Bachelor's Thesis

FO Model Checking on Some Dense Graph Classes Using FO Interpretations

Dimitri Rusin

First Examiner: Prof. Dr. Martin Grohe
Second Examiner: Prof. Dr. Erich Grädel
Supervisor: Patrick Landwehr

Rheinisch-Westfälische Technische Hochschule Aachen
Lehrstuhl für Informatik 7
Logik und Theorie diskreter Systeme

September 11th, 2017

ABSTRACT

FO interpretations can be used to create new classes of structures from old ones. But they are not always constructive, in the sense that there is not always a polynomial time algorithm that can, given a new structure, find an old structure it was created from. Still Gajarský et al. have proved that every such "creation" from graphs of bounded degree admits an fpt algorithm for FO Model Checking. I have extended this fpt algorithm to work for vertex- and edge-coloured directed graphs.

ZUSAMMENFASSUNG

FO-Interpretationen können verwendet werden, um aus bekannten Klassen von Strukturen neue Klassen von Strukturen zu erzeugen. Doch sie sind nicht immer konstruktiv, in dem Sinne, dass nicht immer ein Polynomialzeitalgorithmus existiert, der, gegeben eine neue Struktur, eine alte Struktur finden kann, aus der diese erzeugt wurde. Trotzdem haben Gajarský et al. gezeigt, dass solche "Erzeugnisse" von Graphen beschränkten Grades stets einen FPT-Algorithmus für FO Model Checking zulassen. Ich habe diesen FPT-Algorithmus auf knoten- und kantengefärbte gerichtete Graphen erweitert.

CONTENTS

INTRODUCTION

In [1], Gajarský, Hliněný, Obdržálek, Lokshtanov and Ramanujan have initiated the research led by the question:

> Which graph classes are FO interpretable in nowhere dense graph classes? Furthermore, which graph classes are efficiently FO interpretable in nowhere dense graph classes?

They themselves have characterized all graph classes FO interpretable in graph classes of bounded degree and have shown that all of them are also efficiently FO interpretable in some (other) graph class of bounded degree. For the characterization they have defined a new notion of *near-uniformity* of graph classes. So more precisely, their main results in [1] are as follows:

1. Every graph class $\mathcal{D}$, such that there is an FO interpretation Γ from $\{E\}$ to $\{E\}$ and a graph class $\mathcal{C}$ of bounded degree with $\mathcal{D} \subseteq \Gamma(\mathcal{C})$, is near-uniform.
2. Every near-uniform graph class $\mathcal{D}$ is efficiently FO interpretable in some class $\mathcal{C}$ of coloured graphs of bounded degree.

In this Bachelor's Thesis, I will extend their result from graph classes to classes of vertex- and edge-coloured directed graphs. More precisely, my main results are:

- Every class $\mathcal{D}$ of vertex- and edge-coloured directed graphs, such that there is an FO interpretation Γ from τ to σ and some τ-class $\mathcal{C}$ of bounded degree with $\mathcal{D} \subseteq \Gamma(\mathcal{C})$, is near-uniform.
- Every near-uniform class $\mathcal{D}$ of vertex- and edge-coloured directed graphs is efficiently FO interpretable in some class $\mathcal{C}$ of bounded degree.

Note that in the first result, $\mathcal{C}$ does not have to be a graph class anymore, but can be any class of structures. In the core, I use the same methods as Gajarský et al. have used.
The main benefit of such a result is that it yields new classes of structures on which FO Model Checking is fixed-parameter tractable, that is, efficiently solvable (or at least more efficiently than on the class of all structures). So this produces new meta-theorems, that is, results of the form: Deciding properties of structures definable in a *certain logic* is *tractable* on *certain classes of structures*. The two results from above combined can be formulated as the following meta-theorem: Deciding properties of structures definable in FO is *fixed-parameter tractable* on *classes of vertex- and edge-coloured directed graphs* FO *interpretable in some class of bounded degree*. Grohe and Kreutzer discuss methods for proving such meta-theorems in [2] and the methods we use here can be seen as a combination of their Method 5 (Reduction Method) and Method 7 (Locality based arguments), where it must be noted that Gajarský et al. have already used these methods to show their results.
My Bachelor's Thesis is structured as follows:

- In Section 2, I introduce the reader to my notation and definitions which are mostly but not always conventional.
- In Section 3, I will introduce my main tools, namely FO interpretations and FO reductions.

- In Section 4, I will show that there are "bad" FO interpretations, in the sense that they might not immediately allow a polynomial time reduction, which shows that FO interpretability does not imply FO reducibility.
- In Section 5, I will motivate the notion of near-uniformity, extend it from graphs (as given by Gajarský et al.) to arbitrary structures and eventually prove that every class of vertex- and edge-coloured directed graphs FO interpretable in some class of bounded degree is near-uniform.
- In Section 6, I will give the notion of near-uniformity an algorithmic flavour by showing how to efficiently interpret any near-uniform class of vertex- and edge-coloured directed graphs in some class of bounded degree.

PRELIMINARIES

General Notation

$\mathbb{N}, \mathbb{N}_+$ denote the sets of nonnegative integers and positive integers respectively. Let $k, k' \in \mathbb{N}$. By $[k, k']$ we denote the set of integers not smaller than k and not greater than k', and we also abbreviate $[k] := [1, k] \subseteq \mathbb{N}_+$. Let A, B be sets for this whole section. The *power set of* A is denoted $\mathtt{Pot}(A)$. The set of all k-element subsets of A is denoted $\binom{A}{k}$. Thus if A is finite, then $\mathtt{Pot}(A) = \dot\bigcup_{i \in \mathbb{N}} \binom{A}{i}$. $\binom{A}{k}$

If (and only if) $A \cap B = \emptyset$, we might write $A \,\dot\cup\, B$ instead of $A \cup B$ to denote the union of A and B.

A *family on* A *(in* B*)* is a subset $f \subseteq A \times B$ such that for all $a \in A$, there is exactly one $b \in B$ with $(a, b) \in f$. For all $a \in A$, we denote $f_a := b$ for the unique $b \in B$ with $(a, b) \in f$. We denote $(f_a)_{a \in A} := \{(a, f_a) \mid a \in A\}$. In the context of families (or mappings) we also use the notation $a \mapsto b := (a, b)$; with this notation, it is $(f_a)_{a \in A} = \{a \mapsto f_a \mid a \in A\}$.

Let $l, k \in \mathbb{N}$ and f' be a family on $[k]$ in B. Then we call f' a *(*k*-)tuple (in* B*)* and write $(f'_1, \ldots, f'_l) := f'_1, \ldots, f'_l := f'_1 \ldots\ f'_l := (f'_i)_{i \in [l]}$. If $A = [l]$, we sometimes identify the pair $((f_i)_{i \in [l]}, (f'_i)_{i \in [k]})$ with the $l + k$-tuple $(f_i)_{i \in [l]} \,\dot\cup\, (f'_{i-l})_{i \in [l+1, l+k]}$.

From time to time, we denote $\bar f := f$ to highlight the fact that f is a tuple. $\bar f$
We also might identify $f = \{f_a \mid a \in A\}$. Let $a \in A$. Then we denote $f[a/b] = \bar f[a/b] := \{(a', f_{a'}) \mid a' \in A \setminus \{a\}\} \dot\cup \{(a, b)\}$. For all families $(C_a)_{a \in A}$ $\bar f[a/b]$
such that for all $a \in A$, C_a is a set, we denote by $\prod_{a \in A} C_a$ the set of families g on A such that for all $a \in A$, it is $g_a \in C_a$. We also abbreviate $B^A := \prod_{a \in A} B$. We let $B^l := B^{[1,l]}$ be the set of l-tuples in B. We also let $B^* := \bigcup_{m \in \mathbb{N}} B^m$. If $|f| = 1$, we usually identify $f = f_a$ for the unique $a \in A$. In particular, we identify $B = B^1$.

A *mapping* is a triple (A, B, m), where m is a family on A in B. In contrast to families, we adopt our notation for mappings as follows: For all $a \in A$, we write $m(a) := m_a$. We usually denote $m := (A, B, m)$ and also write $m : A \longrightarrow B$. Let $A' \subseteq A, B' \subseteq B$ and $b \in B$. Furthermore, we denote

$$\mathtt{dom}(m) := A, \mathtt{rg}(m) := B, m(A') := \{m(a) \mid a \in A'\},$$ mappings

$$\mathtt{im}(m) := m(A), m^{-1}(B') := \{a \in A \mid m(a) \in B'\}, m^{-1}(b) := m^{-1}(\{b\}).$$

The main difference between a family and a mapping is that the set B is associated with the mapping m, but not with the family f. For example, for all $C \supseteq B$, f is also a family in C, whereas $\mathtt{rg}(m) = C$ if and only if $C = B$. The family f on A in B can be extended to a mapping (A, B, f).

Let C be another set with $C \cap A = \emptyset$ and C' any arbitrary set. Let $m' : C \longrightarrow B, m'' : B \longrightarrow C'$ be two more mappings and f'' a family on B in C'. We define the *composition* $f'' \circ f := \{(a, f''_{f_a}) \mid a \in A\}$ *of* f *with* f''. We also define the *union* $m \cup m' := (A \,\dot\cup\, C, B, m \cup m')$ *of* m *with* m', and the *composition* $m'' \circ m := (A, C'', m'' \circ m)$ *of* m *with* m''.

The mapping m induces the mapping

$$\bar m^k : A^k \longrightarrow B^k,\ (a_1, \ldots, a_k) \mapsto (m(a_1), \ldots, m(a_k)).$$

We might identify $m = \bar m^k$. Let $\bar a \in A^k, \bar b \in B^k$. Then $\bar a \mapsto \bar b$ denotes the family $\{a_i \mapsto b_i \mid i \in [k]\}$.

Let $r \in \mathbb{N}_+$. A *relation on* A *of arity* r is a triple (R, A, r) such that R is a set with $R \subseteq A^r$. Usually we view $R := (R, A, r)$, but it is important to not forget that A and r are associated with the relation R. A relation on A of arity 2 is also called a *binary relation on* A and a relation on A of arity 1 is also called a *monadic relation on* A. Let R be a binary relation on A. Then R is *reflexive* if for all $a \in A$, $(a, a) \in R$, *symmetric* if for all $a, b \in A$, if $(a, b) \in R$ then $(b, a) \in R$, *transitive* if for all $a, b, c \in A$, if $(a, b), (b, c) \in R$ then $(a, c) \in R$.

An equivalence relation $\sim$ on A is a binary relation on A which is reflexive, symmetric and transitive. Let $a \in A$. Then we call $[a]_\sim := \{b \in A \mid (a, b) \in A\}$ the $\sim$*-equivalence class of* a or the *equivalence class of* a *w.r.t.* $\sim$. We also denote $[a] := [a]_\sim$. An *equivalence class of* $\sim$ is the $\sim$-equivalence class of some $a \in A$. The *index of* $\sim$ is the number of equivalence classes of $\sim$ and is denoted $\mathtt{index}(\sim)$. If $\mathtt{index}(\sim) = 1$, then we say that $\sim$ is the *full relation on* A, and if $\mathtt{index}(\sim) = |A|$, then we say that $\sim$ is the *discrete equivalence on* A. Let $\sim'$ be another equivalence relation on A. We say that $\sim$ is a *refinement of* $\sim'$ if for all $a, b \in A$ with $a \sim b$, it is also $a \sim' b$. If $\sim$ is a refinement of $\sim'$, then $\sim'$ is called a *coarsening of* $\sim$. Obviously, $\sim$ is a refinement of the full relation on A and a coarsening of the discrete equivalence on A. Let $A' \subseteq A$. By $\sim[A'] := \sim \cap A'^2$ we denote the *restriction of* $\sim$ *to* A'.

Logic

For all $r \in \mathbb{N}_+$, we fix a fresh sequence $R_1^{(r)}, R_2^{(r)}, \ldots$ of *predicates of arity* r. We define the mapping

predicates and relations

$$\mathtt{ar} : \{R_i^{(r)} \mid r, i \in \mathbb{N}_+\} \longrightarrow \mathbb{N}_+,\ R_i^{(r)} \mapsto r,$$

that maps a predicate onto its arity. The predicates of arity 1 are also called *monadic predicates* or *colour predicates* or *label predicates*. Predicates of arity 2 are called *binary predicates*, and relations of arity 1 or 2 are called *monadic relations* or *binary relations* respectively. A *signature* is a finite subset of the set of all predicates of any arity. For all $r \in \mathbb{N}_+, i \in \mathbb{N}$, we define $\mathcal{R}_i^{(r)} := \{R_1^{(r)}, \ldots, R_i^{(r)}\}$. For all $l, m \in \mathbb{N}$ we denote $\mathbf{E}_{l,m} := \mathcal{R}_l^{(1)} \dot\cup \mathcal{R}_m^{(2)}$.

$\mathbf{E}_{l,m}$

σ_r

Let σ be a signature. For all $r \in \mathbb{N}_+$, we let $\sigma_r \subseteq \sigma$ be the subset of σ containing all predicates of arity r in σ. So, it is $\sigma = \bigcup_{r \in \mathbb{N}_+} \sigma_r$, and all but finitely many σ_r for $r \in \mathbb{N}_+$ are empty.

FO

The first order predicate logic (together with its syntax and semantics) is denoted FO, although FO also denotes the set of FO formulae (on any signature). The first order language with signature σ is denoted $\mathrm{FO}[\sigma]$. If $\sigma = \{R\}$, then we also write $\mathrm{FO}[R] := \mathrm{FO}[\{R\}]$. A first order formula in $\mathrm{FO}[\sigma]$ is also called a σ*-formula*. We fix a family $\mathtt{Var} := (v_i)_{i \in \mathbb{N}_+} = \{v_1, v_2, v_3, \ldots\}$ of fresh *variables* on $\mathbb{N}_+$. Note that variables inherit a linear ordering from $\mathbb{N}_+$, and so it makes sense to say that $v_3 < v_4, v_2 \not< v_1$. In practice, we usually rename $v_1, v_2, \ldots$ to other letters like $x, y, z, x_1, y_{13}, z_{42}$, where it then still holds, for example, that $z_{42} \in \mathtt{Var}$. Let $\psi \in \mathrm{FO}$. The set of the variables occurring free in ψ is denoted $\mathtt{free}(\psi)$. Let $k \in \mathbb{N}$. If $\mathtt{free}(\psi) \subseteq \{x_1, \ldots, x_k\}$, we also write $\psi(x_1, \ldots, x_k) := \psi$. Moreover, let for every $k \in \mathbb{N}$, $\mathrm{FO}_k[\sigma] \subset \mathrm{FO}[\sigma]$ denote the set of all σ-formulae ψ with $|\mathtt{free}(\psi)| \leqslant k$. The *arity of* ψ is the number $|\mathtt{free}(\psi)|$. A σ-formula in $\mathrm{FO}_0[\sigma]$ is called a σ*-sentence*. Denote $\mathtt{true} := \forall x\ x = x$ and $\mathtt{false} := \exists x\ x \neq x$.

Let $\psi \in \mathrm{FO}_k[\sigma]$. Let $\mathfrak{A}$ be a σ-structure (see 2.3). Let $\mathcal{I} \in A^{\mathtt{free}(\psi)}$. Then we write $\mathfrak{A} \models \psi(\mathcal{I})$ if ψ is true in $\mathfrak{A}$ under the variable assignment $\mathcal{I}$. If

$\mathtt{free}(\psi) = \{x_1, \dots, x_k\}$ and $\bar{a} \in A^k$, then we also write $\mathfrak{A} \models \psi(\bar{a})$ instead of $\mathfrak{A} \models \psi(\bar{x} \mapsto \bar{a})$. The *extension* $\psi(\mathfrak{A})$ *of* ψ *in* $\mathfrak{A}$ is defined to be the relation on A of arity $|\mathtt{free}(\psi)|$ containing exactly the families $\bar{a}$ over $\mathtt{free}(\psi)$ such that $\mathfrak{A} \models \psi(\bar{a})$. In this case, we usually identify the family $\bar{a} = (a_x)_{x \in \mathtt{free}(\psi)}$ with the $|\mathtt{free}(\psi)|$-tuple $(a_{x_i})_{i \in [\mathtt{free}(\psi)]}$, which we can do because $\mathtt{Var}$ has an underlying linear ordering.

$\mathfrak{A} \models \psi(\bar{a})$
$\psi(\mathfrak{A})$

The relation $\psi(\mathfrak{A}) \subseteq A^{\mathtt{free}(\psi)}$ is also called *elementarily definable in* $\mathfrak{A}$ and we say that ψ *defines* $\psi(\mathfrak{A})$ *in* $\mathfrak{A}$.

Structures

Let σ be a signature. A σ*-structure* $\mathfrak{A}$ is a pair (A, λ) where A is a finite set, called the *universe of* $\mathfrak{A}$, and λ a family on σ such that for all $R \in \sigma$, $\lambda_R \subseteq A^{\mathtt{ar}(R)}$ is a relation on A of arity $\mathtt{ar}(R)$. So, our structures are always understood to be finite. Note that $(\lambda_R)_{R\in\sigma} = \{(R, \lambda_R) \mid R \in \sigma\}$. For convenience (and tradition), we also write $V(\mathfrak{A}) := A$ and for all $R \in \sigma$, $R(\mathfrak{A}) := \lambda_R$. Now, we can say that $\mathfrak{A} = (V(\mathfrak{A}), (R(\mathfrak{A}))_{R\in\sigma})$. Let $a \in V(\mathfrak{A})$. Then (and only then) we also say $a \in \mathfrak{A}$. We call the elements of $V(\mathfrak{A})$ *vertices* (note that $\mathfrak{A}$ need not necessarily be a graph for this). If we denote a σ-structure by one of the letters $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}, \mathfrak{A}', \dots$, then in this case we often denote its universe by the corresponding Latin letter $A, B, C, A', \dots$ instead of $V(\mathfrak{A}), V(\mathfrak{B}), V(\mathfrak{C}), V(\mathfrak{A}'), \dots$.

σ-structures
$V(\mathfrak{A}) = A$

Let $\mathfrak{A}, \mathfrak{B}$ be σ-structures. Then an isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$ is a bijection $\pi : A \to B$ such that for all $R \in \sigma$, $\pi(R(\mathfrak{A})) = R(\mathfrak{B})$. In this case, we also write $\pi : \mathfrak{A} \cong \mathfrak{B}$. $\mathfrak{A}$ and $\mathfrak{B}$ are called *isomorphic*, written $\mathfrak{A} \cong \mathfrak{B}$, if there exists an isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$. An isomorphism from $\mathfrak{A}$ to $\mathfrak{A}$ is also called an *automorphism of* $\mathfrak{A}$.

isomorphisms

A class $\mathcal{C}$ is a collection of various objects, usually structures. The containment of some object $\mathfrak{A}$ in the class $\mathcal{C}$ is denoted by $\mathfrak{A} \in \mathcal{C}$. Let $\mathcal{C}$ be a class only containing structures of various signatures. For all signatures σ and all σ-structures $\mathfrak{A} \cong \mathfrak{A}'$ such that $\mathfrak{A} \in \mathcal{C}$, we also have $\mathfrak{A}' \in \mathcal{C}$. So a class is *closed under isomorphisms*. A *subclass* of $\mathcal{C}$ is a class $\mathcal{D}$ such that every structure contained in $\mathcal{D}$ is also contained in $\mathcal{C}$.

$\mathtt{Str}(\sigma)$ denotes the class of all σ-structures. A σ*-class* is a subclass of $\mathtt{Str}(\sigma)$. For all predicates R, we also denote $\mathtt{Str}(R) := \mathtt{Str}(\{R\})$ and say R-class instead of $\{R\}$-class. Let $\mathfrak{A}, \mathfrak{B}$ be σ-structures. Then $\mathfrak{B}$ is a *substructure of* $\mathfrak{A}$, denoted $\mathfrak{B} \subseteq \mathfrak{A}$, if $B \subseteq A$ and for all $R \in \sigma$, $R(\mathfrak{B}) \subseteq R(\mathfrak{A})$. $\mathfrak{B}$ is an *induced substructure of* $\mathfrak{A}$ if $\mathfrak{B}$ is a substructure of $\mathfrak{A}$ and for all $R \in \sigma$, we have $R(\mathfrak{B}) = R(\mathfrak{A}) \cap B^{\mathtt{ar}(R)}$. Note that for every $B' \subseteq A$, there is exactly one induced substructure of $\mathfrak{A}$ with universe B'; we call it the B'*-induced substructure of* $\mathfrak{A}$ and denote it by $\mathfrak{A}[B']$. We call $|\mathfrak{A}| := |A|$ the *order of* $\mathfrak{A}$ and $||\mathfrak{A}|| := \sum_{R\in\sigma} |R(\mathfrak{A})|$ the *size of* $\mathfrak{A}$.

σ-class
substructures
order and size

Let $\sigma' \subseteq \sigma$. The σ'*-reduct of* $\mathfrak{A}$ is the σ'-structure $\mathfrak{A}^{\sigma'} := (A, (R(\mathfrak{A}))_{R\in\sigma'})$. For all $R \in \sigma$, we also call $\mathfrak{A}^R := \mathfrak{A}^{\{R\}}$ the R*-reduct of* $\mathfrak{A}$. Note that the σ-structure can be seen as being obtained by stacking all of its R-reducts on top of each other. In particular, in order to describe $\mathfrak{A}$, it is sufficient to describe all of its R-reducts. For a σ-class $\mathcal{C}$, we also define the σ'-class $\mathcal{D}^{\sigma'}$ of σ'-reducts of σ-structures in $\mathcal{C}$, that is, we let

σ'-reducts

$$\mathcal{D}^{\sigma'} := \left\{\mathfrak{A}^{\sigma'} \mid \mathfrak{A} \in \mathcal{D}\right\}.$$

Again, for all $R \in \sigma$ we let $\mathcal{D}^R := \mathcal{D}^{\{R\}}$.

$\mathcal{D}^R$

$\mathfrak{A} \cup \mathfrak{B}$

Let σ, τ be signatures. Let $\mathfrak{A}$ be a τ-structure and $\mathfrak{B}$ a σ-structure. Then the *union* $\mathfrak{A} \cup \mathfrak{B}$ *of* $\mathfrak{A}$ *and* $\mathfrak{B}$ is defined to be the $\sigma \cup \tau$-structure

$$\mathfrak{A} \cup \mathfrak{B} := (A \cup B, (R(\mathfrak{A}))_{R \in \sigma} \cup (R(\mathfrak{B}))_{R \in \tau}).$$

If $A = B$, we also abbreviate

$$\mathfrak{A} \cup (R(\mathfrak{B}))_{R \in \sigma} := \mathfrak{A} \cup \mathfrak{B}.$$

Graphs and Neighbourhoods

(l, m)-labeled directed graphs

coloured sets

Let $l, m \in \mathbb{N}$. Recall that $\mathbf{E}_{l,m} = \mathcal{R}_l^{(1)} \dot\cup \mathcal{R}_m^{(2)} = \{R_1^{(1)}, \dots, R_l^{(1)}\} \dot\cup \{R_1^{(2)}, \dots, R_m^{(2)}\}$. $\mathbf{E}_{l,m}$-structures are called (l, m)*-labeled directed graphs*. They are structures of the form $(V, P_1, \dots, P_l, E_1, \dots, E_m)$ where the P_i can be seen as vertex colours and the E_i as edge (or vertex pair) colours. A $(0, 1)$-labeled directed graph is also called an *(unlabeled) directed graph*. A (l, m)-labeled directed graph is also said to be *vertex- and edge-coloured*. Let $G \in \mathtt{Str}(\mathbf{E}_{l,m})$ and $i \in [m]$. Let σ be a signature s.t $\sigma = \sigma_1$ (so σ only contains monadic predicates). Then σ-structures are also called *coloured sets* or, more specifically, $|\sigma|$*-coloured sets*.

$\vec{\mathcal{G}}^{(l,m)}$

$\mathcal{G}$

We define $\vec{\mathcal{G}}^{(l,m)} := \mathit{Str}(\mathbf{E}_{l,m})$. Let $E := R_1^{(2)}$. An *(undirected) graph* G is a E-structure such that $E(G)$ is symmetric and irreflexive. Let $\mathcal{G}$ denote the class of graphs. Let G be a graph. A *subgraph of* G is a substructure of G.

paths

$\mathtt{dist}^G(v, w)$

r-neighbourhood

maximum degree

Let $G \in \mathcal{G}$ and $V := V(G)$. Let $v, w \in V, r \in \mathbb{N}$. A *path from* v *to* w *in* G *of length* r is an $(r+1)$-tuple $v = v_0 \dots v_r = w$ such that for all $i \in [0, r-1]$, $(v_i, v_{i+1}) \in E(G)$. The *distance* $\mathtt{dist}^G(v, w)$ *between* v *and* w *in* G is defined as the length of a shortest path between v and w in G. We call $N_r^G(v) := \{w' \in V(G) \mid \mathtt{dist}^G(v, w') \leqslant r\}$ the r*-neighbourhood of* v *in* G. Sometimes, we identify $N_r^G(v) = G[N_r^G(v)]$ but it should always be clear from context whether we mean the set or the induced subgraph. We call $N^G(v) := N_1^G(v) \setminus \{v\}$ the *neighbourhood of* v *in* G. Sometimes, we omit the superscript ^{G} if the graph is clear from context. The *degree* $\deg^G(v)$ *of* v *in* G is the number $|N^G(v)|$ of neighbours of v in G. For all $d \in \mathbb{N}$, G is called d*-regular* if for all $v \in V(G)$, $\deg^G(v) = d$. The *maximum degree* $\Delta(G)$ *of* G is the number $\max_{v \in V} \deg(v)$.

Gaifman Graph

$G_{\mathfrak{A}}$

These definitions easily extend to arbitrary structures via the Gaifman Graph: Let σ be a signature and $\mathfrak{A}$ a σ-structure. Then we define the *Gaifman Graph of* $\mathfrak{A}$ to be the graph $G_{\mathfrak{A}} := (A, E(G_{\mathfrak{A}})) \in \mathcal{G}$ where

$$E(G_{\mathfrak{A}}) := \{\, vw \mid \text{ there are } R \in \sigma, \bar{a} \in R(\mathfrak{A}) \text{ such that } v \neq w \in \bar{a} \,\}.$$

$\mathcal{C}_d$

Let $a, b \in A$. The *distance* $\mathtt{dist}^{\mathfrak{A}}(a, b)$ *between* a *and* b *in* $\mathfrak{A}$ is the distance between a and b in $G_{\mathfrak{A}}$. For every $r \in \mathbb{N}$, the r*-neighbourhood of* a *in* $\mathfrak{A}$ is the r-neighbourhood of a in $G_{\mathfrak{A}}$. The *neighbourhood of* a *in* $\mathfrak{A}$ is the neighbourhood of a in $G_{\mathfrak{A}}$. The *degree* $\deg^{\mathfrak{A}}(a)$ *of* a *in* $\mathfrak{A}$ is the degree $\deg^{G_{\mathfrak{A}}}(a)$ of a in $G_{\mathfrak{A}}$. The *maximum degree* $\Delta(\mathfrak{A})$ *of* $\mathfrak{A}$ is the maximum degree $\Delta(G_{\mathfrak{A}})$ of $G_{\mathfrak{A}}$. Let $\mathcal{C}$ be a σ-class. $\mathcal{C}$ is *of bounded degree* if there is a $d \in \mathbb{N}$ such that for all $\mathfrak{A} \in \mathcal{C}$, $\Delta(\mathfrak{A}) \leqslant d$. For all $d \in \mathbb{N}$, we denote by $\mathcal{C}_d$ the subclass of $\mathcal{C}$ containing exactly the σ-structures in $\mathcal{C}$ of maximum degree at most d. So for all $d \in \mathbb{N}$, $\mathcal{C}_d$ is, in particular, a σ-class of bounded degree. A class $\mathcal{C}$ of bounded degree is a class of structures such that for all structures in $\mathcal{C}$, each vertex of that structure occurs at most a globally bounded number of times in a tuple together with other vertices of the structure, so a vertex cannot be associated with "too many" other vertices.

FO Model Checking Problem

Let $\Sigma := \{0, 1\}$ be our *alphabet*. A *problem* is a mapping $\mathcal{P} : \Sigma^* \to \Sigma^*$. Tuples over Σ are called *strings* or *words*. A *decision problem* is a problem $\mathcal{P}$ with $\mathtt{im}(\mathcal{P}) \subseteq \{0, 1\}$. An algorithm, that, given an $x \in \Sigma^*$, computes $\mathcal{P}(x) \in \Sigma^*$, is said to *compute the problem* $\mathcal{P}$ or to *solve the problem* $\mathcal{P}$. An algorithm that computes a decision problem is said to *decide the decision problem*. The problem $\mathcal{P}$ is said to be *computable* if there is an algorithm that computes $\mathcal{P}$. Let σ be a signature and $\mathcal{C}$ a σ-class. The *First Order Model Checking Problem on* $\mathcal{C}$ is the decision problem to decide for every σ-structure $\mathfrak{A}$ in $\mathcal{C}$ and first order σ-formula $\varphi \in \mathrm{FO}_0[\sigma]$, whether $\mathfrak{A} \models \varphi$. This can be illustrated as follows: problems

$\mathtt{MC}(\mathcal{C})$

Input: $\mathfrak{A} \in \mathcal{C}, \varphi \in \mathrm{FO}_0[\sigma]$
Question: Does $\mathfrak{A} \models \varphi$ hold?

$\mathtt{MC}(\mathcal{C})$

The First Order Model Checking Problem $\mathtt{MC}(\mathcal{C})$ on $\mathcal{C}$ can be formalized as a decision problem in the following way: A structure's universe is taken to be an initial segment of $\mathbb{N}$ and the structure is encoded by a suitable encoding of the tuples in its relations and writing them down consecutively. A formula could be seen as a binary tree where the inner nodes are one of $\forall, \exists, \neg, \vee$ and the leaves are variables (which in turn could be represented as strings over $[0, 9]$). Then this binary tree could be encoded by writing down each layer in some way. The resulting string would be over some finite alphabet and could be easily converted to be over $\Sigma = \{0, 1\}$. The converted string would then be the input in the problem $\mathtt{MC}(\mathcal{C})$. The output would be 1 if $\mathfrak{A} \models \varphi$, and 0 otherwise. encoding of $\mathtt{MC}(\mathcal{C})$ as problem

For $k \in \mathbb{N}$, it is also possible to generalize $\mathtt{MC}(\mathcal{C})$ to the problem with input $\mathfrak{A} \in \mathcal{C}, \bar{a} \in A^k, \varphi \in \mathrm{FO}_k[\sigma]$ and the question whether $\mathfrak{A} \models \varphi(\bar{a})$ holds. But this problem easily reduces to $\mathtt{MC}(\mathcal{C})$ by adjusting the signature σ to contain k additional constants (or monadic relations which each contain exactly one element), substituting the free variables in φ by these constants and interpreting the constants in $\mathfrak{A}$ by $\bar{a}$ respectively. Then for example, we have that $\mathcal{C}$ is a class of bounded degree if and only if $\mathcal{C}'$ is of bounded degree, where $\mathcal{C}'$ is the class obtained from $\mathcal{C}$ by creating new structures from the σ-structures in $\mathcal{C}$ as described above. In general, inserting constants usually does not worsen a class a lot.

We also call the First Order Model Checking Problem on $\mathcal{C}$ the *Model Checking Problem on* $\mathcal{C}$ or *Model Checking on* $\mathcal{C}$.

Complexity

Let $\mathcal{P}$ be a problem and $\kappa : \Sigma^* \to \mathbb{N}$ a function. We say that $\mathcal{P}$ is *fixed-parameter tractable with parameter function* κ or *fpt with parameter function* κ if $\mathcal{P}$ is computable in time fpt

$$x \mapsto f(\kappa(x))\, |x|^c$$

for some computable function $f : \mathbb{N} \to \mathbb{N}$ and constant $c \in \mathbb{N}$. An algorithm which solves the problem $\mathcal{P}$ in this time is called an *fpt algorithm for* $\mathcal{P}$.

Let σ be a signature and $\mathcal{C}$ a σ-class. We say that $\mathtt{MC}(\mathcal{C})$ is *fixed-parameter tractable* or *fpt* if $\mathtt{MC}(\mathcal{C})$ is fpt with parameter function

$$\kappa : \Sigma^* \to \mathbb{N}, \ [\![(\mathfrak{A}, \varphi)]\!] \mapsto |\varphi|$$

where $[\![\cdot]\!]$ denotes some suitable encoding function.
A recent result by Grohe, Kreutzer and Siebertz states that Model Checking on the class of all nowhere dense graph classes is fpt [see [3] for definition and result].

efficient algorithm

An *efficient algorithm* is an algorithm whose runtime function of the input length can be bounded by a polynomial function of the input length.

uniform cost model

All of our runtime analysis will be done on the basis of a uniform cost model, that is, we will consider any addition, multiplication, comparison etc. as an operation with cost 1. Consider, though, that in reality, the cost of such an operation is the number of bits involved, that is, the maximum of the logarithms of the numbers that are added, multiplied, compared etc.

FO INTERPRETATIONS

Now, we introduce FO interpretations which will be an important tool in this thesis. FO interpretations are families of formulae which, in essence, define new structures from old ones. From now on, throughout the whole thesis, let σ, τ be arbitrary (finite, relational) signatures (unless otherwise stated).

Definition and Elementary Properties

We start by formally defining an important tool for our work.

Definition 3.1. *An* FO-interpretation Γ *from* τ *to* σ *is a pair*

$$\Gamma = (\psi_{univ}(x), (\psi_R(\bar{x}))_{R\in\sigma}),$$

where $\psi_{univ} \in \mathrm{FO}_1[\tau]$ *has arity* 1 *and for all* $R \in \sigma$, $\psi_R(\bar{x}) \in \mathrm{FO}_{ar(R)}[\tau]$ *has arity* $ar(R)$. Γ *is called* simple *if* $\psi_{univ} \equiv \mathtt{true}$. Γ *induces the following two mappings:*

$$sem_\Gamma : \mathtt{Str}(\tau) \to \mathtt{Str}(\sigma), \mathfrak{A} \mapsto sem_\Gamma(\mathfrak{A}),$$

$$syn_\Gamma : \mathrm{FO}[\sigma] \to \mathrm{FO}[\tau], \varphi \mapsto syn_\Gamma(\varphi).$$

Let $\mathfrak{A}$ *be a* τ*-structure and* φ *a* σ*-formula. Then* $sem_\Gamma(\mathfrak{A})$ *is defined as* $sem_\Gamma(\mathfrak{A}) := (\psi_{univ}(\mathfrak{A}), \left(\psi_R(\mathfrak{A}) \cap \psi_{univ}(\mathfrak{A})^{ar(R)}\right)_{R\in\sigma})$ *and* $syn_\Gamma(\varphi)$ *is defined by structural induction on* φ *as follows:*

$$\begin{aligned} syn_\Gamma(x = y) &:= x = y, \\ syn_\Gamma(R\bar{x}) &:= \psi_R(\bar{x}), \\ syn_\Gamma(\neg\theta) &:= \neg syn_\Gamma(\theta), \\ syn_\Gamma(\theta \vee \xi) &:= syn_\Gamma(\theta) \vee syn_\Gamma(\xi), \\ syn_\Gamma(\exists x\theta) &:= \exists x(\psi_{univ}(x) \wedge syn_\Gamma(\theta)) \end{aligned}$$

for all variables $x, y, \bar{x}$*, predicates* $R \in \sigma$ *and* σ*-formulae* θ, ξ.
Note that $sem_\Gamma(\mathfrak{A}) \in \mathtt{Str}(\sigma)$ *might be empty.* sem_Γ *is called the semantic mapping induced by* Γ *and* syn_Γ *is called the syntactic mapping induced by* Γ. *We use* Γ *to denote both mappings* sem_Γ *and* syn_Γ*, but it will be clear from context which mapping is meant. For all* $\sigma' \subseteq \sigma$*, we define the* σ'-restriction $\Gamma^{\sigma'} := (\psi_{univ}, (\psi_R)_{R\in\sigma'})$ of Γ. *For all* $R \in \sigma$*, we denote* $\Gamma^R := \Gamma^{\{R\}}$. *The set of all* FO *interpretations from* τ *to* σ *is denoted* $\mathtt{INT}^\sigma_\tau$. *We let* $|\Gamma| := \max\{\, |\psi_{univ}|, |\psi_R| \mid R \in \sigma\}$.

Let Γ be an FO interpretation from τ to σ. Then both mappings $\mathtt{sem}_\Gamma$ and $\mathtt{syn}_\Gamma$ are trivially polynomial time computable; in fact $\mathtt{sem}_\Gamma$ is computable in time $\mathcal{O}(|\mathfrak{A}|^{|\Gamma|})$ and $\mathtt{syn}_\Gamma$ is computable in time $\mathcal{O}(|\varphi|)$. The semantic mapping is computable by an algorithm that computes the extensions of the formulae in Γ which can be done by iterating through all tuples in $A^{qr(\psi_R)}$ for $R \in \sigma$. The syntactic mapping is computable in linear time by an algorithm that recursively replaces certain subformulae by others.

It may be helpful to think of Γ as a *logical decoder of* $\mathtt{Str}(\tau)$: A τ-structure $\mathfrak{A}$ is seen as an encoding for the unique σ-structure $\Gamma(\mathfrak{A}) =: \mathfrak{B}$. The fact that $\mathtt{sem}_\Gamma$ is polynomial time computable then means that there is an algorithm that, given an encoding from $\mathtt{Str}(\tau)$, can decode the encoding. Now, we might ask: How many encodings does $\mathfrak{B}$ have? Obviously, there is exactly one

interpretations as decoders

such encoding for all σ-structures if and only if Γ is injective. But Γ is usually not injective. In fact, the preimages of $\mathfrak{B}$ might not even be isomorphic. This results in the fact that, in general, there are many nonisomorphic encodings of $\mathfrak{B}$ in $\mathtt{Str}(\tau)$. These considerations motivate the following definition.

Γ-encoding

Definition 3.2. *Let Γ be an FO interpretation from τ to σ and $\mathfrak{B}$ a σ-structure. Let $\mathcal{C}$ be a τ-class and $\mathfrak{A} \in \mathcal{C}$ with $\Gamma(\mathfrak{A}) = \mathfrak{B}$. Then we call $\mathfrak{A}$ a* Γ-encoding of $\mathfrak{B}$ (in $\mathcal{C}$).

We choose to give this more restrictive definition of Γ-encoding, where it is required that $\Gamma(\mathfrak{A}) = \mathfrak{B}$ instead of just $\Gamma(\mathfrak{A}) \cong \mathfrak{B}$. This will be more suitable to us and since our classes are always closed under isomorphisms, it holds for every τ-class $\mathcal{C}$, that if there is a τ-structure $\mathfrak{A} \in \mathcal{C}$ with $\Gamma(\mathfrak{A}) \cong \mathfrak{B}$ then there is also a τ-structure $\mathfrak{A}' \in \mathcal{C}$ with $\Gamma(\mathfrak{A}') = \mathfrak{B}$.
The following lemma follows easily from the definition of FO interpretations by structural induction on the formula:

Lemma 3.3 (Interpretation Lemma)**.** *Let Γ be an FO interpretation from τ to σ. Then for all τ-structures $\mathfrak{A}$ and all σ-formulae φ:*

$$\mathfrak{A} \models \Gamma(\varphi) \iff \Gamma(\mathfrak{A}) \models \varphi.$$

Before we take a deep dive, we want to look at simple facts about FO interpretations. At first we observe a connection between the first component ψ_{univ} of an FO interpretation and substructures of the τ-structure an FO interpretation is applied to.

Observation 3.4. *Let $\Gamma := (\psi_{univ}, (\psi_R)_{R\in\sigma})$ be an FO interpretation from τ to σ and $\Gamma' := (\mathit{true}, (\psi_R)_{R\in\sigma})$. Let $\mathfrak{A}$ be a τ-structure and denote $A' := \psi_{univ}(\mathfrak{A})$. Then*

$$\Gamma(\mathfrak{A}) = \Gamma'(\mathfrak{A})[A'].$$

In other words: The Γ-encoding of $\mathfrak{A}$ is the $\psi_{univ}(\mathfrak{A})$-induced substructure of the Γ'-encoding of $\mathfrak{A}$.

Proof. It is

$$V(\Gamma(\mathfrak{A})) = \psi_{univ}(\mathfrak{A}) = A' = V(\Gamma'(\mathfrak{A})[A']).$$

Furthermore, for all $R \in \sigma$, we have

$$\Gamma'(\mathfrak{A})[A']^R = \Big(\psi_R(\mathfrak{A}) \cap \mathtt{true}(\mathfrak{A})^{\mathsf{ar}(R)}\Big) \cap A'^{\mathsf{ar}(R)}$$

$$= \psi_R(\mathfrak{A}) \cap \psi_{univ}(\mathfrak{A})^{\mathsf{ar}(R)} = \Gamma(\mathfrak{A})^R.$$

$\square$

The following observation shows the connection between restrictions of FO interpretations and reducts of structures.

Observation 3.5. *Let $\Gamma := (\psi_{univ}, (\psi_R)_{R\in\sigma})$ be an FO interpretation from τ to σ. Let $\mathfrak{A}$ be a τ-structure and $\sigma' \subseteq \sigma$. Then it is $\Gamma^{\sigma'}(\mathfrak{A}) = \Gamma(\mathfrak{A})^{\sigma'}$. In particular, it is $\Gamma^{\emptyset}(\mathfrak{A}) = (\psi_{univ}(\mathfrak{A}))$.*

Proof. It is

$$\Gamma^{\sigma'}(\mathfrak{A}) = \Big(\psi_{univ}(\mathfrak{A}), (\psi_R(\mathfrak{A}) \cap \psi_{univ}(\mathfrak{A})^{\mathsf{ar}(R)})_{R\in\sigma'}\Big)$$

$$= \left(\psi_{univ}(\mathfrak{A}), (\psi_R(\mathfrak{A}) \cap \psi_{univ}(\mathfrak{A})^{ar(R)})_{R \in \sigma}\right)^{\sigma'} = \Gamma(\mathfrak{A})^{\sigma'}.$$

□

This observation shows us that the interpretation of a structure, thus its "decoding", can be broken down to the interpretation of individual predicates $R \in \sigma$. So if we want to specify $\Gamma(\mathfrak{A})$, it is sufficient to specify $\psi_{univ}(\mathfrak{A})$ and $\Gamma^R(\mathfrak{A})$ for all $R \in \sigma$.

FO Interpretations and Reductions between FO Model Checking Problems

In this subsection, we show how certain FO interpretations can be used as polynomial time reductions from the Model Checking Problem on some σ-class $\mathcal{D}$ to the Model Checking Problem on some τ-class $\mathcal{C}$.
We start by restricting FO interpretations to go from a subclass of $\mathtt{Str}(\tau)$ to a subclass of $\mathtt{Str}(\sigma)$.

Definition 3.6. *Let $\mathcal{D}$ be a σ-class and $\mathcal{C}$ a τ-class. If there is an* FO *interpretation Γ from τ to σ such that $\mathcal{D} \subseteq \Gamma(\mathcal{C})$, then we say that Γ* interprets $\mathcal{D}$ in $\mathcal{C}$ *or that* $\mathcal{D}$ is FO interpretable in $\mathcal{C}$ (via Γ).

So, $\mathcal{D}$ is FO interpretable in $\mathcal{C}$ via Γ if for every σ-structure $\mathfrak{B} \in \mathcal{D}$ there is a τ-structure $\mathfrak{A} \in \mathcal{C}$ such that $\Gamma(\mathfrak{A}) = \mathfrak{B}$, that is, such that $\mathfrak{A}$ is a Γ-encoding of $\mathfrak{B}$. Or again in other words: $\mathcal{D}$ is FO interpretable in $\mathcal{C}$ via Γ if and only if every σ-structure in $\mathcal{D}$ has a Γ-encoding in $\mathcal{C}$.
Now, what are FO interpretations actually good for? Can they be used to transfer fixed-parameter tractability from one class onto another? Let $\mathcal{D}$ be a σ-class FO interpretable in an "easy" τ-class $\mathcal{C}$, that is, where Model Checking on $\mathcal{C}$ is fpt. Then one could easily be seduced to think that now, since we can encode each σ-structure in $\mathcal{D}$ as a τ-structure in $\mathcal{C}$, by Lemma 3.3, Model Checking on $\mathcal{D}$ will be fpt as well. Yes, but only if we actually *can efficiently encode* each σ-structure in $\mathcal{D}$ as a τ-structure in $\mathcal{C}$. What we have given in this context, though, is a mere FO interpretation Γ such that for any σ-structure $\mathfrak{B}$ there *exists* (somewhere in the huge class $\mathcal{C}$) a τ-structure $\mathfrak{A}$ with $\Gamma(\mathfrak{A}) = \mathfrak{B}$. What we might lack (and usually do lack) is an algorithm that computes this existing function that maps each σ-structure in $\mathcal{D}$ to one of its Γ-encodings in $\mathcal{C}$. This motivates the following two definitions.
First, we formalize the idea of a reduction between decision problems. Recall that our alphabet is $\Sigma = \{0, 1\}$.

FO interpretations only *describe*, they do not *compute*

Definition 3.7. *Let $\mathcal{P}_1, \mathcal{P}_2$ be problems. Then a reduction from $\mathcal{P}_1$ to $\mathcal{P}_2$ is an algorithm that, given $p_1 \in \Sigma^*$, computes $p_2 \in \Sigma^*$ such that $\mathcal{P}_1(p_1) = \mathcal{P}_2(p_2)$.*

reductions

Now, we are about to see how an FO interpretation relates to a reduction from $\mathtt{MC}(\mathcal{D})$ to $\mathtt{MC}(\mathcal{C})$. The following definition corresponds to Definition 5.3 in [2].

Definition 3.8. *Let $\mathcal{D}$ be a σ-class and $\mathcal{C}$ a τ-class. A* (polynomial time) FO-reduction *from $\mathcal{D}$ to $\mathcal{C}$ is a pair $(\Gamma, \mathbf{A})$ such that*

(polynomial time) FO *reductions*

- *$\Gamma \in \mathit{INT}^{\sigma}_{\tau}$ such that Γ interprets $\mathcal{D}$ in $\mathcal{C}$,*
- *$\mathbf{A}$ is a polynomial time algorithm that, given a σ-structure $\mathfrak{B} \in \mathcal{D}$, computes a τ-structure $\mathfrak{A} \in \mathcal{C}$ such that $\Gamma(\mathfrak{A}) = \mathfrak{B}$.*

efficiently FO *interpretable*

efficient FO *interpretations*

If there is an FO *reduction* $(\Gamma, \mathbf{A})$ *from* $\mathcal{D}$ *to* $\mathcal{C}$, *then we say that* $\mathcal{D}$ *is* efficiently FO interpretable in $\mathcal{C}$ (via $(\Gamma, \mathbf{A})$). *If* $\mathcal{D}$ *is efficiently* FO *interpretable in* $\mathcal{C}$ *via* $(\Gamma, \mathbf{A})$ *and* Γ *is simple, then we say that* $\mathcal{D}$ *is* efficiently simply FO interpretable in $\mathcal{C}$ (via $(\Gamma, \mathbf{A})$). *Let* $\Gamma \in \mathit{INT}^{\sigma}_{\tau}$. *If there is an algorithm* $\mathbf{A}$ *such that* $(\Gamma, \mathbf{A})$ *is an* FO *reduction from* $\mathcal{D}$ *to* $\mathcal{C}$, *then* Γ *is called* $(\mathcal{D}, \mathcal{C})$-efficient. *If for all* σ*-classes* $\mathcal{D}'$ *and all* τ*-classes* $\mathcal{C}'$, Γ *is* $(\mathcal{D}', \mathcal{C}')$*-efficient, then* Γ *is called* efficient. *Otherwise,* Γ *is called* inefficient.

Here again, we choose to give the more restrictive definition where the algorithm $\mathbf{A}$ is required to compute an $\mathfrak{A} \in \mathcal{C}$ with $\Gamma(\mathfrak{A}) = \mathfrak{B}$ instead of just $\Gamma(\mathfrak{A}) \cong \mathfrak{B}$. Note, though, that if we have an algorithm that does the latter, it is not completely obvious how to modify this algorithm to obtain one that does the former. But all algorithms $\mathbf{A}$ we give in this thesis, will satisfy the stronger condition $\Gamma(\mathfrak{A}) = \mathfrak{B}$, and so it does not really make a difference to us. An example of an efficient interpretation from τ to σ is $(\mathtt{true}, (\mathtt{true})_{R \in \sigma})$. We will see an inefficient interpretation in the next section shortly.
Definition 3.8 is immediately put to use in the following lemma.

Lemma 3.9. *Let* $\mathcal{C}$ *be a* τ*-class and* $\mathcal{D}$ *a* σ*-class such that* $\mathcal{D}$ *is efficiently* FO *interpretable in* $\mathcal{C}$ *and* $\mathit{MC}(\mathcal{C})$ *is fpt. Then* $\mathit{MC}(\mathcal{D})$ *is fpt as well.*

Proof. Let $\mathfrak{B}$ be a σ-structure in $\mathcal{D}$ and φ a σ-formula. We reduce the problem of determining whether $\mathfrak{B} \models \varphi$ to the problem of determining whether $\mathfrak{A} \models \varphi^*$ for some $\mathfrak{A} \in \mathcal{C}$, $\varphi^* \in \mathrm{FO}[\tau]$.
Since $\mathcal{D}$ is efficiently FO interpretable in $\mathcal{C}$, there is an FO reduction $(\Gamma, \mathbf{A})$ from $\mathcal{D}$ to $\mathcal{C}$. Then we obtain $\mathfrak{A}, \varphi^*$ simply by applying $\mathbf{A}$ to $\mathfrak{B}$ and Γ to φ. See Figure 1.

$$
\begin{array}{ccc}
\text{Class } \mathcal{D} & & \text{Class } \mathcal{C} \\
\hline
\mathfrak{B} & \xmapsto{\mathbf{A}} & \mathbf{A}(\mathfrak{B}) =: \mathfrak{A} \\
\varphi & \xmapsto{\Gamma} & \Gamma(\varphi) =: \varphi^*
\end{array}
$$

Figure 1: An FO reduction in action

We agree on some way of encoding σ- and τ-structures. Let n be the length of the encoding of $\mathfrak{B}$ and m the length of the encoding of $\mathfrak{A}$. Without loss of generality, let $n \geqslant 2$. Let $d \in \mathbb{N}$ and $n \mapsto n^d$ be a function bounding the runtime of $\mathbf{A}$ for all input lengths $n \geqslant 2$. Note that $m \leqslant n^d$ and $|\varphi^*| \overset{\text{by Def. 3.1}}{\leqslant} |\varphi|\ |\Gamma|$. The given fpt algorithm for $\mathit{MC}(\mathcal{C})$ will thus have runtime

$$f(|\varphi^*|)m^c \leqslant f(|\varphi|\,|\Gamma|) \cdot (n^d)^c = f(|\varphi|\,|\Gamma|) \cdot n^{d \cdot c}$$

for some computable function $f : \mathbb{N} \longrightarrow \mathbb{N}$ and constant $c \in \mathbb{N}$. So the algorithm that first applies $(\Gamma, \mathbf{A})$ to $\mathfrak{B}, \varphi$ and then the given fpt algorithm on $\mathcal{C}$ to the result $\mathfrak{A}, \varphi^*$ is again an fpt algorithm. By Lemma 3.3, this algorithm is also correct and hence the claim follows. $\square$

NP HARDNESS OF COMPUTING CERTAIN Γ-ENCODINGS

In this section, we want to show that actually, not every FO interpretation is efficient (unless $\mathsf{P} = \mathsf{NP}$). Let E denote the binary predicate $R_1^{(2)}$ for this whole section. Recall that for all $d \in \mathbb{N}$, we denote by $\mathcal{G}_d$ the class of graphs of maximum degree at most d. More precisely, we will show that there is an FO interpretation Γ from $\{E\}$ to $\{E\}$ such that it is an NP-hard problem, given a graph $H \in \Gamma(\mathcal{G}_3)$, to compute a graph G of maximum degree at most 3 with $\Gamma(G) = H$. So not every interpretation of graphs of bounded maximum degree is efficient. We will show in the later sections, though, that for every class interpreted in a class of bounded degree there is another efficient interpretation in another class of bounded degree. So this section serves as a motivation for the effort made in the following sections.

bad news: there are inefficient FO interpretations

Let G be a graph and $k \in \mathbb{N}_+$. A *k-colouring of* G is a mapping $c : V(G) \longrightarrow [k]$ such that for all $uv \in E(G)$, it is $c(u) \neq c(v)$. The graph G is called *k-colourable* if there is a k-colouring of G. Let $\mathcal{C}, \mathcal{D}$ be classes of graphs and Γ an FO interpretation from $\{E\}$ to $\{E\}$. Then we define the following problems.

k-colouring

cENC(Γ,$\mathcal{C}$)

Input: a graph $H \in \Gamma(\mathcal{C})$
Output: a Γ-encoding of H in $\mathcal{C}$, that is, a graph $G \in \mathcal{C}$ with $\Gamma(G) = H$

cENC(Γ, $\mathcal{C}$)

c3COL($\mathcal{D}$)

Input: a graph $H \in \mathcal{D}$
Output: a 3-colouring of H if H is 3-colourable, and `Failure` otherwise

c3COL($\mathcal{D}$)

`m` stands for *modified*, `c` for *compute*, `3COL` for *3-colouring*, `ENC` for *encoding*

m-c3COL($\mathcal{D}$)

Input: a 3-colourable graph $H \in \mathcal{D}$
Output: a 3-colouring of H

m-c3COL($\mathcal{D}$)

Let $\mathcal{R} \subseteq \mathcal{G}_4$ be the class of 4-regular graphs. Then it is a known fact that the problem `c3COL`($\mathcal{R}$) is *NP-hard*. We will give an FO interpretation Γ and reduce `c3COL`($\mathcal{R}$) to `cENC`($\Gamma, \mathcal{G}_3$) (in polynomial time).

Lemma 4.1. *`m-c3COL`($\mathcal{R}$) is NP-hard.*

Proof. It suffices to reduce `c3COL`($\mathcal{R}$) to `m-c3COL`($\mathcal{R}$). Assume that **A** is a polynomial time algorithm for `m-c3COL`($\mathcal{R}$). Then there is a polynomial function $p : \mathbb{N} \longrightarrow \mathbb{N}$ that bounds its runtime for all lengths of the input. Then we construct our reduction as follows: Let H be a 4-regular graph of encoding length n. We compute $h := p(n)$ (which will be possible in polynomial time) and then simulate **A** on input H for $h = p(n)$ steps. If **A** answers with some output c during the simulation, we test (in linear time) whether c is a 3-colouring of H. If it is, we output c. If it is not, we output `Failure`. If **A** does not answer during our simulation at all, we also output `Failure`. Obviously, our algorithm runs in polynomial time and the correctness follows directly from the definition of `m-c3COL`($\mathcal{R}$).

So, if `m-c3COL`($\mathcal{R}$) is in `P`, then `c3COL`($\mathcal{R}$) is in `P`, too (although it would not be obvious how to find the bounding function p and thus the actual reduction algorithm). □

Now comes the main theorem of this section. This theorem means that, given a class $\mathcal{C}$ of bounded degree and an FO interpretation Γ to some signature of vertex- and edge-coloured directed graphs, it is in general not possible to FO reduce $\Gamma(\mathcal{C})$ to $\mathcal{C}$ in polynomial time. This theorem corresponds to Theorem 4.5 of [1], but our proof is slightly modified and also more detailed.

Theorem 4.2. *There is a simple* FO *interpretation* Γ *from* $\{E\}$ *to* $\{E\}$ *such that* $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$ *is* NP*-hard.*

Proof. We will define Γ further below. By Lemma 4.1, it suffices to reduce $\mathtt{m\text{-}c3COL}(\mathcal{R})$ to $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$. Assume that there is a polynomial time algorithm **A** that computes $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$. We want to construct a polynomial time algorithm for $\mathtt{m\text{-}c3COL}(\mathcal{R})$ using **A**.
The idea of the (polynomial time) reduction is illustrated in Figure 2. Note that each ellipse represents a class of graphs.

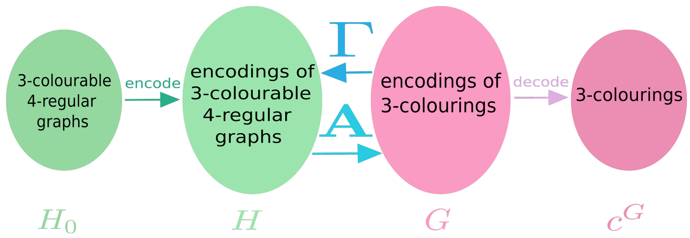

Figure 2: Polynomial Time Reduction from $\mathtt{m\text{-}c3COL}(\mathcal{R})$ to $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$

At first, we come up with an efficient encoding of 4-regular graphs. We also encode all possible 3-colourings of 4-regular graphs such that they are efficiently decodable. Then we give an FO interpretation Γ such that for all graphs $G \in \mathcal{G}_3$ and all graphs H from a special subclass of $\mathcal{G}_4$, we have $\Gamma(G) = H$ if and only if G is the encoding of a 3-colouring of the 4-regular graph encoded by H. Then, given a 3-colourable 4-regular graph H_0, we proceed as follows:

1. We encode it to obtain H. Actually, we will replace its vertices and edges by special gadgets defined below.

2. We use our hypothetical algorithm **A** for $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$ to determine the encoding G of some 3-colouring of the underlying graph. Note that G is technically still a graph.

3. We decode it to obtain a 3-colouring c^G of H_0. Actually, we will revert the above gadget replacements which we will able to do correctly because each vertex colour gadget will have a different number of edges.

In the following, we make our procedure more precise. Let H_0 be a 4-regular, 3-colourable graph. If H_0 is the empty graph, we output the empty mapping $\emptyset \longrightarrow [3]$ which is obviously a 3-colouring of H_0. So assume H_0 is not empty. Let T be the graph in Figure 3 and U the graph in Figure 4.
We call the vertices v^1, v^2, v^3, v^4 of T *terminals of* T and the vertices e^1, e^2 of U *terminals of* U. We construct H from H_0 in the following way:

construction of H from H_0

- Every vertex v of H_0 is replaced by a fresh copy T_v of T.

- Every edge e of H_0 is replaced by a fresh copy U_e of U.

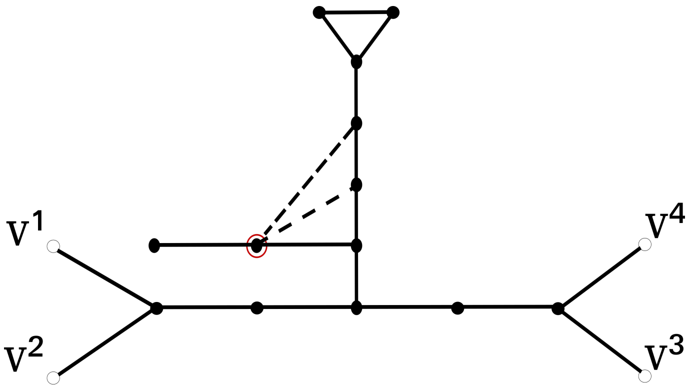

Figure 3: Vertex gadget T in the proof of Theorem 4.2

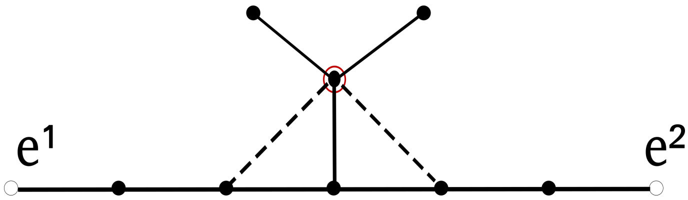

Figure 4: Edge gadget U in the proof of Theorem 4.2

- For every vertex v of H_0, let $N^{H_0}(v) = \{v_1, v_2, v_3, v_4\}$ be some ordering of its neighbours. Let $\{u, v\} \in \binom{V(H_0)}{2}$ such that $e := uv \in E(H_0)$. Let $i, j \in [4]$ such that $u = v_i, v = u_j$. Then we identify the first terminal of U_e with the j^{th} terminal of T_u and the second terminal of U_e with the i^{th} terminal of T_v.

The resulting graph is H. Note that since H_0 is not empty and 4-regular, there is at least one edge e in $E(H_0)$. Since there is a vertex of degree 5 in U_e, we have $H \notin \mathcal{G}_3$. Also note that $V(H) = \dot{\bigcup}_{v \in V(H_0)} T_v \cup \dot{\bigcup}_{e \in E(H_0)} U_e$.

We let T^1, T^2, T^3 be the subgraphs of T obtained from T by leaving out both, one or the other dashed edge(s) respectively. Note that for all $i \neq j \in [1, 3]$, it is $|E(T^i)| \neq |E(T^j)|$. We also let U' be the subgraph of U obtained from U by leaving out both dashed edges. Let $v \in V(H_0)$ and $e \in E(H_0)$. We let T_v^i be a fresh copy of T^i and U'_e a fresh copy of U'. Note that all three graphs T_v^1, T_v^2, T_v^3 are pairwise non-isomorphic and that now $U'_e \in \mathcal{G}_3$. T_v^1, T_v^2, T_v^3 U'_e

vertex-markers Let G be any graph. Let w be a vertex of G. We call w a *vertex-marker in* G if w is adjacent to exactly one vertex of degree 1 in G. We call w an *edge-marker in* G if w is adjacent to at least two vertices of degree 1 in G (see the circled vertices in Figures 3 and 4). Note that for every $v \in V(H_0)$ there is a unique $v^* \in T_v \subseteq V(H)$ such that v^* is a vertex-marker in H, and that for every $e \in E(H_0)$ there is a unique $e^* \in U_e \subseteq V(H)$ such that e^* is an edge-marker in H. edge-markers

Now the following claim is clear.

Claim 1. *Let* G *be a graph constructed from* H_0 *analogously as* H *is constructed from* H_0*, except that every vertex* $v \in V(H_0)$ *is replaced by one of* T_v^1, T_v^2, T_v^3 *and every edge* $e \in V(H_0)$ *is replaced by* U_e *(or* U'_e*). Then we have for all* $e \in E(H_0), u \neq v \in V(H_0)$,

$$N_5^G(v^*) \cong T_v^i \text{ for some } i \in [3], N_3^G(e^*) \cong U_e \text{ (or } N_4^G(e^*) \cong U'_e\text{), and}$$

this directly follows from Figures 3 and 4

$$\mathtt{dist}^G(u^*, e^*) = \mathtt{dist}^G(v^*, e^*) = 8 \text{ (or 9) if and only if } e \in \{uv, vu\}.$$

We define our FO interpretation $\Gamma := (\mathtt{true}, \psi(x,y))$ from $\{E\}$ to $\{E\}$ as follows. We set $\psi(x,y) := E(x,y) \vee \mu(x,y) \vee \eta(x,y)$, where $\mu(x,y), \eta(x,y) \in \mathrm{FO}[E]$ express the following (first order) conditions:

$\mu(x,y)$ ensures that a vertex v of H_0 is replaced by some T^i_v, that is, receives some colour $i \in [3]$

- $\mu(x,y)$ states that one of x, y, say x, is a vertex-marker such that the 5-neighbourhood of x is isomorphic to one of T^1, T^2, T^3 and that x, y are the ends of one of the dashed edges in Figure 3 .
- $\eta(x,y)$ states that one of x, y, say x, is an edge-marker, the 4-neighbourhood of x is isomorphic to U', y is at distance 2 from x and the following holds: There exist vertices z, z' at distance 9 from x such that z, z' are vertex-markers with their 5-neighbourhoods isomorphic to T^i and T^j with $i \neq j \in [3]$.

$\eta(x,y)$ ensures that an edge e of H_0 is replaced by U'_e, but only if both its endvertices have distinct colours

These conditions clearly can be expressed in first order logic.
Our next claim ensures that we can provide H as input to **A**.

Claim 2. It is $H \in \Gamma(\mathcal{G}_3)$.

construction of Γ-encoding $G \in \mathcal{G}_3$ of H from H_0

Proof. Since H_0 is 3-colourable, there is a 3-colouring $c : V(H_0) \longrightarrow [3]$ of H_0. We construct $G \in \mathcal{G}_3$ from H_0 analogously to the construction of H from H_0 with a slight modification: Every vertex $v \in V(H_0)$ is replaced by $T^{c(v)}_v \subseteq T_v$ and every edge $e \in E(H_0)$ is replaced by U'_e. Note that since $E(x,y) \models \psi(x,y)$, we have $G \subseteq \Gamma(G)$.
It suffices to show that $\Gamma(G) = H$. Since $V(G) = V(H)$, we only need to show $\psi(G) = E(H)$. Let $uv \in V(H)^2$. Then the following holds:

the case distinctions are analogous to the definition of $\psi(x,y) \in \mathrm{FO}[E]$

- If $G \models E(u,v)$, then $uv \in E(G)$ and since $G \subseteq H$, we have $uv \in E(H)$.
  If $uv \in E(H)$, then $G \models E(u,v)$ and so $uv \in \psi(G)$.
- If $G \models \mu(u,v)$, then one of u, v, say u, is a vertex-marker and the 5-neighbourhood of u is isomorphic to one of T^1, T^2, T^3 and that u, v are the ends of one of the dashed edges in Figure 3. By construction of H, it is $uv \in E(H)$.
  If there is a $w \in V(H_0)$ such that u and v are the ends of one of the dashed edges in $T^{c(w)}_w \subseteq G$, then one of u, v will be equal to $w^* \in V(G)$ by construction. By Claim 1 there is $i \in [3]$ with $N^G_5(w^*) \cong T^i$, so it follows that $G \models \mu(u,v)$.
- If $G \models \eta(u,v)$, then one of u, v, say u, is an edge-marker, the 4-neighbourhood of u is isomorphic to U' and v is at distance 2 from u. Then by construction of H, this implies $uv \in E(H)$.
  If there is an $e := ww' \in E(H_0)$ such that u and v are the ends of one of the dashed edges in $U'_e \subseteq G$, then by construction it is either $u = e^*$ or $v = e^*$, say $u = e^*$. Then $\mathtt{dist}^G(u,v) = 2$. Since c is a 3-colouring of H_0, we have $c(w) \neq c(w')$ and hence $T^{c(w)}_w \not\cong T^{c(w')}_{w'}$. By Claim 1, we also have $\mathtt{dist}^G(e^*, w^*) = \mathtt{dist}^G(e^*, w'^*) = 9$ and $N^G_4(e^*) \cong U'$. And so we have $G \models \eta(u,v)$.

This implies $\psi(G) = E(H)$ and thus $\Gamma(G) = H$. □

3-colouring c^G

Now we can feed $H \in \Gamma(\mathcal{G}_3)$ as input to **A** to obtain a graph $G \in \mathcal{G}_3$ with $\Gamma(G) = H$. Recall that $V(T_v) \subseteq V(H) = V(G)$ for all $v \in V(H_0)$. Define $c^G : V(H_0) \longrightarrow [3]$ as follows: For all $v \in V(H_0)$, let $c^G(v) := i$ for the unique $i \in [3]$ with $G[V(T_v)] \cong T^i$.

to compute $c^G(v)$, just count the number of edges in $G[V(T_v)]$ to determine its isomorphism type amongst T^1, T^2, T^3

Claim 3. *c^G is a 3-colouring of H_0.*

Before we prove this claim, we want to prove the following claim first.

Claim 4. *The vertex-markers in G are exactly the vertex-markers in H and the edge-markers in G are exactly the edge-markers in H up to automorphisms of G.*

Proof. Since the vertex- and edge-markers are completely determined by vertices of degree 1 in G or H respectively, it clearly suffices to show that the vertices of degree 1 in G are the vertices of degree 1 in H (up to automorphisms of G). Let u be a vertex of degree 1 in G. Since $G \subseteq H$, it is $\mathtt{deg}^H(u) \geqslant 1$. In order to show that $\mathtt{deg}^H(u) = 1$, assume there is $v \in V(G)$ such that one of uv, vu, say uv, is in $\psi(G) \setminus E(G)$. Then at least one of $\mu(u, v), \eta(u, v)$ holds in G.

- Assume $G \models \mu(u, v)$. Then there is $i \in [3]$ such that it is either $N_5^G(u) \cong T^i$ or $N_5^G(v) \cong T^i$. Since $\mathtt{deg}^G(u) = 1$ it must be $N_5^G(v) \cong T_i$. But then the only vertex of degree 1 in $N_5^G(v) \cong T_i$ is adjacent to v (see Figure 3), but this contradicts $uv \notin E(G)$.

- Assume $G \models \eta(u, v)$, then there is a vertex $e \in V(G)$ such that $N_4^G(e) \cong U'$ and ($e = u, \mathtt{dist}^G(u, v) = 2$ or $e = v, \mathtt{dist}^G(v, u) = 2$). It cannot be $e = u$ because $\mathtt{deg}^{N_4^G(e)}(e) \neq 1$ (see Figure 4) and it cannot be $e = v$ because all vertices at distance 2 from e in $N_4^G(e) \cong U'$ (see Figure 4) have degree greater than 1 in G, so this leads to a contradiction as well.

Let u be a vertex of degree 1 in H. Then there must be a vertex $v \in V(H_0)$ or an edge $e \in E(H_0)$ such that u is the vertex in T_v on the left in Figure 3 or one of the vertices in U_e at the top in Figure 4.

- If $u \in T_v$, consider the vertex-marker v^* in H. Since $\mathtt{deg}^H(v^*) = 4$ and $\mathtt{deg}^G(v^*) \leqslant 3$, there must be a vertex $t \in V(G)$ with $v^*t \in \psi(G) \setminus E(G)$. Then at least one of $\mu(v^*, t), \eta(v^*, t)$ must be true in G. Obviously v^* is not an edge-marker. It is $t \in N^H(v^*) \subseteq T_v$ and since there is only one vertex of degree 1 in T_v (see Figure 3), t cannot be an edge-marker. So it is $G \models \mu(v^*, t)$. Since t is not a vertex-marker either, we get that $N_5^G(v^*) \cong T^i$ for some $i \in [3]$. This means that there is exactly one vertex of degree 1 in $N_5^G(v^*)$ and this vertex is equal to u up to automorphisms of G. Without loss of generality we assume that this vertex is actually equal to u and so we have that $\mathtt{deg}^G(u) = 1$.

the "degree argument" relies on the fact that in both T_v, U_e, there are vertices of degree greater than 3, so μ or η must hold respectively

- If $u \in U_e$, consider the edge-marker e^* in H. By the analogous degree argument employed above we get that there must exist $t \in V(G)$ such that at least one of $\mu(e^*, t), \eta(e^*, t)$ is true in G. By analogous arguments we get that $G \models \eta(e^*, t)$ which, as above, implies $N_4^G(e^*) \cong U'_e$. So there are exactly two vertices in $N_4^G(e^*) \cong U'_e$ of degree 1 in G. As above, we neglect a possible difference via automorphisms of G and pick one of those vertices to be u in some (fixed) way.

□

Now we prove our main claim.

first we show that c^G is a well-defined mapping

Proof of Claim 3. Let $v \in V(H_0)$. Since for all $i \neq j \in [3]$ it is $T_v^i \not\cong T_v^j$, there is at most one $i \in [3]$ with $c^G(v) = i$. Consider the vertex-marker $v^* \in V(T_v)$ in H. Since $\deg^H(v^*) = 4 > 3$ and $\Delta(G) \leqslant 3$, there must be a vertex $t \in V(G)$ with $v^*t \in \psi(G) \setminus E(G)$, so $v^*t \in E(H) \setminus E(G)$. Thus it is $G \models \mu(v^*, t) \vee \eta(v^*, t)$. By Claim 4, v^* is a vertex-marker in G and thus no edge-marker in G. We also have $t \in N^H(v^*) \subseteq T_v$ and since there are no edge-markers in T_v, t is not an edge-marker in H and by Claim 4 not an edge-marker in G either. Hence, $G \models \mu(v^*, t)$. Since t is no vertex-marker in G, there must be $i \in [3]$ with $G[V(T_v)] = N_5^G(v^*) \cong T_v^i$, so $c^G(v) = i$. Hence, c^G is a well-defined mapping.

now we show that $c^G(v) \neq c^G(u)$ for $uv \in E(H_0)$

Let $u \in V(H_0)$ be any neighbour of v in H_0. Let $e := uv \in E(H_0)$. Consider the edge-marker $e^* \in V(U_e)$ in H which, by Claim 4, is an edge-marker in G as well. By an analogous degree argument we get that there must exist $t \in V(G)$ such that $e^*t \in \psi(G) \setminus E(G)$, so $e^*t \in E(H) \setminus E(G)$. Then it is $G \models \mu(e^*, t) \vee \eta(e^*, t)$. We have $t \in N^H(e^*) \subseteq U_e$ which is why t cannot be a vertex-marker in H and by Claim 4 not a vertex-marker in G either. Since e^* is an edge-marker in G, it is not a vertex-marker in G. Hence we get $G \models \eta(e^*, t)$. This means that there are vertex-markers $w, w' \in V(G)$ such that $\mathtt{dist}^G(e^*, w) = \mathtt{dist}^G(e^*, w') = 9$ and there are $i \neq j \in [3]$ with $N_5^G(w) \cong T^i, N_5^G(w') \cong T^j$. Since $G \subseteq H$, we have $\mathtt{dist}^H(e^*, w) \leqslant 9, \mathtt{dist}^H(e^*, w') \leqslant 9$. By construction it is $\mathtt{dist}^H(e^*, w) = \mathtt{dist}^H(e^*, w') = 8$ and thus by Claim 1 it is $\{u^*, v^*\} = \{w, w'\}$, say $u^* = w, v^* = w'$. Then we have $c^G(u) = i \neq j = c^G(v)$. □

those distances are equal to 8 because the two dashed edges in U_e are added in H, which are symmetric to each other

summary: what have we done?

To sum it up, we started out with a 4-regular, 3-colourable graph H_0, and wanted to compute one of its 3-colourings. We used a hypothetical algorithm **A** that computes $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$ for a specific FO interpretation $\Gamma = (\mathtt{true}, \psi)$ from $\{E\}$ to $\{E\}$. We transformed the graph H_0 in polynomial time into another graph H which served as input to the problem $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$. Then we fed H to **A** which gave us a graph G from which we could extract in polynomial time a 3-colouring of H_0. Thus, if **A** runs in polynomial time, then our entire procedure runs in polynomial time. Since the problem $\mathtt{m\text{-}c3COL}(\mathcal{R})$ of colouring a 4-regular, 3-colourable graph is $\mathtt{NP}$-hard, as shown in Lemma 4.1, this cannot happen (unless $\mathtt{P} = \mathtt{NP}$). So, $\mathtt{cENC}(\Gamma, \mathcal{G}_3)$ is $\mathtt{NP}$-hard as well. □

FO INTERPRETABILITY IN CLASSES OF BOUNDED DEGREE

In this section we will introduce a modified version of the notion of *near-uniformity* as presented by Gajarský et al. Our version naturally extends their notion from graphs to arbitrary structures. Then we will show that the classes of vertex- and edge-coloured directed graphs FO interpretable in classes of bounded degree are actually near-uniform.

The Notion of Near-Uniformity

We begin by defining a useful concept that generalizes the concept of a neighbour in a graph to arbitrary structures (in a different way than the Gaifman graph does).

Definition 5.1. *Let* R *be some predicate of arity* r *and* $\mathfrak{B}$ *any* $\{R\}$*-structure. Let* $v \in B$ *and* $i \in [r]$*. Then we define* $\mathbf{N}_i^{\mathfrak{B}}(v)$

$$\mathbf{N}_i^{\mathfrak{B}}(v) := \{v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_r \mid (v_1, \ldots, v_{i-1}, v, v_{i+1}, \ldots, v_r) \in R(\mathfrak{B})\}.$$

This example shows that the new definition is consistent with traditional concepts.

Example 5.2. *Let* $\mathfrak{B}$ *be a* $\{R\}$*-structure and* $a \in B$*. If* $\mathfrak{B}$ *is a coloured set, it is* $\mathbf{N}_1^{\mathfrak{B}}(a) = \emptyset$*. If* $\mathfrak{B}$ *is a directed graph, then usually* $\mathbf{N}_1^{\mathfrak{B}}(a) \neq \mathbf{N}_2^{\mathfrak{B}}(a)$ *and we denote* $\mathbf{N}_+^{\mathfrak{B}}(a) := \mathbf{N}_1^{\mathfrak{B}}(a)$ *and* $\mathbf{N}_-^{\mathfrak{B}}(a) := \mathbf{N}_2^{\mathfrak{B}}(a)$*. If* $\mathfrak{B}$ *is a graph, it is* $N^{\mathfrak{B}}(a) = \mathbf{N}_+^{\mathfrak{B}}(a) = \mathbf{N}_-^{\mathfrak{B}}(a)$*.*

We introduce the near-k-twin relation for arbitrary structures, which is the heart of the notion of near-uniformity, via two successive definitions.

Definition 5.3. *Let* R *be a predicate of arity* r *and* $\mathfrak{B}$ *any* $\{R\}$*-structure. Let* $k \in \mathbb{N}$*. Then the* near-k-twin relation $\rho_k^{\mathfrak{B}}$ of $\mathfrak{B}$ *is defined as follows. Let* $u, v \in B$*. Then we shall let* $(u, v) \in \rho_k^{\mathfrak{B}}$ *if and only if* *near-k-twin relation*

$$\left| \bigcup_{i \in [k]} \mathbf{N}_i^{\mathfrak{B}}(u) \triangle \mathbf{N}_i^{\mathfrak{B}}(v) \right| \leqslant k.$$

We also denote $\Delta^{\mathfrak{B}}(u, v) := \bigcup_{i \in [k]} \mathbf{N}_i^{\mathfrak{B}}(u) \triangle \mathbf{N}_i^{\mathfrak{B}}(v)$*. If the near-k-twin relation of* $\mathfrak{B}$ *is an equivalence relation on* B*, it is also called the* near-k-twin equivalence of $\mathfrak{B}$. $\Delta^{\mathfrak{B}}(u, v)$

Recall that for a σ-structure $\mathfrak{B}$ and $R \in \sigma$, we denote by $\mathfrak{B}^R$ the $\{R\}$-reduct of $\mathfrak{B}$.

Definition 5.4. *Let* $\mathfrak{B}$ *be any* σ*-structure. Let* $\bar{k} \in \mathbb{N}^\sigma, p \in \mathbb{N}$*. Then* $\mathfrak{B}$ *is* $(\bar{k}, p)$-near-uniform *if for all* $R \in \sigma$*, the near-*k_R*-twin relation* $\rho_{k_R}^{\mathfrak{B}^R}$ *of* $\mathfrak{B}^R$ *is an equivalence on* B *with index at most* p*.* *$(\bar{k}, p)$-near-uniform*

Example 5.5.

- *Any set is* $((), 1)$*-near-uniform.*
- *Let* $\mathfrak{B}$ *be a* σ*-structure and* $\bar{k} \in \mathbb{N}^\sigma$*. Then* $\mathfrak{B}$ *is* $(\bar{k}, 0)$*-near-uniform if and only if* $B = \emptyset$*.*
- *Consider the directed graph* H *in Figure 5: Then* H *is* $(0, 4)$*-near-uniform*

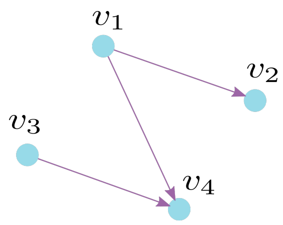

Figure 5: A (k, p)-near-uniform directed graph for $(k, p) \in \{(0, k), (1, 2), (k, 1) \mid k \geqslant 4\}$ which is not (k', p')-near-uniform for $(k', p') \in \{(2, p'), (3, p') \mid p' \in \mathbb{N}\}$

with near-0-twin classes $(\{v_i\})_{i \in [4]}$, $(1, 2)$-near-uniform with near-1-twin classes $\{v_1, v_3\}, \{v_2, v_4\}$ and for all $k \geqslant 4$, $(k, 1)$-near-uniform with the only near-k-twin class $V(H)$. But ρ_2^H and ρ_3^H are not transitive because $v_1v_3 \in \rho_2^H, v_3v_2 \in \rho_2^H, v_1v_2 \notin \rho_2^H$ and $v_1v_3 \in \rho_3^H, v_3v_4 \in \rho_3^H, v_1v_4 \notin \rho_3^H$ respectively, which is why for all $p' \in \mathbb{N}$, H is neither $(2, p')$- nor $(3, p')$-near-uniform.

Now we are ready to say what a near-uniform σ-class should be.

(k_0, p)-near-uniform

Definition 5.6. *Let $\mathcal{D}$ be any σ-class. Let $k_0, p \in \mathbb{N}$. Then $\mathcal{D}$ is* (k_0, p)-near-uniform *if for all $\mathfrak{B} \in \mathcal{D}$ there is a $\bar{k} \in [0, k_0]^\sigma$ such that $\mathfrak{B}$ is $(\bar{k}, p)$-near-uniform. Furthermore, $\mathcal{D}$ is* near-uniform *if there are $k_0 \in \mathbb{N}, p \in \mathbb{N}_+$ such that $\mathcal{D}$ is (k_0, p)-near-uniform.*

Note that every subclass of a (k, p)-near-uniform class is again (k, p)-near-uniform by definition.

Near-Uniformity and FO Interpretations

Now we can state our goal for this section:

Theorem 5.7. *Let σ be a signature that only contains predicates of arity at most 2, and τ be any arbitrary signature. Let $\mathcal{C}$ be a τ-class of bounded degree and Γ an* FO *interpretation from τ to σ. Then the class $\Gamma(\mathcal{C})$ of vertex- and edge-coloured directed graphs is near-uniform.*

Note that from this theorem, it directly follows that every subclass of $\Gamma(\mathcal{C})$ is near-uniform as well.
We want to show that a certain class $\Gamma(\mathcal{C})$ of vertex- and edge-coloured directed graphs is near-uniform. A vertex- and edge-coloured directed graph will have an arbitrary number of relations, but its near-uniformity is, in essence, a property of each of its R-reducts for $R \in \sigma$. So it makes sense to decompose the given vertex- and edge-coloured directed graph into unlabeled directed graphs and 1-labeled sets, and to show near-uniformity for each of those structures individually. The next lemma allows us to do exactly that. Recall that for a σ-class $\mathcal{D}$ and a predicate $R \in \sigma$, we denote by $\mathcal{D}^R$ the class of $\{R\}$-reducts of σ-structures in $\mathcal{D}$.

Lemma 5.8. *Let $\mathcal{D}$ be any σ-class. Then $\mathcal{D}$ is near-uniform if and only if for all $R \in \sigma$, $\mathcal{D}^R$ is near-uniform.*

Proof. The first direction follows easily from Definition 5.4.
For the other direction, assume that for all $R \in \sigma$, $\mathcal{D}^R$ is near-uniform. Then for all $R \in \sigma$, there are $k_R, p_R \in \mathbb{N}$ such that $\mathcal{D}^R$ is (k_R, p_R)-near-uniform. Let $p := \max_{R \in \sigma} p_R$ and $k_0 := \max_{R \in \sigma} k_R$ (or $p := 1$ and $k_0 := 0$ if $\sigma = \emptyset$). Let $\mathfrak{B} \in \mathcal{D}$ be a σ-structure. Then for all $R \in \sigma$, there is a

$k \leqslant k_R \leqslant k_0$ such that $\mathfrak{B}^R$ is (k, p_R)-near-uniform and, since $p_R \leqslant p$, also (k, p)-near-uniform. Hence, by Definition 5.4, $\mathfrak{B}$ is $(\bar{k}, p)$-near-uniform for some $\bar{k} \in [0, k_0]^\sigma$. So by Definition 5.6, $\mathfrak{D}$ is (k_0, p)-near-uniform. $\square$

Considering the class $\Gamma(\mathcal{C})$ again, we see that the FO interpretation Γ involved is arbitrary. We want to reduce our problem to the case where Γ is simple.

Lemma 5.9. *Let* R *be a predicate of arity* r *and* $\mathcal{C}$ *any* $\{R\}$*-class. Let* $k_0, p \in \mathbb{N}$. *Let* $\Gamma := (\psi_{univ}, \psi_R) \in \mathit{INT}_\tau^\sigma$ *and set* $\Gamma' := \left(\mathit{true}, \psi_R(\bar{x}) \wedge \bigwedge_{i \in [r]} \psi_{univ}(x_i)\right) \in \mathit{INT}_\tau^\sigma$. *Then* $\Gamma(\mathcal{C})$ *is* (k_0, p)*-near-uniform if and only if* $\Gamma'(\mathcal{C})$ *is* $(k_0, p+1)$*-near-uniform.*

Proof. Let $\mathfrak{A} \in \mathcal{C}$. It will suffice to show that for all $k \in [0, k_0]$, $\Gamma(\mathfrak{A})$ is (k, p)-near-uniform if and only if $\Gamma'(\mathfrak{A})$ is $(k, p+1)$-near-uniform. Denote $\mathfrak{B} := \Gamma(\mathfrak{A}), \mathfrak{B}' := \Gamma'(\mathfrak{A})$. Note that $B \subseteq B' = A$. Also observe that

$$R(\Gamma(\mathfrak{A})) \overset{\text{by Def. 3.1}}{=} \psi_R(\mathfrak{A}) \cap \psi_{univ}(\mathfrak{A})^r = \left(\psi_R(\bar{x}) \wedge \bigwedge_{i \in [r]} \psi_{univ}(x_i)\right)(\mathfrak{A}) = R(\Gamma'(\mathfrak{A})),$$

so $R(\mathfrak{B}) = R(\mathfrak{B}')$. Hence, for all $b \in B, i \in [r]$, it is $\mathbf{N}_i^{\mathfrak{B}}(b) = \mathbf{N}_i^{\mathfrak{B}'}(b)$. Thus, for all $b_1, b_2 \in B \subseteq B'$, we have $\Delta^{\mathfrak{B}}(b_1, b_2) = \Delta^{\mathfrak{B}'}(b_1, b_2)$ and therefore

$$(b_1, b_2) \in \rho_k^{\mathfrak{B}} \iff (b_1, b_2) \in \rho_k^{\mathfrak{B}'}.$$

So it is $\rho_k^{\mathfrak{B}'}[B] = \rho_k^{\mathfrak{B}}[B] = \rho_k^{\mathfrak{B}}$ and thus $\mathtt{index}(\rho_k^{\mathfrak{B}'}[B]) = \mathtt{index}(\rho_k^{\mathfrak{B}})$. To show the claim, now it suffices to show that $\mathtt{index}\left(\rho_k^{\mathfrak{B}'}[A \setminus B]\right) \leqslant 1$. But since

$$R(\mathfrak{B}') \cap (A \setminus B)^r = R(\mathfrak{B}) \cap (A \setminus B)^r \overset{R(\mathfrak{B}) \subseteq B^r}{=} \emptyset,$$

all pairs of vertices in $A \setminus B$ are near-0-twins in $\mathfrak{B}'$, and therefore, this is clearly the case. $\square$

Putting these two lemmas together, all that separates us from Theorem 5.7 is the following claim.

Theorem 5.10. *Let* τ *be a signature and* R *a predicate of arity at most* 2. *Let* $\mathcal{C}$ *be a* τ*-class of bounded degree and* Γ *a simple* FO *interpretation from* τ *to* $\{R\}$. *Then the* $\{R\}$*-class* $\Gamma(\mathcal{C})$ *is near-uniform.*

We briefly explain how Theorem 5.7 follows from Theorem 5.10 and Lemmas 5.8 and 5.9.

Proof of Theorem 5.7. Say it is $\Gamma = (\psi_{univ}, (\psi_R)_{R \in \sigma})$. Let

$$\Gamma' := \left(\mathtt{true}, (\psi_R(\bar{x}) \wedge \bigwedge_{i \in [r]} \psi_{univ}(x_i))_{R \in \sigma}\right) \in \mathtt{INT}_\tau^\sigma.$$

By Lemma 5.8, it suffices to show that for all $R \in \sigma$, $\Gamma(\mathcal{C})^R$ is near-uniform. Let $R \in \sigma$. Since Γ'^R is simple, by Theorem 5.10, we get that $\Gamma'^R(\mathcal{C})$ is near-uniform, but then, by Lemma 5.9, $\Gamma^R(\mathcal{C})$ is near-uniform. $\square$

Locality of FO and Classes of Bounded Degree

In order to prove Theorem 5.10, we have to prepare more. At first we need Gaifman's Locality Theorem and in order to state it, we define r-local formulae.

r-local formulae

Definition 5.11. *Let $r, k \in \mathbb{N}$. A first order formula $\varphi(x_1, \ldots, x_k) \in \mathrm{FO}[\tau]$ is called* r-local *if for all τ-structures $\mathfrak{A}$ and all vertices $\bar{v} \in A^k$*

$$\mathfrak{A} \models \varphi(\bar{v}) \iff \bigcup_{1 \leqslant i \leqslant k} N_r^{\mathfrak{A}}(v_i) \models \varphi(\bar{v}).$$

Then φ is also denoted $\varphi^{(r)}$.

Let $r \in \mathbb{N}$. Note that the r-local τ-sentences are exactly the tautologies and contradictions (because we allow empty structures). An r-local formula can be understood as a formula whose quantifiers only range over the r-neighbourhoods of the assigned vertices. So, whether or not a r-local formulae holds for a certain tuple of vertices in some structure $\mathfrak{A}$ is only dependent on a small part of $\mathfrak{A}$, namely the r-neighbourhoods of the entries of the tuple in $\mathfrak{A}$. Note that an r-local formula is also r'-local for all $r' \geqslant r$. Before we state Gaifman's Locality Theorem, we define for all $c \in \mathbb{N}$ a τ-formula $\mathtt{dist}(x, y) > c$ capturing all pairs (v, w) of vertices such that the distance from v to w in a given structure is greater than c. This will be defined as follows:

$$\mathtt{dist}(x, y) > c := \neg \exists x_0 \ldots \exists x_c (x = x_0 \wedge y = x_c$$
$$\wedge \bigwedge_{0 \leqslant i < c} (E x_i x_{i+1} \vee x_i = x_{i+1})).$$

Note that for all $\mathfrak{A} \in \mathtt{Str}(\tau), v \in A, r \in \mathbb{N}$, the extension of $\neg\mathtt{dist}(x, y) > r$ in $\mathfrak{A}$ under the variable assignment $x \mapsto v$ is equal to $N_r^{\mathfrak{A}}(v)$. A *basic local τ-sentence* is a sentence $\theta \in \mathrm{FO}[\tau]$ of the form

$$\theta = \exists x_1 \ldots \exists x_k \left(\bigwedge_{1 \leqslant i < j \leqslant k} \mathtt{dist}(x_i, x_j) > 2r \wedge \bigwedge_{1 \leqslant i \leqslant k} \varphi^{(r)}(x_i) \right)$$

for some $k, r \in \mathbb{N}$ and r-local $\varphi^{(r)} \in \mathrm{FO}_1[\tau]$.
Now, we restate Gaifman's Locality Theorem (see [4] for a proof):

Gaifman's Locality

Theorem 5.12 (Gaifman's Locality Theorem)**.** *Let $\varphi(\bar{x}) \in \mathrm{FO}[\tau]$ and set $q := qr(\varphi)$. Then there is a nonnegative integer $r \leqslant 7^{q-1}$ such that φ is equivalent to a Boolean combination of*

- *r-local formulae around $\bar{x}$, and*
- *basic local τ-sentences.*

Before we can state the next lemma, we need the concept of types.

Definition 5.13. *Let $q, k \in \mathbb{N}$. Let $\mathfrak{A}$ be a τ-structure and $\bar{a} \in A^k$. Set $r := \lfloor 7^{q-1} \rfloor$. Then the* (local) q-type of $\bar{a}$ in $\mathfrak{A}$ *is defined as*

$$tp_q^{\mathfrak{A}}(\bar{a}) := \left\{ \varphi^{(r)} \in \mathrm{FO}[\tau] \mid qr(\varphi) \leqslant q, \mathfrak{A} \models \varphi(\bar{a}) \right\}.$$

Now we prove the following lemma.

Lemma 5.14. *Let* $q \in \mathbb{N}$ *and set* $r := \lfloor 7^{q-1} \rfloor$*. Let* $\mathfrak{A}$ *be a* τ*-structure and* $v, w \in A$*. Then we have*

$$N_r^{\mathfrak{A}}(v) \cong N_r^{\mathfrak{A}}(w) \iff tp_q^{\mathfrak{A}}(v) = tp_q^{\mathfrak{A}}(w).$$

Proof. Assume $N_r^{\mathfrak{A}}(v) \cong N_r^{\mathfrak{A}}(w)$. Let $\varphi(x) \in \mathsf{FO}[\tau]$ with $\mathsf{qr}(\varphi) \leqslant q$. Then by Gaifman's Locality Theorem 5.12, we get that φ is equivalent to some Boolean combination $\xi(x)$ of r-local τ-formulae around x and basic local τ-sentences. Let θ be one of the basic local τ-sentences. Since $\mathtt{free}(\theta) = \emptyset$, we have $\mathfrak{A} \models \theta$ if and only if $\mathfrak{A} \models \theta$, so $\theta \in \mathtt{tp}_q^{\mathfrak{A}}(v)$ if and only if $\theta \in \mathtt{tp}_q^{\mathfrak{A}}(w)$.
Let $\theta^{(r)}$ be one of the r-local τ-formulae around x appearing in ξ. Since $N_r^{\mathfrak{A}}(v) \cong N_r^{\mathfrak{A}}(w)$, we get by the isomorphism invariance of first order logic that $\theta \in \mathtt{tp}_q^{\mathfrak{A}}(v)$ if and only if $\theta \in \mathtt{tp}_q^{\mathfrak{A}}(w)$. So any r-local τ-formula and any basic local τ-sentence are either in both or in neither of the types and this obviously extends to their Boolean combination $\xi(x) \equiv \varphi(x)$.
We only sketch the proof for the other direction: If $N_r^{\mathfrak{A}}(v) \not\cong N_r^{\mathfrak{A}}(w)$, then we can construct a τ-formula $\varphi(x)$ such that $\mathsf{qr}(\varphi) \leqslant q$ and $\mathfrak{A} \models \varphi(v), \mathfrak{A} \not\models \varphi(w)$. This can be done mainly because both r-neighbourhoods are finite. □

Definition 5.15. *Let* $\mathfrak{A}$ *be a* τ*-structure and* $q \in \mathbb{N}$*. We define the equivalence relation* $\equiv_q^{\mathfrak{A}}$ *on* A *as follows. For all* $v, w \in A$ *we set* $\equiv_q^{\mathfrak{A}}$

$$v \equiv_q^{\mathfrak{A}} w \;:\Longleftrightarrow\; tp_q^{\mathfrak{A}}(v) = tp_q^{\mathfrak{A}}(w).$$

Intuitively, two vertices v and w of a τ-structure $\mathfrak{A}$ are in the same $\equiv_q^{\mathfrak{A}}$-equivalence class if no τ-formula of quantifier rank at most q can distinguish them.
The next corollary follows essentially from Lemma 5.14 and will prove useful when dealing with classes of bounded degree.

Corollary 5.16. *Let* $\mathcal{C}$ *be a* τ*-class of bounded degree and* $q \in \mathbb{N}$*. Then there is* $s_0 \in \mathbb{N}_+$ *such that for all* $\mathfrak{A} \in \mathcal{C}$*, we have* $\mathtt{index}(\equiv_q^{\mathfrak{A}}) \leqslant s_0$*.*

Proof. Since $\mathcal{C}$ is a class of bounded degree, there is $d \in \mathbb{N}$ such that every τ-structure in $\mathcal{C}$ has maximum degree at most d. We will choose s_0 later in the proof, depending only on d, q, τ.
We want to show that for all $\mathfrak{A} \in \mathcal{C}$, there are at most s_0 vertices of different local q-types in $\mathfrak{A}$.
Let $\mathfrak{A} \in \mathcal{C}$. By Lemma 5.14, there are at most as many vertices of different local q-types in $\mathfrak{A}$, as there are vertices with different r-neighbourhoods in $\mathfrak{A}$. Since $\mathfrak{A}$ has maximum degree at most d, we have for all $v \in A$,

$$\left| N_r^{\mathfrak{A}}(v) \right| \leqslant 1 + (d-1)d^{r-1} \leqslant 1 + d^r.$$

So, for all $v \in A$, the r-neighbourhood of v in $\mathfrak{A}$ has order bounded by a function of d and r, so of d and q. Now, we could simply set s_0 to be the number of isomorphism classes of τ-structures of order at most $1 + d^r$, and we would be done. But we want to give an explicit bound s_0 for the number of isomorphism classes of τ-structures of order most $1 + d^r$. Let $n := 1 + d^r$ and $r_0 := \max_{R \in \tau} \mathsf{ar}(R)$. By simple combinatorial arguments, we get that

$$s_0 := (n+1) \left(2^{n^{r_0}} \right)^{|\tau|}$$

is the number of τ-structures on a universe of the form $[1, m]$ for $m \in [0, n]$, and thus an upper bound for the number of isomorphism classes of τ-structures of order at most n. $\square$

We only need the following corollary before we are equipped to prove Theorem 5.10. It is a slight generalization of Corollary 4.2 in [1].

Corollary 5.17. *Let $k \in \mathbb{N}$ and $\psi(\bar{x}) \in \mathrm{FO}_k[\tau]$. Let $q := qr(\psi)$ and $r := \lfloor 7^{q-1} \rfloor$. Then for all τ-structures $\mathfrak{A}$, $(\bar{a}, b) \in V(\mathfrak{A})^{k+1}, i \in [k]$ with*

- *$tp_q^{\mathfrak{A}}(a_i) = tp_q^{\mathfrak{A}}(b)$, and*
- *$dist^{\mathfrak{A}}(a_j, a_i) > 2r$ and $\mathrm{dist}^{\mathfrak{A}}(a_j, b) > 2r$ for all $j \in [k] \setminus \{i\}$,*

it holds that

$$\mathfrak{A} \models \psi(\bar{a}) \iff \mathfrak{A} \models \psi(\bar{a}[i/b]).$$

This means that an FO formula ψ only uses the $\mathrm{qr}(\psi)$-type of the vertices of the structure on which it is evaluated. So if two vertices have the same $\mathrm{qr}(\psi)$-type, then the only way of telling the difference between them is by their (physical) distances to another fixed vertex. But if they are also far enough apart from all the fixed vertices, then there is no way of telling the difference between them at all.

Proof. Let $\mathfrak{A}$ be any τ-structure and $(\bar{a}, b) \in V(\mathfrak{A})^{k+1}, i \in [k]$ with

$$\mathrm{tp}_q^{\mathfrak{A}}(a_i) = \mathrm{tp}_q^{\mathfrak{A}}(b), \tag{1}$$

$$\mathrm{dist}^{\mathfrak{A}}(a_j, a_i) > 2r \text{ and } \mathrm{dist}^{\mathfrak{A}}(a_j, b) > 2r \text{ for all } j \in [k] \setminus \{i\}. \tag{2}$$

By Lemma 5.14, 1 implies $N_r^{\mathfrak{A}}(a_i) \cong N_r^{\mathfrak{A}}(b)$. So there is an isomorphism $\pi : N_r^{\mathfrak{A}}(a_i) \cong N_r^{\mathfrak{A}}(b)$. By Gaifman's Locality Theorem 5.12, ψ is equivalent to a Boolean combination of basic local τ-sentences and r-local τ-formulae around $\bar{x}$. The claim obviously holds for all τ-sentences and so for all basic local τ-sentences. Let ψ' be any r-local τ-formula around $\bar{x}$ such that $\mathfrak{A} \models \psi'(\bar{a})$. Then by Definition 5.11, it is $N_r^{\mathfrak{A}}(\bar{a}) \models \psi'(\bar{a})$. By 2, we have for all $j \in [k] \setminus \{i\}$,

$$N_r^{\mathfrak{A}}(a_j) \cap N_r^{\mathfrak{A}}(a_i) = \emptyset = N_r^{\mathfrak{A}}(a_j) \cap N_r^{\mathfrak{A}}(b).$$

That is why the extension $\pi' : N_r^{\mathfrak{A}}(\bar{a}) \longrightarrow N_r^{\mathfrak{A}}(\bar{a})$ of π that maps every vertex outside of $N_r^{\mathfrak{A}}(a_i)$ onto itself is an isomorphism with $\pi'(a_i) = b$. Thus we get $\pi'(N_r^{\mathfrak{A}}(\bar{a})) \models \psi'(\pi'(\bar{a}))$, so $N_r^{\mathfrak{A}}(a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_k) \dot\cup N_r^{\mathfrak{A}}(b) \models \psi'(\bar{a}[i/b])$ and since ψ' is r-local, we also get $\mathfrak{A} \models \psi'(\bar{a}[i/b])$. The other direction follows analogously. $\square$

$\Gamma(\mathcal{C})$ is Near-Uniform

We repeat our goal once more and then finally prove it.

Theorem 5.10. *Let τ be a signature and R a predicate of arity at most 2. Let $\mathcal{C}$ be a τ-class of bounded degree and Γ a simple FO interpretation from τ to $\{R\}$. Then the $\{R\}$-class $\Gamma(\mathcal{C})$ is near-uniform.*

Proof. Let us briefly consider the case that the arity of R is 1. Then $\Gamma(\mathcal{C})$ is a class of coloured sets and for all $\{R\}$-structures $\mathfrak{B} \in \Gamma(\mathcal{C})$ and all $v \in B$, we have

$$\mathbf{N}_1^{\mathfrak{B}}(v) = \emptyset.$$

This implies that $\Gamma(\mathcal{C})$ is (k_0, p)-near-uniform with parameters $k_0 := 0, p := 1$.
So let the arity of R be 2 and set $E := R$. Let $\mathcal{D} := \Gamma(\mathcal{C})$. Let d be a positive integer such that every τ-structure in $\mathcal{C}$ has maximum degree at most d. Say, $\Gamma = (\mathtt{true}, \psi(x, y))$ with $\psi(x, y) \in FO[\tau]$.
Set $q := qr(\psi)$. By Corollary 5.16 there is $s_0 \in \mathbb{N}$ such that the equivalence relation $\equiv_q^{\mathfrak{A}}$ has index at most s_0 for all $\mathfrak{A} \in \mathcal{C}$. Let $r := \lfloor 7^{q-1} \rfloor$ and $c := 2r$. We now give the parameters $k_0 \in \mathbb{N}, p \in \mathbb{N}_+$ such that $\mathcal{D}$ is (k_0, p)-near-uniform. We set

$$p := s_0,$$

$$k_0 := 3^{s_0+3}(1 + d^c).$$

choice of k_0, p

We observe that c only depends on ψ and s_0 only depends on d, τ, ψ. Hence k_0, p only depend on d, τ, ψ.
Let $\mathfrak{A} \in \mathcal{C}$ and $G := \Gamma(\mathfrak{A}) \in \mathcal{D}$. Note that $A = V(G)$ and set $V := A = V(G)$. We need to show that there is a $k \leqslant k_0$ such that G is (k, p)-near-uniform. Let $s \leqslant s_0$ be the index of $\equiv_q^{\mathfrak{A}}$ and $U_1, \ldots, U_s$ the equivalence classes w.r.t. $\equiv_q^{\mathfrak{A}}$, listed in ascending order according to their sizes. Note that since $\mathfrak{A}$ has maximum degree at most d, we have for all vertices $v \in A$, $\left|N_c^{\mathfrak{A}}(v)\right| \leqslant 1 + d^c$ as we have already seen in the proof of Lemma 5.14. Hence for all vertices $v, w \in A$ we also have $\left|N_c^{\mathfrak{A}}(v, w)\right| \leqslant 2(1 + d^c)$.
Define $t_i := |U_i|$ for all $i \in [1, s]$. Now, we want to choose our parameter $k \leqslant k_0$ such that $\mathfrak{A}$ is (k, p)-near-uniform. For all $j \in [0, s]$, we let $\mathtt{BND}_j$ abbreviate the claim that for all $j' \in [1, j]$, $t_{j'} \leqslant \sum_{i \in [j'-1]} t_i + 6(1 + d^c)$. $\mathtt{BND}_j$
Note that for all $j \in [s]$, $\mathtt{BND}_j$ implies $\mathtt{BND}_{j'}$ for all $1 \leqslant j' \leqslant j$.

Claim 1. Let $j \in [0, s]$ such that $\mathtt{BND}_j$ holds. Then

$$t_1 + \ldots + t_j \leqslant 3^{s_0+2}(1 + d^c).$$

Proof of Claim 1. We prove by induction on j that for all $j \in [0, s]$ with $\mathtt{BND}_j$ we have

$$t_1 + \ldots + t_j \leqslant 3^j \cdot 6(1 + d^c).$$

Since for all $j \in [0, s]$,

$$3^j \cdot 6(1 + d^c) = 3^{j+1} \cdot 2(1 + d^c) \leqslant 3^{j+2}(1 + d^c) \leqslant 3^{s+2}(1 + d^c) \leqslant 3^{s_0+2}(1 + d^c),$$

this implies the claim.
Let $j = 0 \in [0, s]$ and $\mathtt{BND}_j$ be true. Then $t_1 + \ldots + t_j = 0 \leqslant 3^j \cdot 6(1 + d^c)$.
Now, let $j \in [0, s]$ such that from $\mathtt{BND}_j$ it follows that $t_1 + \ldots + t_j \leqslant 3^j \cdot 6(1 + d^c)$. If $j = s$, we are done. Otherwise assume that $\mathtt{BND}_{j+1}$ holds. Then $\mathtt{BND}_j$ also holds and so we get by induction hypothesis that $t_1 + \ldots + t_j \leqslant 3^j \cdot 6(1 + d^c)$. Thus we have

$$t_1 + \ldots + t_{j+1} \overset{\mathtt{BND}_{j+1}}{\leqslant} 2(t_1 + \ldots + t_j) + 6(1 + d^c)$$

$$\leqslant 2 \cdot \Big(3^j \cdot 6(1 + d^c)\Big) + 6(1 + d^c) \overset{j \geqslant 0}{\leqslant} 2 \cdot 3^j \cdot 6(1 + d^c) + 3^j \cdot 6(1 + d^c) = 3^{j+1} \cdot 6(1 + d^c).$$

□

Now let $j \in [0, s]$ be maximal such that $\mathtt{BND}_j$ holds. We choose

$$k := t_1 + \ldots + t_j + 2(1 + d^c).$$

By Claim 1 we get $k = t_1 + \ldots + t_j + 2(1 + d^c) \leqslant 3^{s_0+2}(1 + d^c) + 2(1 + d^c) \leqslant 3^{s_0+3}(1 + d^c) = k_0$. We want to show that G is a (k, p)-near-uniform directed graph.
Before we do that, we introduce a helpful notion: Let $u, v, w \in V$. We say that w *distinguishes* u *and* v *in* G *w.r.t.* $\mathfrak{A}$ if

- $w \in V \setminus N_c^{\mathfrak{A}}(u, v)$ and
- $w \in \Delta^G(u, v) = \big(\mathbf{N}_+^G(u) \bigtriangleup \mathbf{N}_+^G(v)\big) \cup \big(\mathbf{N}_-^G(u) \bigtriangleup \mathbf{N}_-^G(v)\big)$.

Set $\overline{U} := U_{j+1} \,\dot\cup\, \ldots \,\dot\cup\, U_s$.
Now, we can formulate the claim that will conclude this proof.

Claim 2. For all $u, v \in V$, u and v are near-k-twins in G if and only if for all $u_0, v_0 \in V$ with $u_0 \equiv_q^{\mathfrak{A}} u$ and $v_0 \equiv_q^{\mathfrak{A}} v$, there is no vertex in $\overline{U}$ that distinguishes u_0 and v_0 in G w.r.t. $\mathfrak{A}$.

From Claim 2 it follows that the near-k-twin relation is a coarsening of $\equiv_q^{\mathfrak{A}}$ and as such an equivalence relation of index at most $s_0 = p$. Thus all we need to do is prove Claim 2.

Proof of Claim 2. Let $u, v \in V$ such that for all $u_0, v_0 \in V$ with $u_0 \equiv_q^{\mathfrak{A}} u$ and $v_0 \equiv_q^{\mathfrak{A}} v$, there is no vertex in $\overline{U}$ that distinguishes u_0 and v_0 in G w.r.t. $\mathfrak{A}$. Then there is also no vertex in $\overline{U}$ that distinguishes u and v in G w.r.t. $\mathfrak{A}$. Thus by construction we have

$$\Delta^G(u, v) \subseteq \big(U_1 \,\dot\cup\, \ldots \,\dot\cup\, U_j\big) \cup N_c^{\mathfrak{A}}(u, v),$$

so

$$\left|\Delta^G(u, v)\right| \leqslant \left|\big(U_1 \,\dot\cup\, \ldots \,\dot\cup\, U_j\big) \cup N_c^{\mathfrak{A}}(u, v)\right| \leqslant t_1 + \ldots + t_j + 2(1 + d^c) = k,$$

which implies that u and v are near-k-twins.
Now, let $u, v \in V$ such that there are $u_0, v_0 \in V$ with $u_0 \equiv_q^{\mathfrak{A}} u$ and $v_0 \equiv_q^{\mathfrak{A}} v$ such that there is a vertex $w_0 \in \overline{U}$ that distinguishes u_0 and v_0 in G w.r.t. $\mathfrak{A}$. Now, we want to show that in this case, u and v are not near-k-twins. First note that this cannot happen if $j = s$, because then we would have $\overline{U} = \emptyset$. This means, in this case, the near-k-twin relation of G would be the equivalence relation on V with exactly one equivalence class. Otherwise there is an $l \in [j + 1, s]$ with $w_0 \in U_l$. Let $i \in [2]$ such that $w_0 \in \mathbf{N}_i^G(u_0) \bigtriangleup \mathbf{N}_i^G(v_0)$. Assume that $w_0 \in \mathbf{N}_i^G(u_0) \setminus \mathbf{N}_i^G(v_0)$. Recall that this is equivalent with $\mathfrak{A} \models \psi(u_0, w_0), \mathfrak{A} \not\models \psi(v_0, w_0)$ if $i = 1$ and with $\mathfrak{A} \models \psi(w_0, u_0), \mathfrak{A} \not\models \psi(w_0, v_0)$ if $i = 2$.
Define

$$W := U_l \setminus N_c^{\mathfrak{A}}(u_0, u, v_0, v).$$

Recall that $j \in [s]$ is maximal such that $\mathtt{BND}_j$ holds, and that $j \neq s$. This means that we have $t_{j+1} > t_1 + \ldots + t_j + 6(1 + d^c)$. Then by our observation, the size of W satisfies the strict lower bound

$$|W| = \left|U_l \setminus N_c^{\mathfrak{A}}(u_0, u, v_0, v)\right| \geqslant |U_l| - \left|N_c^{\mathfrak{A}}(u_0, u, v_0, v)\right|$$

SZ(=SIZE)
$$= t_l - 4(1 + d^c) \geqslant t_{j+1} - 4(1 + d^c) > t_1 + \ldots + t_j + 2(1 + d^c) = k.$$

So all we need to show is that $W \subseteq \mathbf{N}_i^G(u) \bigtriangleup \mathbf{N}_i^G(v)$. Let $w_1 \in W \subseteq U_l$. Since

- $w_0 \equiv_q^{\mathfrak{A}} w_1$,

- $\mathtt{dist}^{\mathfrak{A}}(u_0, w_0), \mathtt{dist}^{\mathfrak{A}}(u_0, w_1) > c = 2r$ and
- $w_0 \in \mathbf{N}_i^G(u_0)$,

by Corollary 5.17 we derive that $w_1 \in \mathbf{N}_i^G(u_0)$. Since

- $u_0 \equiv_q^{\mathfrak{A}} u$,
- $\mathtt{dist}^{\mathfrak{A}}(u_0, w_1), \mathtt{dist}^{\mathfrak{A}}(u, w_1) > c = 2r$ and
- $w_1 \in \mathbf{N}_i^G(u_0)$,

by Corollary 5.17 we derive that $w_1 \in \mathbf{N}_i^G(u)$. Analogously one shows that $w_1 \notin \mathbf{N}_i^G(v)$. So $W \subseteq \mathbf{N}_i^G(u) \setminus \mathbf{N}_i^G(v) \subseteq \mathbf{N}_i^G(u) \triangle \mathbf{N}_i^G(v)$ and by $\mathtt{SZ}$, it follows that in this case, u and v are indeed not near-k-twins. $\square$

$\square$

So now we know that classes of vertex- and edge-coloured directed graphs of the form $\Gamma(\mathcal{C})$ are near-uniform. We have characterized them in a quite useful way as we will see in the next section.

FO MODEL CHECKING ON NEAR-UNIFORM CLASSES OF VERTEX- AND EDGE-COLOURED DIRECTED GRAPHS

Now we finally have all the tools we need to prove the following theorem:

Theorem 6.1. *Let $\mathcal{D}$ be a class of vertex- and edge-coloured directed graphs* FO *interpretable in some class of bounded degree. Then $\mathtt{MC}(\mathcal{D})$ is fixed-parameter tractable.*

Theorem 6.1 extends the result of Gajarský et al. in [1], who proved that every graph class FO interpretable in some graph class of bounded degree admits an fpt algorithm for FO Model Checking.
By Theorem 5.7 we know that to prove Theorem 6.1, it suffices to prove the following theorem.

Theorem 6.2. *Let $\mathcal{D}$ be a near-uniform class of vertex- and edge-coloured directed graphs. Then $\mathtt{MC}(\mathcal{D})$ is fixed parameter tractable.*

In order to prove Theorem 6.2, we will, of course, use FO interpretations. The next theorem of Seese will prove very useful to us (see [5] for a proof):

Theorem 6.3 (Seese, 1996)**.** *Let $\mathcal{C}$ be a τ-class of bounded degree. Then $\mathtt{MC}(\mathcal{C})$ is fixed-parameter tractable.*

In fact, to prove Theorem 6.2, we will prove

Theorem 6.4. *Let $\mathcal{D}$ be a near-uniform class of vertex- and edge-coloured directed graphs. Then there is a class $\mathcal{C}$ of bounded degree such that $\mathcal{D}$ is efficiently simply* FO *interpretable in $\mathcal{C}$.*

We start our proof of Theorem 6.4 with the following lemma. Recall that for a σ-class $\mathcal{D}$ and any $R \in \sigma$, $\mathcal{D}^R$ is the class of all R-reducts of σ-structures in $\mathcal{D}$.

Lemma 6.5. *Let $\mathcal{D}$ be a σ-class. Then $\mathcal{D}$ is efficiently simply* FO *interpretable in some class of bounded degree if and only if for all $R \in \sigma$, $\mathcal{D}^R$ is efficiently simply* FO *interpretable in some class of bounded degree.*

Proof. Let $(\mathbf{A}, \Gamma)$ be an FO reduction from $\mathcal{D}$ to a τ-class $\mathcal{C}$ of τ-structures of maximum degree at most d. Let $R \in \sigma$. Then $(\mathbf{A}_R, \Gamma^R)$ is an FO reduction from $\mathcal{D}^R$ to $\mathcal{C}$, where $\mathbf{A}_R$ is an algorithm that, on input $\mathfrak{B} \in \mathcal{D}^R$, constructs $\mathfrak{B}' \in \mathcal{D}$ from $\mathfrak{B}$ by letting $R(\mathfrak{B}') := R(\mathfrak{B})$ and for all $R' \in \sigma \setminus \{R\}$, $R'(\mathfrak{B}') := \emptyset$, and returns the output of $\mathbf{A}$ on $\mathfrak{B}'$. Obviously, this output will have maximum degree at most d and be the Γ^R-encoding of $\mathfrak{B}$.
For the other direction, assume that for all $R \in \sigma$, $(\mathbf{A}_R, \Gamma_R)$ is an FO reduction from $\mathcal{D}^R$ to some τ_R-class $\mathcal{C}_R$ of τ_R-structures of maximum degree at most d_R. Say, for all $R \in \sigma$, it is $\Gamma_R = (\mathtt{true}, \psi_R)$. Let $\Gamma := (\mathtt{true}, (\psi_R)_{R\in\sigma})$ and $\mathbf{A}$ be an algorithm that, on input $\mathfrak{B} \in \mathcal{D}$, simulates $\mathbf{A}_R$ on input $\mathfrak{B}^R$ to obtain $\mathfrak{A}_R \in \mathcal{C}_R$ for all $R \in \sigma$, and then outputs $\mathfrak{A} := (B, (R(\mathfrak{A}_R))_{R\in\sigma})$. Note that for all $R \in \sigma$, $V(\mathfrak{A}_R) = B$ because $\mathfrak{A}_R$ is a Γ_R-encoding of $\mathfrak{B}^R$ and Γ_R is simple. Then $\mathfrak{A}$ will have maximum degree at most $d := \sum_{R\in\sigma} d_R$ and be a Γ-encoding of $\mathfrak{B}$. So $(\Gamma, \mathbf{A})$ is an FO reduction from $\mathcal{D}$ to $\mathcal{C}$. $\square$

By Theorem 6.3 and Lemmas 3.9 and 6.5, to prove Theorem 6.4, it suffices to prove the following theorem.

Theorem 6.6. *Let* R *be either a monadic or binary predicate and* $\mathcal{D}$ *a near-uniform* $\{R\}$*-class. Then* $\mathcal{D}$ *is efficiently simply* FO *interpretable in some class of bounded degree.*

Properties of the Near-k-Twin Equivalence

In this subsection, we want to investigate properties of (k, p)-near-uniform directed graphs. We start with a definition:

Definition 6.7. *Let* R *be a predicate of arity* r *and* $\mathfrak{A}$ *a* $\{R\}$*-structure. Let* $i \in [r]$, $v \in A$ *and* $U \subseteq A$. *Then we define*

$$\alpha_i^U(v) := \min\left\{\left|U \cap \mathbf{N}_i^{\mathfrak{A}}(v)\right|, \left|U \setminus \mathbf{N}_i^{\mathfrak{A}}(v)\right|\right\}.$$

If $r = 2$, *we set* $\alpha_+^U(v) := \alpha_1^U(v)$ *and* $\alpha_-^U(v) := \alpha_2^U(v)$. $\alpha_+^U(v), \alpha_-^U(v)$

The following lemma characterizes the behaviour of a vertex of one near-k-twin class to vertices of another. This lemma extends Lemma 3.4 from [1] to also work for directed graphs.

Lemma 6.8. *Let* $k \in \mathbb{N}_+$ *and* H *be a directed graph such that the near-*k*-twin relation* ρ_k^H *on* H *is an equivalence on* H. *Let* U, V *be near-*k*-twin classes of* H *with at least* $4k + 2$ *vertices each. Then for all* $v \in V$ *we have*

$$\alpha_+^U(v) \leqslant 2k \text{ and } \alpha_-^U(v) \leqslant 2k.$$

Proof. Towards a contradiction, assume there is a $v \in V$ with $\alpha_+^U(v) > 2k$ or $\alpha_-^U(v) > 2k$. Since the other case is very similar, we assume without loss of generality that $\alpha_+^U(v) > 2k$. Then there must be a subset $U' \subseteq U$ with $|U'| = 4k + 2$ and $\alpha_+^{U'}(v) = 2k + 1$.
Let $w \in V$. Assume $\alpha^{U'}(w) < \alpha^{U'}(v) - k$. Then we have

$$\alpha^{U'}(v) - \alpha^{U'}(w) > k. \qquad *$$

Let for all $v \in V(H)$, $\mathbf{N}_+(v) := \mathbf{N}_+^H(v), \mathbf{N}_-(v) := \mathbf{N}_-^H(v)$. Without loss of generality, let $\alpha^{U'}(v) = |U' \cap \mathbf{N}_+(v)|$. We distinguish two cases:

Case 1. $\alpha^{U'}(w) = |U' \cap \mathbf{N}_+(w)|$. It is $\alpha^{U'}(v) \leqslant |U' \cap \mathbf{N}_+(v)|$ by definition and by $*$ we get

$$\begin{aligned} k < \alpha^{U'}(v) - \alpha^{U'}(w) &\leqslant |U' \cap \mathbf{N}_+(v)| - |U' \cap \mathbf{N}_+(w)| \\ &\leqslant |U' \cap (\mathbf{N}_+(v) \setminus \mathbf{N}_+(w))| \overset{vw \in \rho_k^H}{\leqslant} k, \end{aligned}$$

which is a contradiction.

Case 2. $\alpha^{U'}(w) = |U' \setminus \mathbf{N}_+(w)|$. It is $\alpha^{U'}(v) \leqslant |U' \setminus \mathbf{N}_+(v)|$ by definition and by $*$ we get

$$\begin{aligned} k < \alpha^{U'}(v) - \alpha^{U'}(w) &\leqslant |U' \setminus \mathbf{N}_+(v)| - |U' \setminus \mathbf{N}_+(w)| \\ &\leqslant |U' \cap (\mathbf{N}_+(w) \setminus \mathbf{N}_+(v))| \overset{vw \in \rho_k^H}{\leqslant} k, \end{aligned}$$

which is a contradiction.

So, we have $\alpha^{U'}(w) \geqslant \alpha^{U'}(v) - k = 2k+1-k = k+1$. We will also need a lower bound for the term

$$(\alpha^{U'}(w)-1)(|U'|-\alpha^{U'}(w)-1).$$

To give an appropriate lower bound, we argue as follows. Note that $2\alpha^{U'}(w) \leqslant |U'|$ and therefore we have $\alpha^{U'}(w) \in [k+1, 2k+1]$. Then there is $j \in [1, k+1]$ with

$$(\alpha^{U'}(w)-1)(|U'|-\alpha^{U'}(w)-1) = (k+j-1)((4k+2)-(k+j)-1)$$

$$= (k+(j-1))(3k-(j-1)) = 3k^2 + 2k(j-1) - (j-1)^2$$

LOWBND $$= (j-1)(2k-j+1) + 3k^2 \overset{1\leqslant j\leqslant 2k+1}{\geqslant} 3k^2.$$

In order to provoke a contradiction, we are going to bound the number D of pairs $(w, \{u, u'\})$ such that $w \in V$, $u, u' \in U'$ are distinct vertices and exactly one of wu, wu' is an edge of H, from the top and then from the bottom. First, we fix $\{u, u'\} \subseteq U'$ with $u \neq u'$. Then there are exactly $|V \cap (\mathbf{N}_-(u) \triangle \mathbf{N}_-(u'))|$ pairs $(w, \{u, u'\})$ for $w \in V$. This yields

$$D = \sum_{\{u,u'\}\in\binom{U'}{2}} |V \cap \mathbf{N}_-(u) \triangle \mathbf{N}_-(u')| \leqslant \sum_{\{u,u'\}\in\binom{U'}{2}} |\mathbf{N}_-(u) \triangle \mathbf{N}_-(u')|$$

$$\overset{uu'\in\rho_k^H}{\leqslant} \sum_{\{u,u'\}\in\binom{U'}{2}} k = \left|\binom{U'}{2}\right| k = \binom{4k+2}{2} k$$

UB $$= \frac{k(4k+1)}{2}(4k+2) \overset{k\geq 1}{<} \frac{k(6k)}{2}(4k+2) = 3k^2(4k+2).$$

Now, we fix a $w \in V$. Note that $\alpha_+^{U'}(w) \in \{|U' \cap N_+(w)|, |U' \setminus N_+(w)|\}$. If $w \notin U'$, then there must be exactly

$$|U' \cap \mathbf{N}_+(w)| \cdot |U' \setminus \mathbf{N}_+(w)| = \alpha_+^{U'}(w) \cdot (|U'| - \alpha_+^{U'}(w)),$$

pairs $(w, \{u, u'\})$ with distinct $w \in V$, $u, u' \in U'$. If $w \in U'$, then depending on whether $\alpha_+^{U'}(w) = |U' \cap \mathbf{N}_+(w)|$ or $\alpha_+^{U'}(w) = |U' \setminus \mathbf{N}_+(w)|$ and on whether $ww \in E(H)$, we have either

$$(\alpha_+^{U'}(w)-1) \cdot (|U'| - \alpha_+^{U'}(w)),$$

or

$$\alpha_+^{U'}(w) \cdot (|U'| - \alpha_+^{U'}(w) - 1)$$

many desired pairs $(w, \{u, u'\})$. In all cases, we have at least

$$(\alpha_+^{U'}(w)-1) \cdot (|U'| - \alpha_+^{U'}(w) - 1)$$

pairs $(w, \{u, u'\})$. This yields the lower bound on D as follows:

$$D \geqslant \sum_{w\in V} (\alpha_+^{U'}(w)-1) \cdot (|U'| - \alpha_+^{U'}(w) - 1)$$

$$\overset{\text{by LOWBND}}{\geqslant} \sum_{w\in V} 3k^2 \geqslant 3k^2(4k+2),$$

which contradicts our strict upper bound UB on D. $\square$

We will also need the following corollary in the proof of Theorem 6.6. The corollary is the pendant to Corollary 3.5 from [1].

Corollary 6.9. *Let $k \in \mathbb{N}$ and H be a directed graph such that the near-k-twin relation ρ_k^H of H is an equivalence relation on $V(H)$. Let U and V be near-k-twin classes of order at least $5k+1$ each. Then exactly one of the following two cases is true:*

1. *Every vertex of U has at most $2k$ successors in V and every vertex of V has at most $2k$ predecessors in U.* *almost no edges from U to V*

2. *Every vertex of U has at least $|V| - 2k$ successors in V and every vertex of V has at least $|U| - 2k$ predecessors in U.* *almost no non-edges from U to V*

Proof. First we assume $k > 0$ and show that either

1. every vertex of U has at most $2k$ successors in V, or

2. every vertex of U has at least $|V| - 2k$ successors in V.

Let $u \in U$. Then at most one of the two cases can happen since $|V| > 4k$, and by Lemma 6.8 at least one of the two cases has to happen. Assume, we had $u, u' \in U$ such that u has at most $2k$ successors in V, but u' at least $|V| - 2k$. Then we would have $k = |\mathbf{N}_+^H(u) \triangle \mathbf{N}_+^H(u')| \geqslant |V| - 2k - 2k = k + 1$, which is a contradiction. Very similarly one can prove that either

1. every vertex of V has at most $2k$ predecessors in U, or

2. every vertex of V has at least $|U| - 2k$ predecessors in U.

All that remains to be shown is that the following case is impossible: Every vertex of U has at most $2k$ successors in V and every vertex of V has at least $|U| - 2k$ predecessors in U. The possibility of the case that every vertex of U has at least $|V| - 2k$ successors in V and every vertex of V has at most $2k$ predecessors in U, can be disproved in an analogous way. To bring the given case to a contradiction, we want to give a lower and an upper bound to the number of edges from U to V such that the lower bound is strictly greater than the upper bound. On the one hand, this number is at least $|V|(|U| - 2k)$, but on the other hand, this number is at most $|U|2k$. Their difference is

$$|V|(|U| - 2k) - |U|2k = |U| \cdot |V| - 2k(|U| + |V|) =$$

$$\left(|U| \cdot |V| - 4k(|U| + |V|) + 16k^2\right) + 2k(|U| + |V|) - 16k^2 =$$

$$(|U| - 4k)(|V| - 4k) + 2k(|U| + |V|) - 16k^2 > k \cdot k + 2k(5k + 5k) - 16k^2 = 5k^2 > 0,$$

which is again a contradiction.
Now, assume $k = 0$. Then the claim degenerates to the following: It is either

1. $E(H) \cap (U \times V) = \emptyset$, or

2. every vertex of U has at least one successor in V and every vertex of V has at least one predecessor in U.

Obviously, both cases cannot be true at the same time. Assume that it is $E(H) \cap (U \times V) \neq \emptyset$. Then there is an edge $uv \in E(H)$ with $u \in U, v \in V$. Let $u' \in U, v' \in V$. Since $uu' \in \rho_0^H$, it must be $u'v \in E(H)$, and since $vv' \in \rho_0^H$, it must be $uv' \in E(H)$. This shows the claim. $\square$

This corollary motivates the following definition.

Definition 6.10. *Let R be some predicate. Let $k \in \mathbb{N}$ and $\mathfrak{A}$ be a $\{R\}$-structure such that the near-k-twin relation is an equivalence relation on $\mathfrak{A}$. Then a near-k-twin class of $\mathfrak{A}$ is called* big *if its size is at least $5k+1$. Otherwise it is called* small.

FO Reduction From $\mathcal{D}$ to Some Class of Bounded Degree

In this subsection, we are finally able to give the algorithm of our FO reduction from $\mathcal{D}$ to some class of bounded degree.
First we need the following simple algorithm that determines the minimal $k \in [0, k_0]$ such that a directed graph $H \in \mathcal{D}$ is (k,p)-near-uniform, and also computes the corresponding big near-k-twin classes.

Lemma 6.11. *Let $k_0, p \in \mathbb{N}$. Then there is a polynomial time algorithm $\mathbb{B}_{k_0,p}$ that, given a directed graph H such that H belongs to a (k_0,p)-near-uniform class, computes the minimal $k \in [0,k_0]$ and $U_1, \ldots, U_s \subseteq V(H)$ such that ρ_k^H is an equivalence relation on $V(H)$ of index at most p and $U_1, \ldots, U_s$ are exactly the big near-k-twin classes of H (in any order).*

idea of Algorithm 1

Proof. The algorithm is given as Algorithm 1. Its idea is to iterate over all $k \in [0, k_0]$ and to check whether this k induces a near-k-twin relation on $V(H)$ of index at most p, and if it does, to return the big near-k-twin classes of H.

correctness of lines $1-21$

In lines 1 to 9 the sizes of all differences of sets of successors and predecessors of vertices are computed in advance. In lines 11 to 21 we check whether a given $k \in [0, k_0]$ induces a near-k-twin equivalence. The array `class` memorizes for all $v \in V(H)$ the current class of v w.r.t. to ρ_k^H if ρ_k^H were an equivalence relation. At the beginning, `class` is initialized with a discrete equivalence relation. In line 14 the vertices v_i and v_j are found to not be in the same class although they should be as confirmed in line 13. This could mean that we just arrived at the vertex v_j for the first time. But this is only the case if v_j belongs to the class indexed by j, otherwise this means that v_j has already been assigned a different class j' that is smaller than j in line 18. But since v_i and v_j still are in different classes at this point, this can only mean that v_i did not get assign the same class as v_j earlier in the loop. So we have that $v_i v_j \in \rho_k^H$ and that $v_j v_{j'} \in \rho_k^H$, but $v_i v_{j'} \notin \rho_k^H$. Hence, ρ_k^H is not an equivalence relation on $V(H)$ and we discard the current k in line 16.

correctness of lines $22-35$

In lines 22 to 34 we check whether the current near-k-twin equivalence is an equivalence relation of index at most p. To do this, in lines 22 to 29 we initialize a dictionary to store for each $i \in [n]$, the i^{th} near-k-twin class of H w.r.t. to the ordering of the classes implied by `class`. And in lines 30 to 34 we eventually compute the index s of ρ_k^H and check whether it is not greater than p. The for-loop at 10 has to succeed for some $k \in [0, k_0]$ because H belongs to some (k_0, p)-near-uniform class.

start of runtime analysis of Algorithm 1

Now we want to analyse the running time of Algorithm 1. In lines 2 to 5 we sort the successors and the predecessors of each vertex in H w.r.t. to the

Algorithm 1 $\mathbb{B}_{k_0,p}$

Input: directed graph H of order n that belongs to some (k_0, p)-near-uniform class

Output: $(k, (U_1, \dots, U_s))$, where $0 \leqslant k \leqslant k_0$ is minimal such that ρ_k^H is an equivalence relation of index at most p and $U_1, \dots, U_s$ are exactly the big near-k-twin classes of H

Let $V(H) = \{v_1, \dots, v_n\}$ be an arbitrary ordering of $V(H)$
for $i \in [n]$ **do**
  $\mathcal{N}_+(v_i) \leftarrow \mathtt{sort}(\mathbf{N}_+^H(v_i))$ // sort in ascending order
  $\mathcal{N}_-(v_i) \leftarrow \mathtt{sort}(\mathbf{N}_-^H(v_i))$
end for
for $i < j \in [n]$ **do**
  $\mathtt{diff}_+[v_i][v_j] \leftarrow |\mathcal{N}(v_i) \triangle \mathcal{N}(v_j)|$
  $\mathtt{diff}_-[v_i][v_j] \leftarrow |\mathcal{N}(v_i) \triangle \mathcal{N}(v_j)|$
end for
for $k = k_0, k_0 - 1, \dots, 0$ **do**
  $\mathtt{class}[v_i] \leftarrow i$ for all $i \in [1, n]$
  for $i < j \in [1, n]$ **do**
    if $\mathtt{diff}_+[v_i][v_j] \leqslant k$ and $\mathtt{diff}_-[v_i][v_j] \leqslant k$ **then**
      if $\mathtt{class}[v_j] \neq \mathtt{class}[v_i]$ **then**
        if $\mathtt{class}[v_j] \neq j$ **then**
          Continue with $k \leftarrow k - 1$ // ρ_k^H is not an equiv. rel.
        end if
        $\mathtt{class}[v_j] \leftarrow \mathtt{class}[v_i]$
      end if
    end if
  end for
  Initialize dictionary (set of key/value pairs) R
  for $i \in [n]$ **do**
    if $\mathtt{class}[i] \notin R.\mathtt{keys}()$ **then**
      Update R with key/value pair $(\mathtt{class}[i], \{v_i\})$
    end if
    Let U be the value of the key $\mathtt{class}[i]$ in R
    $U \leftarrow U \cup \{v_i\}$
  end for
  Let s be the number of key/value pairs in R
  if $s \leqslant p$ **then**
    Let $U_1, \dots, U_s$ be the values of all keys in R in any order
    return $(k, U_1, \dots, U_s)$
  end if
end for

arbitrarily chosen ordering of the universe of H in line 1. We do this so we can compute the symmetric difference in lines 6 to 9 more efficiently.

compute symmetric difference efficiently

Lines 7 and 8 can be implemented as follows: Let $i < j \in [n]$. For example, in order to compute $|\mathbf{N}_+(v_i) \triangle \mathbf{N}_+(v_j)|$, we use the sorted lists $\mathcal{N}_+(v_i)$ and $\mathcal{N}_+(v_j)$ of the neighbourhoods in the following way: We initialize a variable `sym_dif` which will store the size of the symmetric difference computed so far. Let `sym_dif` $:= 0$. Then we repeat the following procedure until one of the lists is empty: We consider the first element w of $\mathcal{N}_+(v_i)$ and the first element w' of $\mathcal{N}_+(v_j)$. If $w = w'$, then we remove both of them from the lists and repeat this procedure for the new shorter lists. Otherwise, it is $w < w'$ or $w > w'$, say $w < w'$. Since the lists are ordered and $w < w'$, w cannot appear in $\mathcal{N}_+(v_j)$. But it is in $\mathcal{N}_+(v_i)$ and therefore in $\mathbf{N}_+(v_i) \triangle \mathbf{N}_+(v_j)$. We increase `sym_dif` by 1 and repeat this procedure for the lists $\mathcal{N}_+(v_i) \setminus \{w\}$ and $\mathcal{N}_+(v_j)$. When one of the lists is empty, the other list can only contain elements from the symmetric difference, so we increase `sym_dif` by the size of the other list.

It is easy to see that such a procedure computes the size of the symmetric difference of $\mathbf{N}_+(v_i)$ and $\mathbf{N}_+(v_j)$. We needed $\mathcal{O}(n\log(n))$ steps to sort the neighbourhood of one vertex, so $\mathcal{O}(n^2\log(n))$ to sort all the neighbourhoods. For performing the above procedure, we need time $\mathcal{O}(n)$ for each pair of vertices, so time $\mathcal{O}(n^3)$ in total. Hence, we need time $\mathcal{O}(n^3)$ for the initialization in lines 1 to 9.

To execute lines 11 to 21 we apparently need time $\mathcal{O}(n^2)$ and for the lines 22 to 34 time $\mathcal{O}(n^2)$ as well, so for the main loop from 10 to 35, we need in total time $\mathcal{O}(k_0 n^2)$. Thus, our running time is about $\mathcal{O}(k_0 n^2 + n^3) \subseteq \mathcal{O}(k_0 n^3) = \mathcal{O}(n^3)$. □

Note that Algorithm 1 does not return all near-k-twin classes of H but only the big ones.

Now, we start our proof of Theorem 6.6.

case $\mathrm{ar}(R) = 1$

Proof of Theorem 6.6. If R is a monadic predicate, then $\mathcal{D}$ is a class of coloured sets and as such already a class of bounded degree. So we FO reduce $\mathcal{D}$ to itself via a trivial algorithm and a trivial interpretation that do nothing. In the following, we assume that $E := R$ is a binary predicate.

Let $k_0, p \in \mathbb{N}$ such that $\mathcal{D}$ is (k_0, p)-near-uniform. Set $l := 4p^2 + 15k_0p$ and $d := 4k_0p$. We will FO reduce $\mathcal{D}$ to the class $\vec{\mathcal{G}}_d^{(l,1)}$ of $(l,1)$-labeled directed graphs of maximum degree at most d.

We want to find an FO reduction $(\mathbf{A}_{k_0,p}, \Gamma_{k_0,p})$ from $\mathcal{D}$ to $\vec{\mathcal{G}}_d^{(l,1)}$. Let $\mathbf{A} := \mathbf{A}_{k_0,p}, \Gamma := \Gamma_{k_0,p}$. Let $\mathbf{A}$ be Algorithm 2. This algorithm takes as input a directed graph and returns its Γ-encoding in polynomial time. The main idea of the algorithm is as follows: On input $H \in \mathcal{D} \subseteq \mathrm{Str}(\{E\})$, we compute the minimal $k \in [0, k_0]$ and $U_1, \dots, U_s \subseteq V(H)$ such that the near-k-twin relation is an equivalence relation on $V(H)$ with index at most p and big near-k-twin classes $U_1, \dots, U_s$. Then we look at each pair (i,j) of indices of those equivalence classes to determine whether there are "a lot" of edges going from U_i to U_j or "a few". We know by Corollary 6.9 that exactly one of theses cases has to happen. Then we decide whether to use "real edges" or "exception edges" for the pair of equivalence classes indexed by (i,j). After we have done this for all such pairs, the maximum degree of the $U_i \cup U_j$-induced substructure will be at most $4k$ for any $i,j \in [s]$. Thus, the maximum degree of the whole resulting structure will be at most $4ks \leqslant 4k_0p$. In order to not forget which edges are "real" and which edges are "exceptions", we use labels.

main idea of Algorithm 2

Algorithm 2 $\mathbf{A}_{k_0,p}$

Input: directed graph H that belongs to some (k_0, p)-near-uniform class
Output: $(4p^2 + 15k_0p, 1)$-labeled directed graph G of maximum degree at most $4k_0p$ such that $\Gamma_{k_0,p}(G) = H$

$F \leftarrow \emptyset$
Initialize $G := (V(H), E(G), (\mu(G))_{\mu \in L})$ with $E(G(H)) := \mu(G) := \emptyset$ for all $\mu \in L$ (where L is our label set as in the proof of Theorem 6.6)
Initialize array U of length n
Let $(k, (U_1, \dots, U_s))$ be the output of $\mathbb{B}_{k_0,p}$ (Algorithm 1) on input H
for all $i \in [s]$ **do**
    for all $v \in U_i$ **do**
        $U[v] \leftarrow i$
    end for
end for
for all $(i, j) \in [s]^2$ **do**
    Choose $v \in U_i$ arbitrarily
    if $|U_j \cap N_+^H(v)| > 2k$ **then**
        Add ij to F // F will contain the pairs $ij \in [s]^2$ such that there are "a lot" of edges from U_i to U_j
    end if
end for
for all $ij \in F, v \in U_i, w \in U_j$ **do**
    Add v to $\mu'_{(i,j,1)}(G)$
    Add w to $\mu'_{(i,j,2)}(G)$
    if $vw \notin E(H)$ **then**
        Add vw to $E(G)$ // exception edges
    end if
end for
for all $ij \notin F, v \in U_i, w \in U_j$ **do**
    Add v to $\mu_{(i,j,1)}(G)$
    Add w to $\mu_{(i,j,2)}(G)$
    if $vw \in E(H)$ **then**
        Add vw to $E(G)$ // real edges
    end if
end for
Let $\{w_1, \dots, w_{5k_0p}\}$ be an arbitrary enumeration of $V(H) \setminus \left(\bigcup_{i \in [s]} U_i\right)$ // Handle the small near-k-twin classes
for all $j \in [5k_0p]$ **do**
    $\eta_j(G) \leftarrow \{w_j\}$
    $\eta_j^+(G) \leftarrow \mathbf{N}_+^H(w_j)$
    $\eta_j(G) \leftarrow \mathbf{N}^H(w_j)$
end for
return G

We want to define the FO interpretation Γ and then prove that **A** indeed computes a Γ-encoding of the input in polynomial time.
Let $\overline{W} := V(H) \setminus \left(\bigcup_{i\in[s]} U_i\right)$ be the set of all vertices in small near-k-twin classes of H. Then $|\overline{W}| \leqslant 5k(p-s) \leqslant 5k_0 p$ by Definition 6.10.
For all $i \in [p]^2$, let $\mu_{(i,j,1)}, \mu_{(i,j,2)}, \mu'_{(i,j,1)}, \mu'_{(i,j,2)}$ be four fresh monadic predicates. Furthermore, for all $m \in [5k_0p]$, let $\eta_m, \eta^-_m, \eta^+_m$ be three fresh monadic predicates. Then

$$L := \left\{\mu_{(i,j,1)}, \mu_{(i,j,2)}, \mu'_{(i,j,1)}, \mu'_{(i,j,2)}, \eta_m, \eta^-_m, \eta^+_m \mid (i,j) \in [p]^2, m \in [5k_0p]\right\}$$

colour set L

will be our set of colours for the encoding. We easily see that L is finite with $|L| = 4p^2 + 15k_0p = l$. We explain our use of colours and other important variables in the algorithm. Let $(i,j) \in [p]^2$, $m \in [5k_0p]$.

colours $\mu'_{(i,j,1)}, \mu'_{(i,j,2)}$

- For all $(i,j) \in F$, $(u,v) \in U_i \times U_j$, we colour u with $\mu'_{(i,j,1)}$ and v with $\mu'_{(i,j,2)}$.

μ' refers to "exception edges" and μ refers to "real edges"

colours $\mu_{(i,j,1)}, \mu_{(i,j,2)}$

- For all $(i,j) \notin F$, $(u,v) \in U_i \times U_j$, we colour u with $\mu_{(i,j,1)}$ and v with $\mu_{(i,j,2)}$.

colour η_i

- We colour a vertex with η_m if it is the m^{th} vertex in a fixed enumeration of the union $V(G) \setminus \bigcup_{h\in[s]} U_h$ of "small" equivalence classes

colour η^-_m

- We colour a vertex with η^-_m if it is a predecessor of the m^{th} vertex in that same enumeration.

colour η^+_m

- We colour a vertex with η^+_m if it is a successor of the m^{th} vertex in that same enumeration.
- The set

  $$F = \{ij \in [s]^2 \mid \text{ it is } |U_j \cap N^H_+(v)| > 2k \text{ for some arbitrarily chosen } v \in U_j\}$$

  set F

  contains by Corollary 6.9 exactly those pairs $ij \in [s]^2$ such that there are "a lot" of edges from U_i to U_j.

$\psi(x,y)$

Now, we want to give a simple FO interpretation $\Gamma = \Gamma_{k_0,p} := (\mathtt{true}, \psi(x,y))$ from $\{E\} \cup L$ to $\{E\}$ such that Algorithm **A** computes, given a directed graph from $\mathcal{D} \subseteq \mathtt{Str}(E)$, one of its Γ-encodings in $\vec{\mathcal{G}}^{(l,1)}_d$. We set

DIS1
$$\psi(x,y) := \bigvee_{ij\in[p]^2} \left(\mu'_{(i,j,1)}(x) \wedge \mu'_{(i,j,2)}(y) \wedge \neg E(x,y)\right)$$

DIS2
$$\vee \bigvee_{ij\notin[p]^2} \left(\mu_{(i,j,1)}(x) \wedge \mu_{(i,j,2)}(x) \wedge E(x,y)\right)$$

DIS3
$$\vee \bigvee_{j\in[5k_0p]} \left(\eta_j(x) \wedge \eta^+_j(y)\right)$$

DIS4
$$\vee \bigvee_{j\in[5k_0p]} \left(\eta_j(y) \wedge \eta^-_j(x)\right).$$

Then $\psi(x,y) \in \mathrm{FO}[\{E\} \cup L]$. Note that ψ, and thus Γ, is independent of F and thus of the particular $\{E\}$-structure H in $\mathcal{D}$.

We want to prove that the algorithm **A**, given a directed graph H, computes in polynomial time a labeled directed graph G_H of maximum degree at most $4k_0p$ such that G_H is a Γ-encoding of H. correctness of Algorithm 2

To this goal, let H be a directed graph and $G := G_H$ the output of **A** on input H. Then the universes of H and G are equal by line 2 of the algorithm. Let $V := V(H) = V(G)$. Let $(k, (U_1, \ldots, U_s))$ be as computed in line 4. Let $\{w_1, \ldots, w_{5k_0p}\}$ be the enumeration of $\overline{W} = V(H) \setminus \left(\bigcup_{i\in[s]} U_i\right)$ as chosen in line 30. We prove the following claim.

Claim 1. Let $v, w \in V$. Then vw is an edge in H if and only if one of the following cases is true.

Case 1. There is $ij \in F$ such that $v \in \mu'_{(i,j,1)}(G), w \in \mu'_{(i,j,2)}(G)$ and $vw \notin E(G)$.

Case 2. There is $ij \notin F$ such that $v \in \mu_{(i,j,1)}(G), w \in \mu_{(i,j,2)}(G)$ and $vw \in E(G)$.

Case 3. There is $m \in [5k_0p]$ such that $\eta_m(G) = \{v\}$ and $w \in \eta^+_m(G)$.

Case 4. There is $m \in [5k_0p]$ such that $\eta_m(G) = \{w\}$ and $v \in \eta^-_m(G)$.

Proof of Claim 1. Let $vw \in E(H)$. Assume there is $ij \in [p]^2$ with $v \in U_i, w \in U_j$, and $ij \in F$. Then in lines 17, 18 we will colour $v \in \mu'_{(i,j,1)}(G), w \in \mu'_{(i,j,2)}(G)$, but since $vw \in E(H)$, we will never add the edge vw to G in line 20, so $vw \notin E(G)$. Assume $ij \notin F$. Then analogously in lines 23 to 29, we will colour $v \in \mu_{(i,j,1)}(G), w \in \mu_{(i,j,2)}(G)$ and add the edge vw to G, so $vw \in E(G)$.
Now, assume that there is no such ij. Then there must be an $m \in [5k_0p]$ such that $v = w_m$ or $w = w_m$, say $v = w_m$. Then $\eta_m(G) = \{v\}$ by line 32. Since $vw \in E(H)$, by construction we will add w to $\eta^+_m(G)$ in line 33. If $w = w_m$, then by a similar argument we will have $\eta_m(G) = \{w\}$ and $v \in \eta^-_m(G)$.
For the other direction, assume Case 1. Then in line 20, the edge vw has not been added to G which, since v and w are both members of big near-k-twin classes, can only happen if $vw \in E(H)$. Assume Case 2. Then in line 27, the edge vw has been added to G which means that $vw \in E(H)$. Assume Case 3. Then according to line 33 it must be that $w \in N^H_+(v)$, so $vw \in E(H)$. Assume Case 4. Then according to line 34 it must be that $v \in N^H_-(w)$, so $vw \in E(H)$. Thus, if one of the cases is true, then $vw \in E(H)$. □

We want to prove that $\Gamma(G) = H$. Since $\psi_{univ} \equiv \mathtt{true}$ and $V(H) = V = V(G)$, all that remains to prove is that $E(H) = \psi(G)$. Let $v, w \in V$. Then at least one of the following four cases must be true.

Case 1. There is $ij \in [p]^2$ with $v \in U_i, w \in U_j$ and $ij \in F$. By construction it is $v \in \mu'_{(i,j,1)}(G), w \in \mu'_{(i,j,2)}(G)$, and so it is $(v, w) \notin E(G)$ if and only if $(v, w) \in E(H)$ by Claim 1. Obviously, in this case, the disjunctions `DIS2`, `DIS3`, `DIS4` will also be false in G. If $(v, w) \in E(H)$, then one disjunct of the disjunction `DIS1` will be true in H. Otherwise, the same disjunct will be false in G and since for all other $i'j' \in F$, at least one of $v \in \mu'_{(i',j',1)}(G), b \in \mu'_{(i',j',2)}(G)$ will be false, all other disjuncts in `DIS1` will be false in G as well. Hence, we have $(v, w) \in E(H)$ if and only if $G \models \psi(v, w)$.

Case 2. There is $ij \in [p]^2$ with $v \in U_i, w \in U_j$ and $ij \notin F$. Then the argument is analogous to the one in Case 1. Hence, we have $(v, w) \in E(H)$ if and only if $G \models \psi(v, w)$.

Case 3. There is $m \in [5k_0p]$ such that $v = w_m \in \overline{W}$, so $\eta_m(G) = \{v\}$. Then, of course, v is in none of the big near-k-twin classes of H, and thus by construction, $v \notin \mu_{(i,j,1)}(G), v \notin \mu'_{(i,j,1)}(G)$ for all $ij \in [p]^2$, which means DIS1, DIS2 will be false in G. By Claim 1, it is $G \models \eta^+_m(w)$ if and only if $(v, w) = (w_m, w) \in E(H)$. So, if $(v, w) \in E(H)$, then at least one disjunct of the disjunction DIS3 will be true in G. Otherwise, the disjunction DIS3 will be false in G. Now, we argue that DIS4 must be false, too: If there is no $m' \in [5k_0p]$ with $w = w_{m'}$, this is clear. Otherwise, let $m' \in [5k_0p]$ with $w = w_{m'}$. Then by Claim 1, it is $G \models \eta^-_{m'}(v)$ if and only if $(v, w) = (v, w_{m'}) \in E(H)$. Since the latter is false, the former is false, too. Hence, we have $(v, w) \in E(H)$ if and only if $G \models \psi(v, w)$.

Case 4. There is $m \in [5k_0p]$ such that $w = w_m \in \overline{W}$. Then the argument is analogous to the one in Case 3. Hence, we have $(v, w) \in E(H)$ if and only if $G \models \psi(v, w)$.

All in all, we have $(v, w) \in E(H)$ if and only if $G \models \psi(v, w)$, so $E(H) = \psi(G)$. Hence, we have $\Gamma(G) = H$.

runtime analysis of Algorithm 2

Now, we want to analyse the running time of Algorithm 2. As discussed in the proof of Lemma 6.11 we need time $\mathcal{O}(n^3)$ for line 4. In lines 5 to 9 we apparently need time $\mathcal{O}(sn) \subseteq \mathcal{O}(pn) = \mathcal{O}(n)$. In lines 10 to 15 we need time $\mathcal{O}(p^2n) = \mathcal{O}(n)$ if we use the array U that we initialized earlier. The code in lines 17 to 21 can be executed in constant time if we use an adjacency matrix encoding for H to look up whether vw is an edge in H, and the code in lines 24 to 28 analogously. So the for-loops in lines 16 to 22 and lines 23 to 29 respectively can be executed in time $\mathcal{O}(p^2n^2) = \mathcal{O}(n^2)$. The last for-loop from 31 to 35 runs in time $\mathcal{O}(5k_0pn) = \mathcal{O}(k_0pn) = \mathcal{O}(n)$. All in all, our algorithm runs in time

$$\mathcal{O}(k_0n^3 + pn + p^2n + 2p^2n^2 + k_0pn)$$
$$= \mathcal{O}(k_0n^3 + p^2n^2 + k_0pn) \subseteq \mathcal{O}(k_0p^2n^3) = \mathcal{O}(n^3).$$

□

The following is a simple but meaningful corollary of the whole thesis.

Corollary 6.12. *The classes of vertex- and edge-coloured directed graph* FO *interpretable in some class of bounded degree are exactly the near-uniform classes of vertex- and edge-coloured directed graphs.*

Proof. By Theorem 5.7, every class of vertex- and edge-coloured directed graphs FO interpretable in some class of bounded degree is near-uniform. By Theorem 6.4, every near-uniform class of vertex- and edge-coloured directed graphs is efficiently simply FO interpretable, in particular FO interpretable, in some class of bounded degree. □

CONCLUSION

What is new about the approach in [1] is that the classes on which we want to solve the FO Model Checking Problem have the form $\Gamma(\mathcal{C})$, where Γ is an FO interpretation and $\mathcal{C}$ a class admitting an fpt algorithm for FO Model Checking. We have shown that all of those classes admit an fpt algorithm for Model Checking if $\mathcal{C}$ is a class of bounded degree and Γ is an FO interpretation to some signature σ containing only predicates of arity at most 2. Now there are several possibilities to go from here:

1. It is quite unlikely that a similar construction will work if we also allow predicates of arity 3 in σ. An indicator for this is that a natural way to extend our Algorithm 2 to also work for predicates of arity 3 would be to replace the predicates η_l^+, η_l^- by $\eta_l^1, \eta_l^2, \eta_l^3$ where for example η_l^1 would contain all pairs (a, b) such that (w_l, a, b) is a triple in the relation. But then these new predicates could increase the maximum degree of the output structure G. For example, consider the $\{R\}$-structure $\mathfrak{B}$ in Figure 6 with $r := \mathrm{ar}(R) \geqslant 3$, where we let $R(\mathfrak{B})$ contain all r-tuples that are formed from vertices in the red square together with the lonely vertex or from vertices in the red and green squares. Then the vertices from the red and green squares

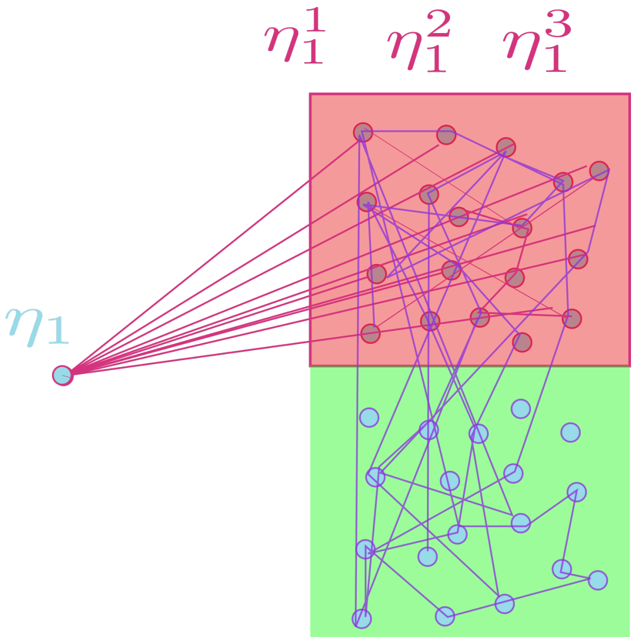

Figure 6: $(1, 2)$-near-uniform structure with a vertex in a small near-1-twin class that has an unbounded degree

together form one big near-1-twin class and the lonely vertex forms a small near-1-twin class. And so if we built the structure G from $\mathfrak{B}$ analogously to our construction in Algorithm 2 and also used the new colours as explained above, then even the maximum degree of the $\{\eta_1^1\}$-reduct of G could not be bounded in terms of k_0, p. This might motivate to try to prove that one cannot FO reduce classes of the form $\Gamma(\mathcal{C})$ to any class of bounded degree, if Γ is an FO interpretation to $\{R\}$ where $\mathrm{ar}(R) \geqslant 3$, or one could maybe even try to prove that these classes do not admit an fpt algorithm at all.

2. Another possibility would be to change the sparse class that we interpret in, so go from classes of bounded degree to classes of locally bounded treewidth or classes with an excluded minor or even nowhere dense classes. For this purpose, one could define a class of structures to be of locally bounded treewidth, have an excluded minor or be nowhere dense if the class of corresponding Gaifman graphs is of locally bounded treewidth, has an excluded minor or is nowhere dense etc, or use a more individual definition. Or maybe it is even more productive to restrict oneself to graph classes at first.

This might be the beginning of a fruitful study of dense (graph) classes.

REFERENCES

[1] J. Gajarský, P. Hliněný, J. Obdržálek, D. Lokshtanov, and M. S. Ramanujan. A new perspective on fo model checking of dense graph classes. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '16, pages 176–184, New York, NY, USA, 2016. ACM.

[2] Martin Grohe and Stephan Kreutzer. Methods for algorithmic meta theorems. 558, 01 2011.

[3] Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. *CoRR*, abs/1311.3899, 2013.

[4] Haim Gaifman. On local and non-local properties. *Studies in Logic and the Foundations of Mathematics*, 107:105–135, 1982.

[5] Detlef Seese. Linear time computable problems and first-order descriptions, 1996.