we refer to as clause vertices, for every variable. Clauses get encoded into edges between cycle and clause vertices. For illustration see Figure 9.1. Note that a 3-CNF on n variables gets mapped to a graph on $4d'n$ vertices which implies property (LR1). This also implies we can compute the auxiliary information for graph G_φ from the auxiliary information for φ showing property (LR2).

In order to prove that f defines a local reduction we need to prove that f can be computed locally (LR3), that if φ is satisfiable then G_φ has a dominating set of size at most $\frac{|V(G_\varphi)|}{4}$ (LR4) and that there is a function $h : (0,1) \to (0,1)$ such that if φ is ϵ-far from satisfiable, then G_φ is $h(\epsilon)$-far from $\mathcal{P}_d^{\mathrm{DS}}$ (LR5).

In order to prove (LR3), we first define a sequence of functions which, given a query (v, j) to G_φ, determine adaptively the appropriate sequence of queries to φ needed to answer query (v, j). By construction of G_φ we can determine all neighbours of any vertex in $v \in \{x_i, \neg x_i, \tilde{x}_i, x_i^c\}$ by knowing that $\mathrm{ans}_\varphi(x, i) = \ell^1 \vee \ell^2 \vee \ell^3$ and knowing for every other variable y contained in $\ell^1 \vee \ell^2 \vee \ell^3$ for which $j \in \{1, \ldots, d'\}$ we have that $\mathrm{ans}_\varphi(x, i) = \mathrm{ans}_\varphi(y, j)$. Hence the sequence of queries required to answer (v, i) for $v \in \{x_i, \neg x_i, \tilde{x}_i, x_i^c\}$ is to first query (x, i) and then query (y, j) for every variable y contained in $\mathrm{ans}_\varphi(x, i)$ and every $j \in \{1, \ldots, d'\}$. On the other hand we can now define a function which, given a query (v, i) to G_φ and given this sequence of queries and their respective answers, provides the answer to query (v, i). Since the number of queries to φ needed to answer a query to G_φ is $2d' + 1$ which is independent of n we get that (LR3) holds.

To prove (LR4) we assume that φ is satisfiable and that $\alpha : \textsc{Var}(\varphi) \to \{0, 1\}$ is a satisfying assignment of the variables of φ. Now we define set $D \subseteq V(G_\varphi)$ as follows. Let

$$D := \Big\{x_i \;\Big|\; i \in \{1, \ldots, d'\}, \alpha(x) = 1\Big\} \cup \Big\{\neg x_i \;\Big|\; i \in \{1, \ldots, d'\}, \alpha(x) = 0\Big\}.$$

As G_φ has $4d'n$ vertices and D contains d' vertices for each of the n variables of φ we get that $|D| = \frac{|V(G_\varphi)|}{4}$. We now argue that D is a dominating set of G_φ. Since for each of the cycles of G_φ the set D contains every third vertex of the cycle, clearly every vertex on any of the cycles is dominated by D. Now let x be any variable of φ and $1 \le i \le d'$. If $\mathrm{ans}(x, i) = \ell^1 \vee \ell^2 \vee \ell^3$ then there must be one literal $\ell \in \{\ell^1, \ell^2, \ell^3\}$ such that ℓ evaluates to true under the assignment α. Assume $\ell \in \{y, \neg y\}$ for some variable y and let $1 \le j \le d'$ be the index such that $\mathrm{ans}_\varphi(y, j) = \mathrm{ans}_\varphi(x, i)$. Since ℓ evaluates to true under the assignment α we get that $\ell_j \in D$ by construction of D. Since additionally $\{\ell_j, x_i^c\} \in E(G_\varphi)$ the vertex x_i^c is dominated by D. On the other hand if $\mathrm{ans}(x, i) = \perp$ then x_i^c is dominated by D as $\{x_i, x_i^c\}, \{\neg x_i, x_i^c\} \in E(G_\varphi)$ and either x_i or $\neg x_i$ is in D. Therefore all clause vertices are dominated by D and D is a dominating set of G_φ.

To prove (LR5) let $\epsilon' := g(\epsilon) := \frac{\epsilon}{12(d')^2}$. We argue by contraposition. Therefore assume G_φ is ϵ'-close to having a dominating set containing no more than $\frac{1}{4}$ of the vertices. Hence there is a set $E' \subseteq \{e \subseteq V(G_\varphi) \mid |e| = 2\}$ with $|E'| \le \epsilon' d |V(G_\varphi)|$ such that the graph $(V(G_\varphi), E(G_\varphi) \triangle E')$ has a dominating set D with $|D| \le \frac{|V(G_\varphi)|}{4}$. Let B be the set of all variables $x \in \textsc{Var}(\varphi)$ which satisfy the following two conditions

1. $\{x_i, \neg x_i, \tilde{x}_i, x_i^c \mid 1 \le i \le d'\} \cap e \neq \emptyset$ for some $e \in E'$,
2. $|\{x_i, \neg x_i, \tilde{x}_i, x_i^c \mid 1 \le i \le d'\} \cap D| \neq d'$.

As $|E'| \le \epsilon' d |V(G_\varphi)|$ there can be at most $2\epsilon' d |V(G_\varphi)|$ variables with the first property. Furthermore, since the minimum size of a dominating set of any cycle with $3d'$ vertices is d', any edge in E' can produce at most one variable for which the first condition does not hold and the second condition holds. Hence in total $|B| \leq 3\epsilon' d' |V(G_\varphi)|$. Now let φ' be the formula obtained from φ by removing every clause containing at least one variable from B. Now we define the following assignment $\alpha : \textsc{Var}(\varphi) \to \{0,1\}$ by

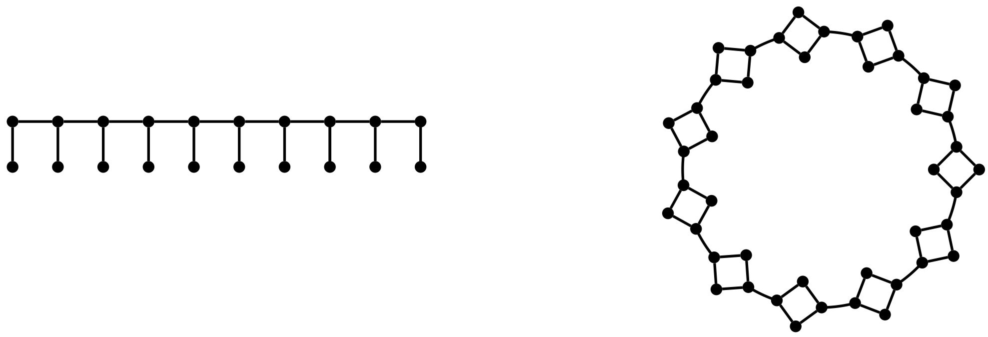
$$\alpha(x) := \begin{cases} 0 & \text{if } x \in B, \\ 1 & \text{if } x \notin B \text{ and } x_1 \in D, \\ 0 & \text{if } x \notin B \text{ and } x_1 \notin D. \end{cases}$$

Now assume that $\ell^1 \vee \ell^2 \vee \ell^3$ is any clause in φ' and assume that $\mathrm{ans}_\varphi(x,i) = \mathrm{ans}_\varphi(y,j) = \mathrm{ans}_\varphi(z,k) = \ell^1 \vee \ell^2 \vee \ell^3$. Hence x, y and z are not contained in B by construction of φ' and therefore $\{x_i, \neg x_i, \tilde{x}_i, x_i^c \mid 1 \leq i \leq d'\} \cap D$ must be either $\{x_i \mid 1 \leq i \leq d'\}$ or $\{\neg x_i \mid 1 \leq i \leq d'\}$ or $\{\tilde{x}_i \mid 1 \leq i \leq d'\}$. Since D therefore cannot contain x_i^c and D is a dominating set the intersection of the set of neighbours of x_i^c with D must be not empty, i. e. $\{\ell_i^1, \ell_j^2, \ell_k^3\} \cap D \neq \emptyset$. Assume without loss of generality that $\ell_i^1 \in D$. Then if $\ell^1 = x$ then $x_i \in D$ which implies that $x_1 \in D$ and hence α satisfies $\ell^1 \vee \ell^2 \vee \ell^3$. On the other hand if $\ell^1 = \neg x$ then $\neg x_i \in D$ which implies that x_1 can not be in D and hence α satisfies $\ell^1 \vee \ell^2 \vee \ell^3$. Therefore α satisfies every clause of φ' and hence φ' is satisfiable. As we obtained φ' by removing no more than $d'|B| \leq 12(d')^2\epsilon' d' n$ clauses from φ we get that φ is ϵ-close to being satisfiable. □

9.2 Lower bound for testing Hamiltonicity with one-sided error and an explicit construction of hard instances

In this section we study graphs that are far from being Hamiltonian, with the aim of finding hard instances for property testing Hamiltonicity in the bounded degree model. Here we call a graph G *Hamiltonian* if G has a *Hamiltonian cycle*, i. e. a cycle which contains every vertex of G. For fixed $d \in \mathbb{N}$ the property Hamiltonicity on $\mathcal{C}_d$ contains all Hamiltonian graphs of bounded degree d.

First note that it is easy to find graphs that are far from being Hamiltonian. For example, let G be a caterpillar graph on $n = 2k$ vertices as shown in Figure 9.2 for $k = 10$ (i. e. G is a path of length k where every vertex has a pendant edge). With a degree bound of at most 3, G is $\frac{1}{13}$-far from being Hamiltonian, because $\frac{n}{4}$ edges need to be added to make G 2-connected. As another example, consider the graph H consisting of k 4-cycles (C_4's) arranged in a cycle as shown in Figure 9.2 for $k = 9$. Assume $k > 1$. The graph H has $n = 4k$ vertices and, with a degree bound of 3, H is $\frac{1}{25}$-far from being Hamiltonian. This is because any Hamiltonian cycle in a graph has to traverse both edges incident to any vertex of degree 2. Hence in H a Hamiltonian cycle would have to traverse all four edges of every C_4. To avoid this we have to increase the degree of at least one of the degree 2 vertices for every C_4 and hence we have to



(a) Caterpillar (b) C_4's arranged in a cycle.

Figure 9.2: Example graphs which are far from being Hamiltonian but are not locally Hamiltonian

add at least $\frac{n}{8}$ edges to make H Hamiltonian.

In both examples it is possible to see locally, in the neighbourhood of a constant number of vertices, that the graphs are not Hamiltonian. We ask whether there exist graphs that locally look as if they might be Hamiltonian, but globally they are far from being Hamiltonian, and we give a positive answer to this. More precisely, we construct a sequence of graphs $(G_N)_{N\in\mathbb{N}}$ of increasing order such that there are constants $\epsilon, \delta \in (0,1)$ for which every G_N is ϵ-far from being Hamiltonian but the 1-neighbourhood of any δ-fraction of vertices of G_N appears in some Hamiltonian graph. Hence non-Hamiltonicity can not be observed locally, even looking at a large portion of G_N. We call this notion local Hamiltonicity and define it formally below. The existence of the sequence $(G_N)_{N\in\mathbb{N}}$ has implications in property testing. We can show the known linear lower bound for property testing Hamiltonicity in the case of one-sided error testers for which our constructed sequence $(G_N)_{N\in\mathbb{N}}$ yields a sequence of hard instances.

Definition 9.2.1 (Locally Hamiltonian)**.** Let $\mathcal{C}$ be a class of graphs and let $\delta \in (0,1]$. A graph $G \in \mathcal{C}$ is called *δ-locally Hamiltonian on $\mathcal{C}$* if for every set $S \subseteq V(G)$ of at most $\delta\cdot|V(G)|$ vertices there is a Hamiltonian graph $H := H_S \in \mathcal{C}$ with $|V(H)| = |V(G)|$, a subset $T := T_S \subseteq V(H)$ and an isomorphism from $G[N_1^G(S)]$ to $H[N_1^H(T)]$ which maps S onto T.

Note that by relaxing $|V(G)| = |V(H)|$ to $|V(H)| > \delta|V(G)|$ we get an equivalent definition in the sense that the same graphs are δ-locally Hamiltonian.

Remark 9.2.2. Let $\mathcal{C}$ be a graph class. Every Hamiltonian graph in $\mathcal{C}$ is δ-locally Hamiltonian for every $\delta \in (0,1]$. And every graph $G \in \mathcal{C}$ is 1-locally Hamiltonian iff G is Hamiltonian.

Let $d \geq 2$. A graph on n vertices is $\frac{1}{n}$-locally Hamiltonian on $\mathcal{C}_d$ iff the minimum degree of G is greater than 1.

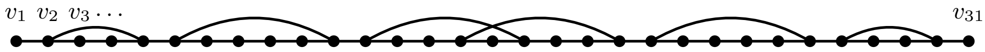


Figure 9.3: Illustration of $P(v_1, \ldots, v_{31})$.

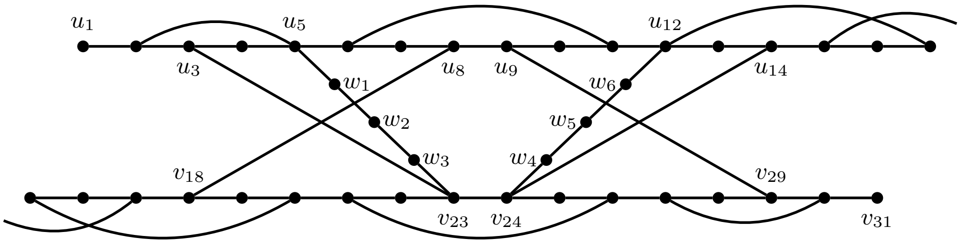


Figure 9.4: A link from $P(u_1, \ldots, u_{31})$ to $P(v_1, \ldots, v_{31})$ via $w_1, \ldots, w_6$.

9.2.1 Construction

In this section we introduce the main step of our construction of graphs which are locally Hamiltonian and far from being Hamiltonian. At a high level, we construct a graph $G_{\mathcal{E}}$ by choosing a d-regular base graph $\mathcal{E}$ and building $G_{\mathcal{E}}$ by introducing a path-gadget for every edge of $\mathcal{E}$, connecting these path-gadgets into a large cycle and linking path gadgets together if the edges of $\mathcal{E}$ corresponding to the path gadgets are incident to the same vertex. We give the precise construction in the following.

First we create a gadget (see Figure 9.3 for illustration). Let $\{v_1, \ldots, v_{31}\}$ be a set of vertices. Then we let $P(v_1, \ldots, v_{31})$ be the graph with vertex set $\{v_1, \ldots, v_{31}\}$ and edge set

$$\big\{\{v_i, v_{i+1}\}, \{v_j, v_{j+3}\}, \{v_k, v_{k+5}\} \mid i \in \{1, \ldots, 30\}, j \in \{2, 30\}, k \in \{6, 12, 15, 21\}\big\}.$$

For a graph G for which $\{u_1, \ldots, u_{31}, v_1, \ldots, v_{31}, w_1, \ldots, w_6\} \subseteq V(G)$ and it holds that $G[u_1, \ldots, u_{31}] = P(u_1, \ldots, u_{31})$ and $G[v_1, \ldots, v_{31}] = P(v_1, \ldots, v_{31})$ we say that G contains a *link from* $P(u_1, \ldots, u_{31})$ *to* $P(v_1, \ldots, v_{31})$ *via* $w_1, \ldots, w_6$ (see Figure 9.4 for illustration), if $E(G)$ contains

$$\Big\{\{u_3, v_{23}\}, \{v_{18}, u_8\}, \{u_9, v_{29}\}, \{v_{24}, u_{14}\}, \{u_5, w_1\}, \{w_1, w_2\}, \{w_2, w_3\}, \{w_3, v_{23}\},$$
$$\{v_{24}, w_4\}, \{w_4, w_5\}, \{w_5, w_6\}, \{w_6, u_{12}\}\Big\}.$$

Finally to any graph G we associate a directed graph $\vec{G}$ which is the graph that is obtained from G by replacing every edge $\{u, v\} \in E(G)$ by the two directed edges (u, v) and (v, u). We can now define the graph construction.

Definition 9.2.3. Let $\mathcal{E}$ be a d-regular graph (the *base graph*) and $f : E(\vec{\mathcal{E}}) \to \{1, \dots, |E(\vec{\mathcal{E}})|\}$ be any linear order on $E(\vec{\mathcal{E}})$. We define the graph $G_{\mathcal{E}}$ as follows.

$$V(G_{\mathcal{E}}) := \{a_1^e, \dots, a_{31}^e \mid e \in E(\vec{\mathcal{E}})\} \cup \{b_1^v, \dots, b_6^v \mid v \in V(\vec{\mathcal{E}})\}.$$

$E(G_{\mathcal{E}})$ consists of the minimum set of edges such that

- $G_{\mathcal{E}}[a_1^e, \dots, a_{31}^e] = P(a_1^e, \dots, a_{31}^e)$ for every $e \in E(\vec{\mathcal{E}})$,
- $a_{31}^{f^{-1}(i)}$ is adjacent to $a_1^{f^{-1}(j)}$ for every $i \in [|E(\vec{\mathcal{E}})|]$, $j := 1$ if $i = |E(\vec{\mathcal{E}})|$ and $j := i+1$ otherwise and
- $G_{\mathcal{E}}$ contains a link from $P(a_1^{(v,w)}, \dots, a_{31}^{(v,w)})$ to $P(a_1^{(u,v)}, \dots, a_{31}^{(u,v)})$ via $b_1^v, \dots, b_6^v$ for every triple of vertices $u, v, w \in V(\vec{\mathcal{E}})$ with $(u, v), (v, w) \in E(\vec{\mathcal{E}})$.

See Figure 9.5 for an illustration. Note that the construction of $G_{\mathcal{E}}$ depends on f as well as $\mathcal{E}$, but since the properties of $G_{\mathcal{E}}$ are independent of which linear order f we use, we omit the dependency on f.

Remark 9.2.4. If $\mathcal{E}$ is d-regular, for $d \geq 1$, and $|V(\mathcal{E})| = n$, then the degree of $G_{\mathcal{E}}$ is at most $d+3$ and $|V(G_{\mathcal{E}})| = (6+31d)n$.

Note 9.2.5. $G_{\mathcal{E}}$ contains a large cycle of length $31dn$, i. e., the cycle

$$(\dots\dots, a_{31}^{f^{-1}(i-1)}, a_1^{f^{-1}(i)}, a_2^{f^{-1}(i)}, \dots, a_{31}^{f^{-1}(i)}, a_1^{f^{-1}(i+1)}, \dots\dots).$$

However $G_{\mathcal{E}}$ also contains $6n$ vertices which are not part of this cycle.

9.2.2 The construction is far from being Hamiltonian

In this section we prove the following.

Theorem 9.2.6. *For every $d \in \mathbb{N}_{>1}$ there is $\epsilon = \epsilon(d) \in (0, 1)$ such that for any d-regular graph $\mathcal{E}$ the graph $G_{\mathcal{E}}$ constructed in Definition 9.2.3 is ϵ-far from being Hamiltonian.*

For technical reasons we use this slightly unusual definition of a subpath of a cycle in which it does not matter whether the subpath appears in the cycle or the reversed cycle. Let G be an undirected graph. A path $(s_0, \dots, s_k)$ in G is a *subpath* of a cycle $(c_0, \dots, c_\ell)$ in G if there is an index $0 \leq i \leq \ell$ such that $(s_0, \dots, s_k)$ is either a subpath of the path $(c_i, \dots, c_\ell, c_1, \dots, c_{i-1})$ or of the path $(c_{i-1}, \dots, c_1, c_\ell, \dots, c_i)$. Here a subpath of a path is defined in the usual way (see Section 2).

To prove Theorem 9.2.6 we require the two technical Lemmas below (Lemma 9.2.8 and Lemma 9.2.9). They will be applied to graphs G obtained from $G_{\mathcal{E}}$ by modifying a small fraction of the edges of $G_{\mathcal{E}}$. Therefore they are stated for graphs G which share certain induced

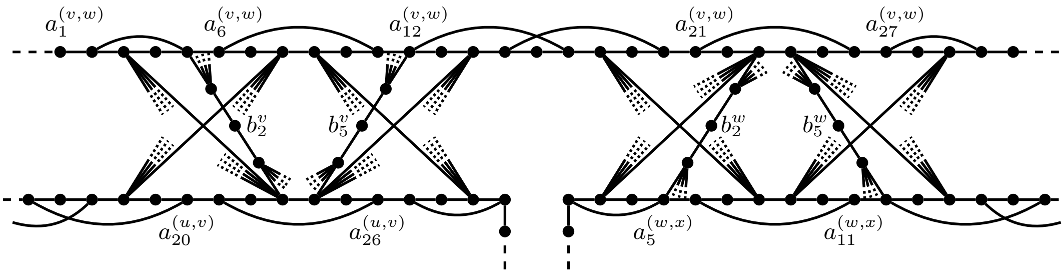


Figure 9.5: Close-up of $G_{\mathcal{E}}$ with vertices of high degree ($d+1$, $d+2$ or $d+3$) indicated by 'fans'.

subgraphs with $G_{\mathcal{E}}$. The first of the two Lemmas (Lemma 9.2.8) states that if G has a Hamiltonian cycle and a certain induced subgraph, which also appears in $G_{\mathcal{E}}$, then the Hamiltonian cycle has certain subpaths. The proof of Lemma 9.2.8 is illustrated in Figure 9.5. We will use the following easy observation in the proof of Lemma 9.2.8.

Remark 9.2.7. Let G be a graph, $u \in V(G)$ a vertex of degree 2 and v, w the two neighbours of u. Then any cycle C containing the vertices u, v and w must contain (v, u, w) as a subpath.

Lemma 9.2.8. *Let $\mathcal{E}$ be any d-regular graph and $G_{\mathcal{E}}$ as defined in Definition 9.2.3. Pick $v \in V(\vec{\mathcal{E}})$ and let $S_v := \{a_i^e \mid e \in E(\vec{\mathcal{E}}), e \text{ is incident to } v\} \cup \{b_1^v, \ldots, b_6^v\}$. Let G be a graph with $S_v \subseteq V(G)$. Assume $G_{\mathcal{E}}[N_1^{G_{\mathcal{E}}}(S_v)] \cong G[N_1^G(S_v)]$ and $f : S_v \to S_v$ defined by $f(v) = v$ for $v \in S_v$ is an isomorphism from $G_{\mathcal{E}}[S_v]$ to $G[S_v]$. Then for every Hamiltonian cycle C in G and every edge $e \in V(\vec{\mathcal{E}})$ incident to v the following properties hold.*

1. *Either $(a_1^e, \ldots, a_5^e)$ or $(a_1^e, a_2^e, a_5^e, a_4^e, a_3^e)$ is a subpath of C.*
2. *Either $(a_{27}^e, \ldots, a_{31}^e)$ or $(a_{29}^e, a_{28}^e, a_{27}^e, a_{30}^e, a_{31}^e)$ is a subpath of C.*
3. *Either $(a_{12}^e, \ldots, a_{20}^e)$ or $(a_{14}^e, a_{13}^e, a_{12}^e, a_{17}^e, a_{16}^e, a_{15}^e, a_{20}^e, a_{19}^e, a_{18}^e)$ is a subpath of C.*
4. *If $e \in E_G^+(v)$ then either $(a_6^e, \ldots, a_{11}^e)$ or $(a_8^e, a_7^e, a_6^e, a_{11}^e, a_{10}^e, a_9^e)$ is a subpath of C.*
5. *If $e \in E_G^-(v)$ then either $(a_{21}^e, \ldots, a_{36}^e)$ or $(a_{23}^e, a_{22}^e, a_{21}^e, a_{26}^e, a_{25}^e, a_{24}^e)$ is a subpath of C.*

Proof. To prove (1) let us observe that a_1^e and a_4^e have degree 2 in G, as $G_{\mathcal{E}}[N_1^{G_{\mathcal{E}}}(S_v)] \cong G[N_1^G(S_v)]$ and a_1^e and a_4^e have degree 2 in $G_{\mathcal{E}}$. Hence (a_1^e, a_2^e) and (a_3^e, a_4^e, a_5^e) have to be subpaths of C as in Remark 9.2.7. Since a_2^e has exactly three neighbours a_1^e, a_3^e and a_5^e in $G_{\mathcal{E}}$ and $G_{\mathcal{E}}[N_1^{G_{\mathcal{E}}}(S_v)] \cong G[N_1^G(S_v)]$ we get that either $(a_1^e, \ldots, a_5^e)$ is a subpath of C or $(a_1^e, a_2^e, a_5^e, a_4^e, a_3^e)$ is a subpath of C. Property (2) follows with a similar argumentation.

For (3) let us assume that neither $(a_{12}^e, \ldots, a_{17}^e)$ nor $(a_{14}^e, a_{13}^e, a_{12}^e, a_{17}^e, a_{16}^e, a_{15}^e)$ appear in C as a subpath. Since both a_{13}^e and a_{16}^e have degree 2 in G, we know that $(a_{12}^e, a_{13}^e, a_{14}^e)$ and $(a_{15}^e, a_{16}^e, a_{17}^e)$ are subpaths of C. Hence neither (a_{14}^e, a_{15}^e) nor (a_{12}^e, a_{17}^e) are subpaths of C. Since both a_{15}^e and a_{17}^e have degree 3 in G, this implies that $(a_{20}^e, a_{15}^e, a_{16}^e, a_{17}^e, a_{18}^e)$ is a subpath of C. Since a^e_{19} has degree 2, then $(a^e_{20}, a^e_{15}, a^e_{16}, a^e_{17}, a^e_{18}, a^e_{19}, a^e_{20})$ has to be a subpath of C. Since this is a cycle, C must be equal to $(a^e_{20}, a^e_{15}, a^e_{16}, a^e_{17}, a^e_{18}, a^e_{19}, a^e_{20})$ which contradicts the assumption that S_v is contained in C. A symmetric argument shows that either $(a^e_{15}, \dots, a^e_{20})$ or $(a^e_{17}, a^e_{16}, a^e_{15}, a^e_{20}, a^e_{19}, a^e_{18})$ has to be a subpath of C, proving (3).

We will prove (4) and (5) simultaneously using a counting argument. Let us first observe that for every edge $e \in E(\vec{\mathcal{E}})$ incident to v we know that (a^e_6, a^e_7, a^e_8), $(a^e_9, a^e_{10}, a^e_{11})$, $(a^e_{21}, a^e_{22}, a^e_{23})$ and $(a^e_{24}, a^e_{25}, a^e_{26})$ are subpaths of C, because a^e_7, a^e_{10}, a^e_{22} and a^e_{25} have degree 2 in G. Let S be the set of all maximal subpaths of C which only contain vertices from $\{a^e_{21}, \dots, a^e_{26}, a^{\tilde{e}}_6, \dots, a^{\tilde{e}}_{11} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\}$. Since there are no edges of the form $\{a^e_i, a^{\tilde{e}}_j\}$ for $i, j \in \{6, \dots, 11, 21, \dots, 26\}$, $e \neq \tilde{e} \in E(\vec{\mathcal{E}})$, every subpath in S is either of length 3 or length 6. For every path $P = (p_1, \dots, p_\ell) \in S$, we define the vertices u_P, w_P to be the neighbours of P on C, i.e. $(u_P, p_1, \dots, p_\ell, w_P)$ is a subpath of C. Since $G_{\mathcal{E}}[N^{G_{\mathcal{E}}}_1(S_v)] \cong G[N^G_1(S_v)]$ and every path $P \in S$ is maximal, we know that $u_P, w_P \in \{a^e_{18}, a^e_{20}, a^e_{27}, a^e_{29}, a^{\tilde{e}}_3, a^{\tilde{e}}_5, a^{\tilde{e}}_{12}, a^{\tilde{e}}_{14} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\} \cup \{b^v_3, b^v_4\}$. Properties (1),(3) imply that for every edge $e \in E^-_{\vec{\mathcal{E}}}(v)$, only one of the two vertices a^e_{18}, a^e_{20} and only one of the two vertices a^e_{27}, a^e_{29} can be in the set $\{u_P, w_P \mid P \in S\}$. Similarly, (2),(3) imply that for every edge $e \in E^+_{\vec{\mathcal{E}}}(v)$ only one of the two vertices a^e_3, a^e_5 and only one of the two vertices a^e_{12}, a^e_{14} can be in the set $\{u_P, w_P \mid P \in S\}$. Furthermore there are two not necessarily distinct edges $e, \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)$ such that $(a^e_1, \dots, a^e_5, b^v_1)$ and $(b^v_6, a^{\tilde{e}}_{12}, \dots, a^{\tilde{e}}_{20})$ are subpaths of C and hence the vertices $a^e_3, a^e_5, a^{\tilde{e}}_{12}, a^{\tilde{e}}_{14}$ cannot be in $\{u_P, w_P \mid P \in S\}$. Hence $|\{u_P, w_P \mid P \in S\}| \leq 2|E^+_{\vec{\mathcal{E}}}(v)| - 2 + 2|E^-_{\vec{\mathcal{E}}}(v)| + 2 = 4d$. In addition, note that (1), (2), (3) and $\deg_G(b^v_2) = 2$ and $\deg_G(b^v_5) = 2$ imply that no maximal subpath of C only containing vertices in $S_v \setminus \{a^e_{21}, \dots, a^e_{26}, a^{\tilde{e}}_6, \dots, a^{\tilde{e}}_{11} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\}$ has length at most 1 and hence $|\{u_P, w_P \mid P \in S\}| = 2|S|$. Therefore $|S| \leq 2d$. If any path in S has length 3 then $|S| > 2d$, since $|\{a^e_{21}, \dots, a^e_{26}, a^{\tilde{e}}_6, \dots, a^{\tilde{e}}_{11} \mid e \in E^-_{\vec{\mathcal{E}}}(v), e' \in E^+_{\vec{\mathcal{E}}}(v)\}| = 12d$. This yields a contradiction and hence (4) and (5) are true. $\square$

Let G be a graph with $a^e_i, \dots, a^e_j \in V(G)$ for some edge $e \in E(\vec{\mathcal{E}})$ and $1 \leq i \leq j \leq 31$. Assume C is a cycle in G which contains $a^e_i, \dots, a^e_j$. We say that C *traverses* the vertices $a^e_i, \dots, a^e_j$ *in order* if $(a^e_i \dots, a^e_j)$ is a subpath of C and we say that C *traverses* $a^e_i, \dots, a^e_j$ *out of order* otherwise. Note that for certain $1 \leq i \leq j \leq 31$ and $e \in E(\vec{\mathcal{E}})$ there is only one way in which a cycle C can traverse $a^e_i, \dots, a^e_j$ out of order (as specified in Lemma 9.2.8).

The next lemma shows that for every vertex $v \in V(\vec{\mathcal{E}})$ and every Hamiltonian cycle C in $G_{\mathcal{E}}$ the number of edges $e \in E^-_{\vec{\mathcal{E}}}(v)$ for which C traverses $a^e_{12}, \dots, a^e_{20}$ out of order is exactly one larger than the number of edges $\tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)$ for which C traverses $a^{\tilde{e}}_{12}, \dots, a^{\tilde{e}}_{20}$ out of order. This still holds for every graph G which contains a certain induced subgraph of $G_{\mathcal{E}}$.

Lemma 9.2.9. *Let $\mathcal{E}$ be any d-regular graph and $G_{\mathcal{E}}$ as defined in Definition 9.2.3. Let $S_v := \{a_i^e \mid e \in E(\vec{\mathcal{E}}), e \text{ is incident to } v\} \cup \{b_1^v, \ldots, b_6^v\}$ for some $v \in V(\vec{\mathcal{E}})$. Let G be a graph with $S_v \subseteq V(G)$. Assume $G_{\mathcal{E}}[N_1^{G_{\mathcal{E}}}(S_v)] \cong G[N_1^G(S_v)]$ and $f : S_v \to S_v$ defined by $f(v) = v$ for $v \in S_v$ is an isomorphism from $G_{\mathcal{E}}[S_v]$ to $G[S_v]$. Then for every Hamiltonian cycle C in G the cardinalities of the two sets*

$$T_{v,C}^{\text{in}} := \left\{e \in E_{\vec{\mathcal{E}}}^-(v) \mid (a_{12}^e, a_{17}^e) \text{ is a subpath of } C\right\} \text{ and} \tag{9.1}$$

$$T_{v,C}^{\text{out}} := \left\{e \in E_{\vec{\mathcal{E}}}^+(v) \mid (a_{12}^e, a_{17}^e) \text{ or } (a_{12}^e, b_6^v) \text{ is a subpath of } C\right\} \tag{9.2}$$

are equal.

Proof. Note that the condition $G_{\mathcal{E}}[N_1^{G_{\mathcal{E}}}(S_v)] \cong G[N_1^G(S_v)]$ implies that no vertex in the set $\{a_{15}^e, \ldots, a_{30}^e, a_2^{\tilde{e}}, \ldots, a_{17}^{\tilde{e}} \mid e \in E_{\vec{\mathcal{E}}}^-(v), \tilde{e} \in E_{\vec{\mathcal{E}}}^+(v)\} \cup \{b_1^v, \ldots, b_6^v\}$ has neighbours in $G \setminus S_v$. This will implicitly be used in the following argument whenever we exhaustively consider neighbours of vertices in G as successors on C.

Let us first define a map $F_{v,C} : T_{v,C}^{\text{in}} \to T_{v,C}^{\text{out}}$, given by $F_{v,C}(e) := \tilde{e}$, where $\tilde{e} \in T_{v,C}^{\text{out}}$ is the edge such that $(a_{18}^e, a_8^{\tilde{e}})$ is a subpath of C. We first have to argue that $F_{v,C}$ is well defined.

By Lemma 9.2.8 (3), $e \in T_{v,C}^{\text{in}}$ implies that $(a_{14}^e, a_{13}^e, a_{12}^e, a_{17}^e, a_{16}^e, a_{15}^e, a_{20}^e, a_{19}^e, a_{18}^e)$ is a subpath of C. Since the two neighbours a_{17}^e and a_{19}^e of a_{18}^e are already part of this subpath this implies that $(a_{18}^e, a_8^{\tilde{e}})$ has to be a subpath of C for some edge $\tilde{e} \in E_{\vec{\mathcal{E}}}^+(v)$. This implies that $(a_6^{\tilde{e}}, \ldots, a_{11}^{\tilde{e}})$ cannot be a subpath of C and hence, by Lemma 9.2.8 (4), $(a_8^{\tilde{e}}, a_7^{\tilde{e}}, a_6^{\tilde{e}}, a_{11}^{\tilde{e}}, a_{10}^{\tilde{e}}, a_9^{\tilde{e}})$ has to be a subpath of C. This further implies that $(a_{11}^{\tilde{e}}, a_{12}^{\tilde{e}})$ cannot be a subpath of C. If $(a_{12}^{\tilde{e}}, \ldots, a_{20}^{\tilde{e}})$ is a subpath of C then $(a_{12}^{\tilde{e}}, b_6^v)$ has to be a subpath of C by excluding all possible other neighbours of $a_{12}^{\tilde{e}}$. On the other hand, if $(a_{12}^{\tilde{e}}, \ldots, a_{20}^{\tilde{e}})$ is not a subpath of C then, by Lemma 9.2.8 (3), $(a_{14}^{\tilde{e}}, a_{13}^{\tilde{e}}, a_{12}^{\tilde{e}}, a_{17}^{\tilde{e}}, a_{16}^{\tilde{e}}, a_{15}^{\tilde{e}}, a_{20}^{\tilde{e}}, a_{19}^{\tilde{e}}, a_{18}^{\tilde{e}})$ is a subpath of C and hence $(a_{12}^{\tilde{e}}, a_{17}^{\tilde{e}})$ is a subpath of C. Therefore $\tilde{e} \in T_{v,C}^{\text{out}}$. This shows that $F_{v,C}$ is well defined.

Furthermore $F_{v,C}$ is injective since if $(a_{18}^e, a_8^{\tilde{e}})$ and $(a_{18}^e, a_8^{e'})$ are subpaths of C then $\tilde{e} = e'$ because (a_{19}^e, a_{18}^e) is also a subpath of C. $F_{v,C}$ is surjective as for $\tilde{e} \in T_{v,C}^{\text{out}}$ both $(a_{12}^{\tilde{e}}, a_{17}^{\tilde{e}})$ or $(a_{12}^{\tilde{e}}, b_6^v)$ being a subpath of C together with Lemma 9.2.8 (3) implies that $(a_{12}^{\tilde{e}}, a_{11}^{\tilde{e}})$ cannot be a subpath of C. This further implies that $(a_8^{\tilde{e}}, a_7^{\tilde{e}}, a_6^{\tilde{e}}, a_{11}^{\tilde{e}}, a_{10}^{\tilde{e}}, a_9^{\tilde{e}})$ is a subpath of C by Lemma 9.2.8 (4) and hence there is an edge $e \in E_{\vec{\mathcal{E}}}^-(v)$ such that $(a_{18}^e, a_8^{\tilde{e}})$ is a subpath of C. Then with the same argument as before (a_{12}^e, a_{17}^e) is a subpath of C and hence $e \in T_{v,C}^{\text{in}}$ and $F_{v,C}(e) = \tilde{e}$. Therefore $F_{v,C}$ is bijective which implies the statement of the lemma. □

As a direct consequence from Lemma 9.2.9 we get that $G_{\mathcal{E}}$ cannot be Hamilonian for any base graph $\mathcal{E}$. That is true because if there is a Hamiltonian cycle C in $G_{\mathcal{E}}$ then by Lemma 9.2.9 the equation $\sum_{v \in V(\vec{\mathcal{E}})} |T_{v,C}^{\text{in}}| = \sum_{v \in V(\vec{\mathcal{E}})} |T_{v,C}^{\text{out}}|$ must hold. But since every edge in $T_{v,C}^{\text{in}}$ is also contained in $T_{v,C}^{\text{out}}$ and $T_{v,C}^{\text{out}}$ must contain some edges (all the edges (v,w) for which $(a_{12}^{(v,w)}, b_6^v)$ is a subpath of C) that are not contained in $T_{v,C}^{\text{in}}$, the equation cannot hold and hence $G_{\mathcal{E}}$ cannot be Hamiltonian. This argument works similarly if a small number of edges in $G_{\mathcal{E}}$ have been altered and the equality from Lemma 9.2.9 still has to hold for many vertices. This will be our proof strategy for Theorem 9.2.6.

Proof of Theorem 9.2.6. Let $\epsilon := \frac{1}{(8(d+3)^2(6+31d))}$. Assume $\mathcal{E}$ is d-regular and $n := |V(\mathcal{E})|$. Let $n' := V(G_{\mathcal{E}}) = (6+31d)n$ and $d' := d+3$ the degree of $G_{\mathcal{E}}$.

Towards a contradiction let us assume that $G_{\mathcal{E}}$ is not ϵ-far to being Hamiltonian and let E be a set of edges such that $|E| \leq \epsilon d'n'$ and the graph $G := (V(G_{\mathcal{E}}), E(G_{\mathcal{E}}) \triangle E)$ is Hamiltonian. Let $B \subseteq V(\vec{\mathcal{E}})$ be the set of vertices defined by

$$\begin{aligned} B :=& \{v \in V(\vec{\mathcal{E}}) \mid \text{there is } e \in E, i \in \{1, \dots, 31\}, \tilde{e} \in E_G^-(v) \cup E_G^+(v) \text{ such that } a_i^{\tilde{e}} \in e\} \\ &\cup \{v \in V(\vec{\mathcal{E}}) \mid \text{there is } e \in E, i \in \{1, \dots, 6\} \text{ such that } b_i^v \in e\}. \end{aligned}$$

Note that $|B| \leq 4 \cdot \epsilon d'n'$, because every edge $e \in E$ contributes at most 4 vertices to B, and hence $|V(\vec{\mathcal{E}}) \setminus B| \geq n - 4\epsilon d'n' > \frac{n}{2}$.

Let C be a Hamiltonian cycle in G. Then for every vertex $v \in V(\vec{\mathcal{E}}) \setminus B$ we have that $S_v \subseteq V(G)$, $G_{\mathcal{E}}[N_1^{G_{\mathcal{E}}}(S_v)] \cong G[N_1^G(S_v)]$ and $f : S_v \to S_v$ defined by $f(v) = v$ for $v \in S_v$ is an isomorphism from $G_{\mathcal{E}}[S_v]$ to $G[S_v]$ where $S_v := \{a_i^e \mid e \in E(\vec{\mathcal{E}}), e \text{ is incident to } v\} \cup \{b_1^v, \dots, b_6^v\}$. Since C is Hamiltonian, C contains all vertices in S_v for every $v \in V(\vec{\mathcal{E}}) \setminus B$ (amongst others). Hence by Lemma 9.2.9 we have $|T_{v,C}^{\text{in}}| = |T_{v,C}^{\text{out}}|$ for every $v \in V(\vec{\mathcal{E}}) \setminus B$ where $T_{v,C}^{\text{in}}$ and $T_{v,C}^{\text{out}}$ are as defined in Equation 9.1 and Equation 9.2. Therefore

$$\sum_{v \in V(\vec{\mathcal{E}}) \setminus B} |T_{v,C}^{\text{in}}| = \sum_{v \in V(\vec{\mathcal{E}}) \setminus B} |T_{v,C}^{\text{out}}|. \tag{9.3}$$

As b_6^v has precisely one neighbour in G for every $v \in V(\vec{\mathcal{E}}) \setminus B$, which is not of the form a_{12}^e for some $e \in E_G^+(v)$ and this neighbour has degree 2 in G, we know that for precisely one edge $e \in E_G^+(v)$ the sequence (b_6^v, a_{12}^e) is a subpath of C. Hence

$$\sum_{v \in V(\vec{\mathcal{E}}) \setminus B} \left| \left\{ e \in E_G^+(v) \mid (a_{12}^e, b_6^v) \text{ is a subpath of } C \right\} \right| = \left| E(\vec{\mathcal{E}}) \setminus B \right| > \frac{n}{2}. \tag{9.4}$$

Since every edge $(u,v) \in E(\vec{\mathcal{E}})$ such that $u, v \in V(\vec{\mathcal{E}}) \setminus B$ contributes 1 to both sides of Equation 9.3, Equation 9.3 and Equation 9.4 imply that

$$\sum_{v \in V(\vec{\mathcal{E}}) \setminus B} \left| \left\{ (u,v) \in E(\vec{\mathcal{E}}) \mid u \in B, (a_{12}^{(u,v)}, a_{17}^{(u,v)}) \text{ is a subpath of } C \right\} \right| > \frac{n}{2}.$$

But this is a contradiction as the number of edges $(u,v) \in E(\vec{\mathcal{E}})$ for which $u \in B$ is bounded from above by $d'|B| \leq \frac{n}{2}$. $\square$

9.2.3 Ensuring local Hamiltonicity

In this Section we prove the following Theorem.

Theorem 9.2.10. *For any d-regular graph $\mathcal{E}$ with expansion ratio $h(\mathcal{E}) \geq 1$ the graph $G_{\mathcal{E}}$ constructed in Definition 9.2.3 is δ-locally Hamiltonian for some constant $\delta = \delta(d) \in (0,1]$.*

Our proof strategy for Theorem 9.2.10 is to add edges to $G_{\mathcal{E}}$ which are incident to at most one vertex in $N_1^{G_{\mathcal{E}}}(S)$ to obtain a graph H which is Hamiltonian, for any given $S \subseteq V(G_{\mathcal{E}})$ of size at most $\delta|V(G)|$. We prove the Hamiltonicity of H by dividing the vertex set of H into pairwise disjoint small sets. For each of these sets we obtain a set of vertex disjoint paths which cover the entire small set and start and end in prescribed vertices. To conclude the proof of the Hamiltonicity of H we find a Hamiltonian cycle by patching together these paths. The next Lemma will be used to show the existence of such paths for all those subsets of vertices of H which contain a vertex from S.

Lemma 9.2.11. *Let $\mathcal{E}$ be any d-regular graph and $G_{\mathcal{E}}$ as defined in Definition 9.2.3. Let $v \in V(\vec{\mathcal{E}})$ and $S_v := \{a^e_{18}, \ldots, a^e_{31}, a^{\tilde{e}}_1, \ldots, a^{\tilde{e}}_{17} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\} \cup \{b^v_1, \ldots, b^v_6\}$. Let G be a graph such that $G_{\mathcal{E}}[S_v]$ is a subgraph of G. Then for any two sets $T^{\text{in}}_v \subseteq E^-_{\vec{\mathcal{E}}}(v)$ and $T^{\text{out}}_v \subseteq E^+_{\vec{\mathcal{E}}}(v)$ with $|T^{\text{in}}_v| - 1 = |T^{\text{out}}_v|$ there is a set of $2d$ pairwise vertex disjoint simple paths $\{P^{\text{in}}_e, P^{\text{out}}_{\tilde{e}} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\}$ in G with the following properties.*

- *If $e \in T^{\text{in}}_v$ then P^{in}_e is a path from a^e_{20} to a^e_{31}.*
- *If $e \in E^-_{\vec{\mathcal{E}}}(v) \setminus T^{\text{in}}_v$ then P^{in}_e is a path from a^e_{18} to a^e_{31}.*
- *If $e \in T^{\text{out}}_v$ then P^{out}_e is a path from a^e_1 to a^e_{15}.*
- *If $e \in E^+_{\vec{\mathcal{E}}}(v) \setminus T^{\text{out}}_v$ then P^{out}_e is a path from a^e_1 to a^e_{17}.*
- *The set $\{x \in V(G) \mid x$ is contained in P^{in}_e or P^{out}_e for some $e\}$ is equal to S_v.*

Proof. First we pick a vertex $n(v) \in V(\vec{\mathcal{E}})$ such that $(v, n(v)) \notin T^{\text{out}}_v$. This is possible because v has the same number of incoming and outgoing edges and $|T^{\text{in}}_v| - 1 = |T^{\text{out}}_v|$. Then $|T^{\text{in}}_v| = |T^{\text{out}}_v \cup \{(v, n(v))\}|$, and hence we can find a bijection $g : T^{\text{in}}_v \to T^{\text{out}}_v \cup \{(v, n(v))\}$. Then we can define the paths as follows. For $e \in T^{\text{in}}_v$ we let

$$P^{\text{in}}_e := (a^e_{20}, a^e_{19}, a^e_{18}, a^{g(e)}_8, a^{g(e)}_7, a^{g(e)}_6, a^{g(e)}_{11}, a^{g(e)}_{10}, a^{g(e)}_9, a^e_{29}, a^e_{28}, a^e_{27}, a^e_{30}, a^e_{31}),$$

$$P^{\text{out}}_{g(e)} := (a^{g(e)}_1, \ldots, a^{g(e)}_5, b^v_1, b^v_2, b^v_3, a^e_{23}, a^e_{22}, a^e_{21}, a^e_{26}, a^e_{25}, a^e_{24}, b^v_4, b^v_5, b^v_6, a^{g(e)}_{12}, \ldots, a^{g(e)}_{17})$$

if $g(e) = (v, n(v))$ and

$$P^{\text{out}}_{g(e)} := (a^{g(e)}_1, a^{g(e)}_2, a^{g(e)}_5, a^{g(e)}_4, a^{g(e)}_3, a^e_{23}, a^e_{22}, a^e_{21}, a^e_{26}, a^e_{25}, a^e_{24}, a^{g(e)}_{14}, a^{g(e)}_{13}, a^{g(e)}_{12}, a^{g(e)}_{17}, a^{g(e)}_{16}, a^{g(e)}_{15})$$

if $g(e) \neq (v, n(v))$.

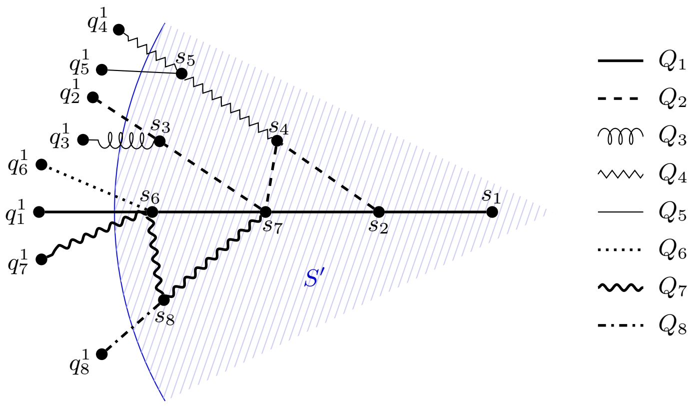


Figure 9.6: Set of edge disjoint path as in Claim 1.

Furthermore, for $e \in E^-_{\vec{\mathcal{E}}}(v) \setminus T_v^{\text{in}}$, we let $P_e^{\text{in}} := (a_{18}^e, \ldots, a_{31}^e)$ and for $e \in \{(v,w) \in E^+_{\vec{\mathcal{E}}}(v) \setminus T_v^{\text{out}}$, we let $P_e^{\text{out}} := (a_1^e, \ldots, a_{17}^e)$. These paths clearly satisfy all conditions. □

Proof of Theorem 9.2.10. Let $\delta := \frac{1}{(2\cdot(6+31d))}$ and let $S \subseteq V(G_{\mathcal{E}})$ be any set of vertices with $|S| \leq \delta \cdot |V(G_{\mathcal{E}})|$. We will find a Hamiltonian graph H by modifying $G_{\mathcal{E}}$ in such a way that $G_{\mathcal{E}}[N_1^{G_{\mathcal{E}}}(S)]$ is not affected by any modifications.

Let $S_v := \{a_{18}^e, \ldots, a_{31}^e, a_1^{\tilde{e}}, \ldots, a_{17}^{\tilde{e}} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\} \cup \{b_1^v, \ldots, b_6^v\}$ for every $v \in V(\vec{\mathcal{E}})$. Let $S' := \{v \in V(\vec{\mathcal{E}}) \mid S_v \cap S \neq \emptyset\}$. By Remark 9.2.4 $|V(G_{\mathcal{E}})| = (6+31d) \cdot |V(\mathcal{E})|$. Since the sets S_v are pairwise disjoint this implies that $|S'| \leq |S| \leq \delta \cdot |V(G_{\mathcal{E}})| = \frac{1}{2} \cdot |V(\mathcal{E})|$. Let $S' = \{s_1', \ldots, s_m'\}$ where $m := |S'|$.

Claim 1. There are pairwise edge disjoint paths $Q_1, \ldots, Q_m$ in $\mathcal{E}$ such that Q_i is of the form $Q_i = (q_i^1, \ldots, q_i^{\ell_i})$ for some $\ell_i \in \mathbb{N}$ and $q_i^{\ell_i} = s_i$, $q_i^j \in S'$ for all $j > 1$ and $q_i^1 \in V(\mathcal{E}) \setminus S'$.

For illustration at an example, see Figure 9.6.

Proof of Claim 1. By induction on the size of S'. If $|S'| = 1$ then this is trivially true. If $|S'| = n$ then $h(\mathcal{E}) \geq 1$ implies that there must be a vertex v with at least as many neighbours in $V(\mathcal{E}) \setminus S'$ as neighbours in S'. Then $S \setminus \{v\}$ has $n-1$ vertices. Hence by induction hypothesis there is such a set of paths for $S' \setminus \{v\}$. But then we can extend every path which starts in v by a different edge so it starts in $V(\mathcal{E}) \setminus S$. ■

Let $Q_1, \ldots, Q_m$ be as in Claim 1. Further, for every vertex $v \in V(\mathcal{E}) \setminus S'$ we pick one vertex $u \in V(\mathcal{E})$ with $(v,u) \in E(\vec{\mathcal{E}})$ and define $n(v) := u$. Now let E be the set

$$\left\{\left\{b_3^v, a_4^{(v,n(v))}\right\}, \left\{b_4^v, a_{13}^{(v,n(v))}\right\} \;\middle|\; v \in V(\mathcal{E}) \setminus S'\right\} \cup \left\{\left\{a_{14}^{(q_i^1,q_i^2)}, a_{17}^{(q_i^1,q_i^2)}\right\} \;\middle|\; 1 \leq i \leq m\right\}.$$

We now define the graph H by setting $V(H) := V(G_{\mathcal{E}})$ and $E(H) := E(G_{\mathcal{E}}) \cup E$. Note that H

has degree $d+3$, as we only added at most one edge to vertices of degree at most $d+1$. Further note that by definition of S' we have that $S \subseteq \bigcup_{v \in S'} S_v$. Since every edge in E is incident to at most one vertex in $N_1^G(\bigcup_{v \in S'} S_v)$ it follows that if H is Hamiltonian then it fulfils the conditions from Definition 9.2.1. Therefore, if we prove that H has a Hamiltonian cycle then $G_{\mathcal{E}}$ must be locally Hamiltonian.

Claim 2. There is a set of $2d$ pairwise vertex disjoint simple paths $\{P_e^{\text{in}}, P_{\tilde{e}}^{\text{out}} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\}$ for every $v \in V(\vec{\mathcal{E}}) \setminus S'$ with the following properties.

- If $e \in E^-_{\vec{\mathcal{E}}}(v)$ then P_e^{in} is a path from a_{18}^e to a_{31}^e.
- If $e = (q_i^1, q_i^2)$ for some $i \in \{1, \dots, m\}$ then P_e^{out} is a path from a_1^e to a_{15}^e.
- If $e \in E^+_{\vec{\mathcal{E}}}(v) \setminus \{(q_i^1, q_i^2) \mid 1 \leq i \leq m\}$ then P_e^{out} is a path from a_1^e to a_{17}^e.
- The set $\{x \in V(G) \mid x$ is contained in P_e^{in} or P_e^{out} for some $e\}$ is equal to S_v.

Proof of Claim 2. This can be achieved by letting $P_e^{\text{in}} := (a_{18}^e, \dots, a_{31}^e)$ for $e \in E^-_{\vec{\mathcal{E}}}(v)$. Additionally, for every edge $e = (q_i^1, q_i^2)$ we let $P_e^{\text{out}} := (a_1^e, \dots, a_{14}^e, a_{17}^e, a_{16}^e, a_{15}^e)$ if $q_i^2 \neq n(q_i^1)$ and $P_e^{\text{out}} = (a_1^e, \dots, a_4^e, b_3^v, b_2^v, b_1^v, a_5^e, \dots, a_{12}^e, b_4^v, b_5^v, b_6^v, a_{13}^e, a_{14}^e, a_{17}^e, a_{16}^e, a_{15}^e)$ otherwise. Finally for $e \in E^+_{\vec{\mathcal{E}}}(v) \setminus \{(q_i^1, q_i^2) \mid 1 \leq i \leq m\}$ we set $P_e^{\text{out}} := (a_1^e, \dots, a_{17}^e)$ for $e = (v, w)$, $w \neq n(v)$ and $P_e^{\text{out}} := (a_1^e, \dots, a_4^e, b_3^v, b_2^v, b_1^v, a_5^e, \dots, a_{12}^e, b_4^v, b_5^v, b_6^v, a_{13}^e, \dots, a_{17}^e)$ for $e = (v, n(v))$. ■

For $v \in S'$ we define the sets $T_v^{\text{in}} := \{(q_i^{j-1}, q_i^j) \mid 1 \leq i \leq m, 2 \leq j \leq \ell_i, q_i^j = v\}$ and $T_v^{\text{out}} := \{(q_i^j, q_i^{j+1}) \mid 1 \leq i \leq m, 2 \leq j \leq \ell_i - 1, q_i^j = v\}$. Since for every $v \in S'$ there is exactly one path out of $Q_1, \dots, Q_m$ that ends in v, we get that $|T_v^{\text{in}}| - 1 = |T_v^{\text{out}}|$ and hence the preconditions for Lemma 9.2.11 are met. Therefore we obtain a set of paths $\{P_e^{\text{in}}, P_{\tilde{e}}^{\text{out}} \mid e \in E^-_{\vec{\mathcal{E}}}(v), \tilde{e} \in E^+_{\vec{\mathcal{E}}}(v)\}$ for every $v \in S'$ as in Lemma 9.2.11.

Since $S_v \cap S_w = \emptyset$ for every pair $v, w \in V(\vec{\mathcal{E}})$ with $v \neq w$, we now have a set of pairwise vertex disjoint simple paths $\{P_e^{\text{in}}, P_e^{\text{out}} \mid e \in E(\vec{\mathcal{E}})\}$ such that every vertex of H is contained in one of the paths. For every edge $e \in E(\vec{\mathcal{E}})$ we now concatenate P_e^{out} with P_e^{in} to a path P_e. This is possible as for every edge $e \in E(\vec{\mathcal{E}})$ the end vertex of P_e^{out} and the start vertex of P_e^{in} are adjacent. Finally we concatenate all paths P_e in the order given by the ordering $f : E(\vec{\mathcal{E}}) \to \{1, \dots, |E(\vec{\mathcal{E}})|\}$ used in the construction of $G_{\mathcal{E}}$. This gives us a cycle which contains every vertex in H precisely once. Hence H is Hamiltonian.

□

Theorem 9.2.12. *There are $d \in \mathbb{N}$ and constants $\delta := \delta(d), \epsilon := \epsilon(d) \in (0, 1)$ and a sequence of graphs $(G_N)_{N \in \mathbb{N}}$ of bounded degree d and increasing order such that G_N is δ-locally Hamiltonian and ϵ-far from being Hamiltonian for every $N \in \mathbb{N}$.*

Proof. Let $D \in \mathbb{N}$ and $(\mathcal{E}_N)_{N \in \mathbb{N}}$ a sequence of D-bounded degree expanders of increasing order. Such expanders exist and there are even some known explicit constructions (see for example [104] or [115]). Then for every $N \in \mathbb{N}$ we set $G_N := G_{\mathcal{E}_N}$ be the graph constructed in Definition 9.2.3. By Theorem 9.2.6 and Theorem 9.2.10 there is a degree bound d and constants $\delta, \epsilon \in (0,1)$, whose size only depends on D, such that G_N has degree bounded by d and G_N is δ-locally Hamiltonian and ϵ-far from being Hamiltonian. □

9.2.4 Deriving the lower bound

We now obtain the following result as a corollary of Theorem 9.2.12.

Corollary 9.2.13. Hamiltonicity is not testable with one-sided error and query complexity $o(n)$ in the bounded degree model.

Proof. Pick d as in Theorem 9.2.12 and let $\mathcal{P} \subseteq \mathcal{C}_d$ be the class of all Hamiltonian graphs of degree at most d. Towards a contradiction, assume that for every $\epsilon \in (0,1]$ and $n \in \mathbb{N}$ there is a one-sided error ϵ-tester for $\mathcal{P} \cap \{G \in \mathcal{C}_d \mid |V(G)| = n\}$ with query complexity $o(n)$. Let $\delta, \epsilon \in (0,1)$ be constants such that there is a sequence of d-bounded degree graphs $(G_N)_{N\in\mathbb{N}}$ of increasing order such that G_N is δ-locally Hamiltonian and ϵ-far from being Hamiltonian for every $N \in \mathbb{N}$. Note that δ and ϵ exist by Theorem 9.2.12. Let T be an ϵ-tester for $\mathcal{P}$ with query complexity $f(n) \in o(n)$. Since $f(n) \in o(n)$ there must be $n_0 \in \mathbb{N}$ such that $f(n) \leq \delta n$ for all $n \geq n_0$. Let $N \in \mathbb{N}$ such that $|V(G_N)| \geq n_0$. Since G_N is ϵ-far from $\mathcal{P}$ there must be a sequence of queries $(q_1, \dots, q_m)$ with $m \leq \delta n$ such that T queries the sequence $(q_1, \dots, q_m)$ with non-zero probability and rejects G_N with non-zero probability after performing the queries $(q_1, \dots, q_m)$. Let S be the set of vertices $v \in V(G_N)$ such that there is a query $q_i = (v, j)$ for $i \in [m]$. Because G_N is δ-locally Hamiltonian and $|S| \leq \delta n$ there is a graph $H \in \mathcal{P}$ on n vertices and $T \subseteq V(H)$ such that there is an isomorphism $G_N[N_1^{G_N}(S)]$ to $H[N_1^H(T)]$ which maps S to T. Hence, after renaming the vertices in $N_1^H(T)$, the tester T gets exactly the same answers for queries in $q_1, \dots, q_m$ for G_N and H. This implies that T queries the sequence $(q_1, \dots, q_m)$ in H with non-zero probability and hence must reject H with non-zero probability. This contradicts the assumption that T was a one-sided error tester for Hamiltonicity. □

Note 9.2.14. Note that the above argument is not sufficient for two-sided error testers, because a two-sided error tester would be allowed to reject H with probability less than $\frac{1}{3}$.

9.3 Lower bound for testing treewidth

In this section we show that treewidth is not constant query testable. Let us first recall the concept of treewidth. A *tree decomposition* of a graph G is a pair (T, B) where T is a tree and B is a function $B : V(T) \to \{X \mid X \subseteq V(G)\}$ with the following properties.

(TW1) For every $v \in V(G)$ there is $t \in V(T)$ such that $v \in B(t)$.

(TW2) For every $e \in E(G)$ there is $t \in V(T)$ such that $e \subseteq B(t)$.

(TW3) The graph $T[\{t \in V(T) \mid v \in B(t)\}]$ is connected for every $v \in V(G)$.

The *width* of a tree decomposition (T, B) of a graph G is the size of the largest bag reduced by 1, i. e. $\max_{t \in V(T)} |B(t)| - 1$. The *treewidth* of a graph G, denoted by $\mathrm{tw}(G)$ is the minimum width of any tree decomposition of G. For the rest of the section we let $d \in \mathbb{N}$ be a fixed degree bound. For any function $f : \mathbb{N} \to \mathbb{N}$ we define the property $\mathcal{P}^{\mathrm{tw}}_{\leq f} := \{G \in \mathcal{C}_d \mid \mathrm{tw}(G) \leq f(|V(G)|)\}$ on the class $\mathcal{C}_d$ of graphs of bounded degree d. We show the following theorem.

Theorem 9.3.1. *The property $\mathcal{P}^{\mathrm{tw}}_{\leq f}$ is not constant query testable for every $f \in o(n) \cap \omega(1)$.*

To prove this we use a theorem from Grohe and Marx showing that (vertex) expanders have treewidth linear in the amount of vertices [80]. This enables us to argue similarly as in the proof of Theorem 6.2.1 using both Theorem 5.2.2 from [102] and Theorem 5.1.2 from [2].

We first show that edge expanders have linear treewidth which is a straight forward consequence of [80].

Corollary 9.3.2 (Edge expansion version of Theorem 3 in [80])**.** Let $d \in \mathbb{N}$ and $\mathcal{E}$ be a class of expanders of bounded degree d. There is a constant $\beta > 0$ such that $\mathrm{tw}(G) \geq \beta \cdot |V(G)|$ for every $G \in \mathcal{E}$.

Proof. Since $\mathcal{E}$ is a class of edge expanders we know that there is a constant $\epsilon > 0$ such that the edge expansion $h(G) > \epsilon$ for every $G \in \mathcal{E}$, where

$$h(G) := \min_{\substack{\{S \subseteq V(G) \mid \\ 0 \leq |S| \leq |V(G)|/2\}}} \frac{|\langle S, \overline{S} \rangle_G|}{|S|}.$$

Additionally, since every $G \in \mathcal{E}$ has bounded degree d, every vertex in the open neighbourhood $N_1^G(S) \setminus S$ of any set $S \subseteq V(G)$ can contribute at most d edges to the set of S-crossing edges $\langle S, \overline{S} \rangle_G$. Hence we get for the vertex expansion of any graph $G \in \mathcal{E}$

$$\mathrm{vx}(G) := \min_{\substack{\{S \subseteq V(G) \mid \\ 0 \leq |S| \leq |V(G)|/2\}}} \frac{|N_1^G(S) \setminus S|}{|S|} \geq \frac{h(G)}{d} > \frac{\epsilon}{d} =: \epsilon' > 0.$$

Now [80, Theorem 3] implies that there is a constant $\beta > 0$ such that $\mathrm{tw}(G) \geq \beta \cdot |V(G)|$ for every $G \in \mathcal{E}$. $\square$

In the following lemma we argue that reducing the treewidth of a class of expanders to some sublinear function takes at least a linear amount of edge deletions.

Lemma 9.3.3. *Let $\mathcal{E}$ be a class of expanders and $\beta > 0$ the constant from 9.3.2. For every $f \in o(n)$ and every $\epsilon \in (0, \frac{\beta}{2d})$ there is a constant $n_0 \in \mathbb{N}$ such that every $G \in \mathcal{E}$ with $|V(G)| \geq n_0$ is ϵ-far from $\mathcal{P}^{\mathrm{tw}}_{\leq f}$.*

Proof. Let $f \in o(n)$ and $\epsilon \in (0, \frac{\beta}{2d})$ be arbitrary. Since $f \in o(n)$ there is a constant $n_0 \in \mathbb{N}$ such that

$$f(n) < \frac{\beta}{2} \cdot n \tag{9.5}$$

for every $n \geq n_0$.

Assume that the assertion is not true. Hence there must be a graph $G \in \mathcal{E}$ with $|V(G)| \geq n_0$ which is ϵ-close to $\mathcal{P}^{\text{tw}}_{\leq f}$. Hence there is a graph G' with $\text{tw}(G') \leq f(|V(G')|)$ which can be obtained from G by adding/deleting no more than ϵdn edges. Let (T', B') be a tree decomposition of G' with $|B'(t)| \leq f(|V(G')|)$ for every $t \in V(T')$. We now pick S to be a minimal set containing at least one vertex of every edge in $E(G) \setminus E(G')$. Then (T, B) where $T := T'$ and $B(t) := B'(t) \cup S$ for every $t \in T'$ is a tree decomposition of G. Furthermore, for every $t \in T$ we have

$$|B(t)| = |B'(t)| + |S| \leq f(|V(G)|) + \epsilon dn < \beta n$$

where the last inequality follows by choice of ϵ and n_0. This contradicts that $\text{tw}(G) \geq \beta n$. □

We can now prove Theorem 9.3.1.

Proof of Theorem 9.3.1. Let $(G_i)_{i \in \mathbb{N}}$ be a family of expanders of bounded degree d and $\mathcal{E} := \{G_i \mid i \in \mathbb{N}\}$. Let $\beta > 0$ be the constant from Corollary 9.3.2 and $\tilde{n}_0$ be the constant from Lemma 9.3.3.

To prove that $\mathcal{P}^{\text{tw}}_{\leq f}$ is not testable we prove $\mathcal{P}^{\text{tw}}_{\leq f}$ is not repairable and then apply Theorem 5.1.2. Let $\epsilon \in (0, \frac{\beta}{2d})$ and let $r \in \mathbb{N}$, $\lambda > 0$ and $n_0 \in \mathbb{N}$ be arbitrary. We set $\lambda' := \frac{\lambda}{(t \cdot 2^{r+1})}$, where t is the number of r-types of bounded degree d, and let n_0' be the positive integer from Theorem 5.2.2 corresponding to λ'. Let $i \in \mathbb{N}$ be any fixed index such that for $G := G_i$ we have $n := |V(G)| \geq n_0$, $f(n) \geq n_0'$, $n \geq \frac{4n_0'}{\lambda}$ and $n \geq \tilde{n}_0$. Choosing i in such a way is always possible as $(G_i)_{i \in \mathbb{N}}$ is a family of expanders, implying that there are arbitrarily large expanders in $\mathcal{E}$ and $f \in \omega(1)$. By Theorem 5.2.2 there is a graph H with $m := |V(H)| \leq n_0'$ such that the sampling distance $\delta_{\odot}(G, H)$ (see Definition 5.2.1) is less than or equal to λ'. Let G' be the graph consisting of $\lfloor \frac{n}{m} \rfloor$ copies of H and $n \bmod m$ isolated vertices. Note that we picked G' such that $|V(G)| = |V(G')|$. Furthermore, the treewidth of H is less than m and therefore

$$\text{tw}(G') \leq m \leq n_0' \leq f(n) = f(|V(G')|).$$

Hence $G' \in \mathcal{P}^{\text{tw}}_{\leq f}$. On the other hand, we know that G is ϵ-far from $\mathcal{P}^{\text{tw}}_{\leq f}$ by Lemma 9.3.3 as $|V(G)| \geq \tilde{n}_0$.

Let $\tau_1, \ldots, \tau_t$ be a list of all r-types and $T_r := \{\tau_1, \ldots, \tau_t\}$. Similarly to the argument in the proof of Theorem 6.2.1 we have

$$
\begin{aligned}
\|\mathrm{freq}_r(G) - \mathrm{freq}_r(G')\|_1 &= \sum_{i=1}^{t} |\rho_{G,r}(\{\tau_i\}) - \rho_{G',r}(\{\tau_i\})| \\
&= \sum_{i=1}^{t} \left|\rho_{G,r}(\{\tau_i\}) - \frac{n \bmod m}{n} \cdot \rho_{K_1,r}(\{\tau_i\}) - \left\lfloor \frac{n}{m} \right\rfloor \cdot \frac{m}{n} \cdot \rho_{H,r}(\{\tau_i\})\right| \\
&\leq \sum_{i=1}^{t} \left|\rho_{G,r}(\{\tau_i\}) - \rho_{H,r}(\{\tau_i\})\right| + \sum_{i=1}^{t} \left|\frac{n \bmod m}{n} \cdot \rho_{K_1,r}(\{\tau_i\})\right| \\
&\qquad + \sum_{i=1}^{t} \left|\rho_{H,r}(\{\tau_i\}) - \left\lfloor \frac{n}{m} \right\rfloor \cdot \frac{m}{n} \cdot \rho_{H,r}(\{\tau_i\})\right| \\
&\leq \sum_{i=1}^{t} \left|\rho_{G,r}(\{\tau_i\}) - \rho_{H,r}(\{\tau_i\})\right| + \frac{2m}{n} \\
&\leq t \cdot \sup_{X \subseteq T_r} |\rho_{G,r}(X) - \rho_{H,r}(X)| + \frac{2m}{n} \\
&\leq t \cdot 2^r \cdot \delta_{\odot}(G, H) + \frac{2m}{n} \\
&\leq \frac{\lambda}{2} + \frac{\lambda}{2} \\
&= \lambda.
\end{aligned}
$$

This proves that $\mathcal{P}^{\mathrm{tw}}_{\leq f}$ is not repairable. □

9.4 Summary

In this section we studied property testing for some NP-hard problems. Utilising a standard approach of showing hardness for property testing, we use a local reduction to show that testing dominating set size is NP-hard, where the local reduction is derived from a standard polynomial time reduction. Motivated by understanding non-testable properties we constructed a sequence of hard instances for Hamiltonicity, a problem which is known to have no tester of query complexity $o(n)$ [70, 129]. At the core of the construction we use expanders to guarantee local Hamiltonicity, while farness from Hamiltonicity is enforced by using gadgets to encode a condition which can not be satisfied for too many vertices at once. Finally, we show that treewidth is not testable which is a consequence of expanders having linear treewidth. This result is partial in the sense that we currently do not know what the query complexity of testing treewidth is. We elaborate on this open question in the Conclusions.

Chapter 10

Conclusions

In this thesis we have tackled the question of whether FO definable properties are testable in the bounded degree model, raised by Adler and Harwath in [2]. We take a first step towards a full logical characterisation of testable graph properties in the bounded degree model by showing that all properties defined by a sentence in the prefix class Σ_2 are testable while there exist a sentence in the prefix class Π_2 which is not testable. This yields the first study of testability of properties defined in FO in the bounded degree model without restrictions and provides a missing equivalent to a study of FO definable properties in the dense model by Alon et al. [6]. Our negative result is furthermore interesting from a model theoretic point of view providing insights into the expressive power of FO. Specifically, it proves the existence of a class of bounded degree expanders which is definable in FO. We further prove testability for properties defined by specific classes of FO definable properties expressing that the frequency vector capturing which neighbourhood types appear in a graph are of a particular form.

Utilising the FO counter example we answer an open question asked by Goldreich in the context of characterising properties testable by a one-sided error POT in [76]. More precisely, we showed that there exists a GSF-local property which is not testable. Since every GSF-local property which is non-propagating admits a POT (see [76]) and hence is testable, our result specifically implies that there is a property which is GSF-local and non-propagating. This is a valuable realisation concerning the search for a characterisation of testable properties in the bounded degree model. Ito et al. suggest in [89] that properties which are close to being GSF-local might characterise testability, which we disproved.

In the later part of this thesis we have drawn inspiration from classical complexity theory and studied problems which are NP-hard. We gave a local reduction to prove that testing dominating set size is not possible with $o(n)$ queries, which was unknown to our knowledge. We further gave a construction of hard instances for testing Hamiltonicity with one-sided error with the goal of understanding structural connections to hardness for property testing. We have further given a partial result into the direction of determining the complexity of testing treewidth in the bounded degree model.

10.1 Future Work

One of the major open questions in the field of graph property testing is the question of how to characterise testable properties in the bounded degree model. The crucial piece of understanding we are missing here, is the question of when GSF-local properties are propagating. Understanding this connection is therefore a very interesting goal. The following more approachable questions of interest are left open in this thesis.

- Can we find other, possibly smaller or simpler, counter examples, i. e. properties that are GSF-local and are propagating?
- Are there any other applications of the class of expanders which is FO definable and GSF-local?
- Can sentences in Σ_2 be tested uniformly?
- Can neighbourhood regularity be tested in general?
- Can we characterise which FO-definable properties are testable?
- Can we characterise which FO definable properties are GSF-local?

Regarding the last question we would like to remark that example 8.2.8 shows that it is not true that FO definable properties are GSF-local if and only if they are definable by 0-profiles. Hence the connection between FO definability and GSF-locality is complicated. An answer to this question would yield a characterisation of FO properties having a one-sided error POT dependent on the non-propagating condition building on the characterisation from [76].

An effort worth while would also be to put some effort into gearing results towards application. We could consider and try to improve the dependency of the query complexity on ϵ and d and the running times for the testers developed in this thesis.

We think that the question of the relationship between classical polynomial time reductions and local reductions yields interesting research questions. In general, it is not true that NP-hard problems are not testable as there are testable problems which are NP-hard. On the other hand, when the reduction uses gadgets, which interact depending on single edges, polynomial time reductions are local. Hence, it is interesting to study problems like treewidth, for which the classical reduction is not of this form. Furthermore, we are interested in the question of whether linear time reductions yields local reductions.

A further research direction of interest is to extend results from the bounded degree model to more general sparse graph classes. There is a rich variety of structure of sparse graph classes being exploited for algorithmic result and it would be interesting to explore what can be achieved in the property testing setting. For example, in [100] the authors show sublinear testability of Hamiltonicity in the general model for minor-free classes of graphs.

Bibliography

[1] I. Adler and P. Fahey. Faster property testers in a variation of the bounded degree model. In *40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2020, December 14-18, 2020, BITS Pilani, K K Birla Goa Campus, Goa, India (Virtual Conference)*, pages 7:1–7:15, 2020.

[2] I. Adler and F. Harwath. Property testing for bounded degree databases. In *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France*, pages 6:1–6:14, 2018.

[3] A. V. Aho and J. D. Ullman. Universality of data retrieval languages. In *Proceedings of the 6th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, pages 110–119, 1979.

[4] N. Alon. Testing subgraphs in large graphs. *Random Structures & Algorithms*, 21(3–4):359–370, 2002.

[5] N. Alon, E. Fischer, M. Krivelevich, and M. Szegedy. Efficient testing of large graphs. In *FOCS '99 Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, pages 656–666, 1999.

[6] N. Alon, E. Fischer, M. Krivelevich, and M. Szegedy. Efficient testing of large graphs. *Combinatorica*, 20(4):451–476, 2000. Preliminary version in FOCS'99.

[7] N. Alon, E. Fischer, I. Newman, and A. Shapira. A combinatorial characterization of the testable graph properties: It's all about regularity. *SIAM Journal on Computing*, 39(1):143–167, 2009.

[8] N. Alon, T. Kaufman, M. Krivelevich, and D. Ron. Testing triangle-freeness in general graphs. *SIAM Journal on Discrete Mathematics*, 22(2):786–819, 2008.

[9] N. Alon, M. Krivelevich, I. Newman, and M. Szegedy. Regular languages are testable with a constant number of queries. *SIAM Journal on Computing*, 30(6):1842–1862, 2000.

[10] N. Alon and V. D. Milman. $\lambda 1$, isoperimetric inequalities for graphs, and superconcentrators. *Journal of Combinatorial Theory, Series B*, 38(1):73–88, 1985.

[11] N. Alon and A. Shapira. A characterization of easily testable induced subgraphs. *Combinatorics Probability and Computing*, 15(6):791–806, 2006.

[12] N. Alon and A. Shapira. A characterization of the (natural) graph properties testable with one-sided error. *SIAM Journal on Computing*, 37(6):1703–1727, 2008.

[13] N. Alon and J. Spencer. *The Probabilistic Method*. John Wiley, New York, 2 edition, 2000.

[14] S. Arnborg, D. G. Corneil, and A. Proskurowski. Complexity of finding embeddings in ak-tree. *SIAM Journal on Algebraic Discrete Methods*, 8(2):277–284, 1987.

[15] G. Bagan. Mso queries on tree decomposable structures are computable with linear delay. In *International Workshop on Computer Science Logic*, pages 167–181. Springer, 2006.

[16] A. Belovs, E. Blais, and A. Bommireddi. Testing convexity of functions over finite domains. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2030–2045. SIAM, 2020.

[17] M. A. Bender and D. Ron. Testing properties of directed graphs: Acyclicity and connectivity. *Random Structures & Algorithms*, 20(2):184–205, 2002.

[18] I. Benjamini, O. Schramm, and A. Shapira. Every minor-closed property of sparse graphs is testable. In *Advances in Mathematics*, volume 223, pages 2200–2218. 2010.

[19] S. van Bergerem. Learning concepts definable in first-order logic with counting. In *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13. IEEE, 2019.

[20] S. van Bergerem, M. Grohe, and M. Ritzert. On the parameterized complexity of learning logic. *arXiv preprint arXiv:2102.12201*, 2021.

[21] E. Blais, J. Brody, and K. Matulef. Property testing lower bounds via communication complexity. *computational complexity*, 21(2):311–358, 2012.

[22] A. Bogdanov, K. Obata, and L. Trevisan. A lower bound for testing 3-colorability in bounded-degree graphs. In *43rd Symposium on Foundations of Computer Science (FOCS 2002), 16-19 November 2002, Vancouver, BC, Canada, Proceedings*, pages 93–102, 2002.

[23] B. Bollig and D. Kuske. An optimal construction of Hanf sentences. *Journal of Applied Logic*, 10(2):179–186, 2012.

[24] J. R. Büchi. Weak second-order arithmetic and finite automata. *Mathematical Logic Quarterly*, 6(1-6):66–92, 1960.

[25] C. L. Canonne. A survey on distribution testing: Your data is big. But is it blue? *Theory of Computing*, pages 1–100, 2020.

[26] X. Chen, R. A. Servedio, L.-Y. Tan, E. Waingarten, and J. Xie. Settling the query complexity of non-adaptive junta testing. *Journal of the ACM (JACM)*, 65(6):1–18, 2018.

[27] S. A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, STOC '71, page 151–158, New York, NY, USA, 1971. Association for Computing Machinery.

[28] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to algorithms*. MIT press, 2009.

[29] B. Courcelle. The monadic second-order logic of graphs. I. recognizable sets of finite graphs. *Information and computation*, 85(1):12–75, 1990.

[30] B. Courcelle, J. A. Makowsky, and U. Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33(2):125–150, 2000.

[31] B. Courcelle, J. A. Makowsky, and U. Rotics. On the fixed parameter complexity of graph enumeration problems definable in monadic second-order logic. *Discrete applied mathematics*, 108(1-2):23–52, 2001.

[32] A. Czumaj, H. Fichtenberger, P. Peng, and C. Sohler. Testable properties in general graphs and random order streaming. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2020, August 17-19, 2020, Virtual Conference*, pages 16:1–16:20, 2020.

[33] A. Czumaj, P. Peng, and C. Sohler. Testing cluster structure of graphs. In *Proceedings of the forty-seventh annual ACM symposium on Theory of Computing*, pages 723–732, 2015.

[34] A. Czumaj, P. Peng, and C. Sohler. Relating two property testing models for bounded degree directed graphs. In *STOC '16 Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 1033–1045, 2016.

[35] A. Czumaj, A. Shapira, and C. Sohler. Testing hereditary properties of nonexpanding bounded-degree graphs. *SIAM Journal on Computing*, 38(6):2499–2510, 2009.

[36] A. Czumaj and C. Sohler. Testing expansion in bounded-degree graphs. *Combinatorics, Probability and Computing*, 19(5-6):693–709, 2010.

[37] A. Czumaj and C. Sohler. A characterization of graph properties testable for general planar graphs with one-sided error (it's all about forbidden subgraphs). In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1525–1548. IEEE, 2019.

[38] A. Dawar, M. Grohe, and S. Kreutzer. Locally excluding a minor. In *22nd Annual IEEE Symposium on Logic in Computer Science (LICS 2007)*, pages 270–279. IEEE, 2007.

[39] A. Dawar, M. Grohe, S. Kreutzer, and N. Schweikardt. Approximation schemes for first-order definable optimisation problems. In *21st Annual IEEE Symposium on Logic in Computer Science (LICS'06)*, pages 411–420. IEEE, 2006.

[40] A. Dawar, M. Grohe, S. Kreutzer, and N. Schweikardt. Model theory makes formulas large. In *International Colloquium on Automata, Languages, and Programming*, pages 913–924. Springer, 2007.

[41] A. Dawar and S. Kreutzer. Parameterized complexity of first-order logic. In *Electronic Colloquium on Computational Complexity, TR09-131*, page 39, 2009.

[42] R. Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.

[43] J. Dodziuk. Difference equations, isoperimetric inequality and transience of certain random walks. *Transactions of the American Mathematical Society*, 284(2):787–794, 1984.

[44] J. Doner. Tree acceptors and some of their applications. *Journal of Computer and System Sciences*, 4(5):406–451, 1970.

[45] R. G. Downey, M. R. Fellows, and U. Taylor. The parameterized complexity of relational database queries and an improved characterization of W[1]. *DMTCS*, 96:194–213, 1996.

[46] A. Durand and E. Grandjean. First-order queries on structures of bounded degree are computable with constant delay. *ACM Transactions on Computational Logic (TOCL)*, 8(4):21–es, 2007.

[47] A. Durand, N. Schweikardt, and L. Segoufin. Enumerating answers to first-order queries over databases of low degree. In *Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 121–131, 2014.

[48] Z. Dvořák. Baker game and polynomial-time approximation schemes. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2227–2240. SIAM, 2020.

[49] Z. Dvořák. Approximation metatheorem for fractionally treewidth-fragile graphs. *arXiv preprint arXiv:2103.08698*, 2021.

[50] Z. Dvořák, D. Král, and R. Thomas. Deciding first-order properties for sparse graphs. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 133–142. IEEE, 2010.

[51] H. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer-Verlag, Berlin Heidelberg New York, 2 edition, 1999.

[52] H. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic*. Springer-Verlag, New York, 2 edition, 1994.

[53] T. Eden and W. Rosenbaum. Lower bounds for approximating graph parameters via communication complexity. *arXiv preprint arXiv:1709.04262*, 2017.

[54] R. Fagin, L. J. Stockmeyer, and M. Y. Vardi. On monadic NP vs monadic co-NP. *Information and Computation*, 120(1):78–92, 1995.

[55] H. Fichtenberger, P. Peng, and C. Sohler. On constant-size graphs that preserve the local structure of high-girth graphs. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2015, August 24-26, 2015, Princeton, NJ, USA*, pages 786–799, 2015.

[56] H. Fichtenberger, P. Peng, and C. Sohler. Every testable (infinite) property of bounded-degree graphs contains an infinite hyperfinite subproperty. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 714–726. Society for Industrial and Applied Mathematics, 2019.

[57] E. Fischer. The art of uninformed decisions: A primer to property testing. *Science*, 75:97–126, 2001.

[58] E. Fischer and A. Matsliah. Testing graph isomorphism. *SIAM Journal on Computing*, 38(1):207–225, 2008.

[59] J. Flum, M. Frick, and M. Grohe. Query evaluation via tree-decompositions. *Journal of the ACM (JACM)*, 49(6):716–752, 2002.

[60] J. Flum and M. Grohe. Fixed-parameter tractability, definability, and model-checking. *SIAM Journal on Computing*, 31(1):113–145, 2001.

[61] F. V. Fomin, P. A. Golovach, and D. M. Thilikos. On the parameterized complexity of graph modification to first-order logic properties. *Theory of Computing Systems*, 64(2):251–271, 2020.

[62] M. Frick and M. Grohe. Deciding first-order properties of locally tree-decomposable structures. *Journal of the ACM (JACM)*, 48(6):1184–1206, 2001.

[63] H. Gaifman. On local and non-local properties. In *Proceedings of the Herbrand Symposium, Logic Colloquium '81*, pages 105–135. North-Holland, 1982.

[64] R. Ganian, P. Hliněný, J. Obdržálek, J. Schwartz, J. Teska, et al. Fo model checking of interval graphs. In *International Colloquium on Automata, Languages, and Programming*, pages 250–262. Springer, 2013.

[65] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.

[66] M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified NP-complete problems. In *Proceedings of the sixth annual ACM symposium on Theory of computing*, pages 47–63, 1974.

[67] M. R. Garey, D. S. Johnson, and R. E. Tarjan. The planar hamiltonian circuit problem is NP-complete. *SIAM J. Comput.*, 5(4):704–714, 1976.

[68] F. Gavril. Some NP-complete problems on graphs. Technical report, Computer Science Department, Technion, 2011.

[69] O. Goldreich. *Introduction to Property Testing*. Cambridge University Press, New York, 1 edition, 2017.

[70] O. Goldreich. On testing hamiltonicity in the bounded degree graph model. *Electronic Colloquium on Computational Complexity (ECCC)*, (18), 2020.

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

[72] O. Goldreich, M. Krivelevich, I. Newman, and E. Rozenberg. Hierarchy theorems for property testing. *computational complexity*, 21(1):129–192, 2012.

[73] O. Goldreich and D. Ron. Property testing in bounded degree graphs. In *STOC '97 Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 406–415, 1997.

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

[75] O. Goldreich and D. Ron. On testing expansion in bounded-degree graphs. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, pages 68–75. Springer, 2011.

[76] O. Goldreich and D. Ron. On proximity-oblivious testing. *SIAM Journal on Computing*, 40(2):534–566, 2011. Preliminary version appeared at *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC 2009).*

[77] O. Goldreich and I. Shinkar. Two-sided error proximity oblivious testing. *Random Structures & Algorithms*, 48(2):341–383, 2016.

[78] O. Goldreich and L. Trevisan. Three theorems regarding testing graph properties. *Random Structures & Algorithms*, 23(1):23 – 57, 2003.

[79] M. Grohe, S. Kreutzer, and S. Siebertz. Deciding first-order properties of nowhere dense graphs. *Journal of the ACM (JACM)*, 64(3):1–32, 2017.

[80] M. Grohe and D. Marx. On tree width, bramble size, and expansion. *J. Comb. Theory, Ser. B*, 99(1):218–228, 2009.

[81] M. Grohe and M. Ritzert. Learning first-order definable concepts over structures of small degree. *arXiv:1701.05487*, 2017.

[82] M. Grohe and G. Turán. Learnability and definability in trees and similar structures. *Theory of Computing Systems*, 37(1):193–220, 2004.

[83] W. Hanf. Model-theoretic methods in the study of elementary logic. In J. W. Addison, L. Henkin, and A. Tarski, editors, *The Theory of Models*, pages 132–145. North Holland, 1965.

[84] A. Hassidim, J. A. Kelner, H. N. Nguyen, and K. Onak. Local graph partitions for approximation and testing. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 22–31. IEEE, 2009.

[85] L. Heimberg, D. Kuske, and N. Schweikardt. An optimal Gaifman normal form construction for structures of bounded degree. In *2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 63–72. IEEE, 2013.

[86] F. Hellweg and C. Sohler. Property testing in sparse directed graphs: Strong connectivity and subgraph-freeness. In L. Epstein and P. Ferragina, editors, *Algorithms – ESA 2012*, volume 7501 of *Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, 2012.

[87] S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. *BULL. AMER. MATH. SOC.*, 43(4):439–561, 2006.

[88] N. Immerman. Languages that capture complexity classes. *SIAM Journal on Computing*, 16(4):760–778, 1987.

[89] H. Ito, A. Khoury, and I. Newman. On the characterization of 1-sided error strongly-testable graph properties for bounded-degree graphs. *(to appear) Journal of Computational Complexity. arXiv:1909.09984*, 2019.

[90] K. Iwama and Y. Yoshida. Parameterized testability. *ACM Trans. Comput. Theory*, 9(4):16:1–16:16, 2018.

[91] S. Kale and C. Seshadhri. An expansion tester for bounded degree graphs. *SIAM Journal on Computing*, 40(3):709–720, 2011.

[92] R. M. Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.

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

[94] K. Kawarabayashi and Y. Yoshida. Testing subdivision-freeness: property testing meets structural graph theory. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 437–446, 2013.

[95] W. Kazana and L. Segoufin. First-order query evaluation on structures of bounded degree. *arXiv preprint arXiv:1105.3583*, 2011.

[96] W. Kazana and L. Segoufin. Enumeration of first-order queries on classes of structures with bounded expansion. In *Proceedings of the 32nd ACM SIGMOD-SIGACT-SIGAI symposium on Principles of database systems*, pages 297–308, 2013.

[97] A. Kumar, C. Seshadhri, and A. Stolman. Random walks and forbidden minors II: a poly($d\varepsilon^{-1}$)-query tester for minor-closed properties of bounded degree graphs. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 559–567, 2019.

[98] A. Kumar, C. Seshadhri, and A. Stolman. Random walks and forbidden minors III: poly($d\varepsilon^{-1}$)-time partition oracles for minor-free graph classes. *arXiv preprint arXiv:2102.00556*, 2021.

[99] R. Levi and D. Ron. A quasi-polynomial time partition oracle for graphs with an excluded minor. *ACM Transactions on Algorithms (TALG)*, 11(3):1–13, 2015.

[100] R. Levi and N. Shoshan. Testing hamiltonicity (and other problems) in minor-free graphs. *arXiv preprint arXiv:2102.11728*, 2021.

[101] L. Libkin. *Elements of Finite Model Theory.* Texts in Theoretical Computer Science. An EATCS Series. Springer, 2004.

[102] L. Lovász. *Large Networks and Graph Limits*, volume 60 of *Colloquium Publications.* American Mathematical Society, 2012.

[103] F. Magniez and M. de Rougemont. Property testing of regular tree languages. *Algorithmica*, 49(2):127–146, 2007.

[104] G. A. Margulis. Explicit constructions of expanders. *Problemy Peredači Informacii*, 9(4):71–80, 1973.

[105] A. McGregor. Graph stream algorithms: a survey. *ACM SIGMOD Record*, 43(1):9–20, 2014.

[106] M. Monemizadeh, S. Muthukrishnan, P. Peng, and C. Sohler. Testable bounded degree graph properties are random order streamable. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, pages 131:1–131:14, 2017.

[107] B. Monien and I. H. Sudborough. Min cut is NP-complete for edge weighted trees. *Theoretical Computer Science*, 58(1-3):209–229, 1988.

[108] S. Muthukrishnan. *Data streams: Algorithms and applications.* Now Publishers Inc, 2005.

[109] A. Nachmias and A. Shapira. Testing the expansion of a graph. *Information and Computation*, 208(4):309–314, 2010.

[110] J. Nešetřil and P. Ossona de Mendez. On nowhere dense graphs. *European Journal of Combinatorics*, 32(4):600–617, 2011.

[111] J. Nešetřil and P. Ossona de Mendez. *Sparsity: graphs, structures, and algorithms*, volume 28. Springer Science & Business Media, 2012.

[112] I. Newman and C. Sohler. Every property of hyperfinite graphs is testable. *SIAM Journal on Computing*, 42(3):1095–1112, 2013.

[113] K. Oono and Y. Yoshida. Testing properties of functions on finite groups. *Random Structures & Algorithms*, 49(3):579–598, 2016.

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

[115] O. Reingold, S. Vadhan, and A. Wigderson. Entropy waves, the zig-zag graph product, and new constant-degree expanders. *Annals of mathematics*, pages 157–187, 2002.

[116] D. Ron. Algorithmic and analysis techniques in property testing. *Foundations and Trends in Theoretical Computer Science*, 5(2):73–205, 2010.

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

[118] N. Schweikardt, L. Segoufin, and A. Vigny. Enumeration for fo queries over nowhere dense graphs. In *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 151–163, 2018.

[119] D. Seese. Linear time computable problems and first-order descriptions. *Mathematical Structures in Computer Science*, 6(6):505–526, 1996.

[120] L. J. Stockmeyer. *The complexity of decision problems in automata theory and logic.* PhD thesis, Massachusetts Institute of Technology, 1974.

[121] E. Szemerédi. Regular partitions of graphs. In *Proc. Colloq. Inter. CNRS*, 1978.

[122] J. W. Thatcher and J. B. Wright. Generalized finite automata theory with an application to a decision problem of second-order logic. *Mathematical systems theory*, 2(1):57–81, 1968.

[123] A. M. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London mathematical society*, 2(1):230–265, 1937.

[124] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[125] M. Y. Vardi. The complexity of relational query languages. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 137–146, 1982.

[126] A. C. Yao. Lower bounds by probabilistic arguments. In *24th Annual Symposium on Foundations of Computer Science (sfcs 1983)*, pages 420–428. IEEE, 1983.

[127] Y. Yoshida. Lower bounds on query complexity for testing bounded-degree csps. In *2011 IEEE 26th Annual Conference on Computational Complexity*, pages 34–44. IEEE, 2011.

[128] Y. Yoshida. Optimal constant-time approximation algorithms and (unconditional) inapproximability results for every bounded-degree CSP. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 665–674, 2011.

[129] Y. Yoshida and H. Ito. Query-number preserving reductions and linear lower bounds for testing. *IEICE Trans. Inf. Syst.*, 93-D(2):233–240, 2010.

[130] Y. Yoshida and H. Ito. Testing k-edge-connectivity of digraphs. *Journal of systems science and complexity*, 23(1):91–101, 2010.

[131] Y. Yoshida and H. Ito. Testing outerplanarity of bounded degree graphs. *Algorithmica*, 73(1):1–20, 2015.