Algorithmic Model Theory
SS 2016

Prof. Dr. Erich Grädel and Dr. Wied Pakusa

Mathematische Grundlagen der Informatik
RWTH Aachen

Contents

1 The classical decision problem

The classical decision problem was generally considered as the main problem of mathematical logic until its unsolvability was proved by Church and Turing in 1936/37.

> Das Entscheidungsproblem ist gelöst, wenn man ein Verfahren kennt, das bei einem vorgelegten logischen Ausdruck durch endlich viele Operationen die Entscheidung über die Allgemeingültigkeit bzw. Erfüllbarkeit erlaubt. (...) Das Entscheidungsproblem muss als das Hauptproblem der mathematischen Logik bezeichnet werden. [1]
>
> (D. Hilbert and W. Ackermann, Grundzüge der theoretischen Logik, 1928)

By a *logical expression*, Hilbert and Ackermann meant what we now call a formula of first-order logic (FO). Historically, the classical decision problem was part of Hilbert's formalist programme for the foundations of mathematics. Its importance stems from the fact that first-order logic provides a framework to express almost all aspects of mathematics.

We present three equivalent formulations of the classical decision problem.

Satisfiability: Construct an algorithm that decides for any given formula of FO whether it has a model.

Validity: Construct an algorithm that decides for any given formula of FO whether it is valid, i.e. whether it holds in all models where it is defined.

[1] The Entscheidungsproblem is solved when we know a procedure that allows for any given logical expression to decide by finitely many operations its validity or satisfiability. [...] The Entscheidungsproblem must be considered the main problem of mathematical logic.

Provability: Construct an algorithm that decides for any given formula ψ of FO whether $\vdash \psi$, meaning that ψ is provable from the empty set of axioms in some complete formal system such as the sequent calculus.

Since ψ is satisfiable if, and only if, $\neg\psi$ is not valid, satisfiability and validity are equivalent problems with respect to computability. The equivalence with provability is a much more intricate result and in fact a consequence of Gödel's Completeness Theorem.

Theorem 1.1 (Completeness Theorem (Gödel))**.** For any given set of sentences $\Phi \subseteq \mathrm{FO}(\tau)$ and any sentence $\psi \in \mathrm{FO}(\tau)$ it holds that

$$\Phi \models \psi \iff \Phi \vdash \psi \,.$$

In particular $\emptyset \models \psi \Leftrightarrow \emptyset \vdash \psi$.

Corollary 1.2. The set of valid first-order formulae is recursively enumerable.

1.1 Basic notions on decidability

In our formulation of the decision problem it was not precisely specified what an algorithm is. It was not until the 1930s that Church, Kleene, Gödel, and Turing provided precise definitions of an abstract algorithm. Their approaches are today known to be equivalent. We introduce the concept of a Turing machine.

Definition 1.3. A *Turing machine* (TM) M is a tuple $M = (Q, \Sigma, \Gamma, q_0, F, \delta)$, where

- Q is a finite set of (control) states,
- Σ, Γ are finite alphabets, where Σ is the working alphabet with a special blank symbol $\square \in \Sigma$, and $\Gamma \subseteq \Sigma \setminus \{\square\}$ is the input alphabet,
- $q_0 \in Q$ is the initial state,
- $F \subseteq Q$ is the set of final states and
- $\delta : (Q \setminus F) \times \Sigma \to Q \times \Sigma \times \{-1, 0, 1\}$ is the transition function.

A *configuration* is a triple $C = (q, p, w) \in Q \times \mathbb{N} \times \Sigma^*$, representing the situation that M is in state q, reads tape cell p and that the inscription of the infinite tape is $w = w_0 \dots w_k$, followed by infinitely

many blank-symbols. The transition function δ induces a partial function on the set of all configurations $C \mapsto \text{Next(C)}$, where for $\delta(q, w_p) = (q', a, m)$, the successor configuration of C is defined as $\text{Next(C)} = (q', p + m, w_0 \dots w_{p-1} a w_{p+1} \cdots w_k)$. A *computation* of the TM M on an input word $x \in \Gamma^*$ is a sequence

$$C_0, C_1, \dots$$

where $C_0 = C_0(x) := (q_0, 0, x)$ is the input configuration and $C_{i+1} = \text{Next}(\text{C}_\text{i})$ for all i.

M *halts* on x if the computation of M on x is finite and ends in a final configuration $C_f = (q, p, w)$ with $q \in F$. Further

$$L(M) := \{x \in \Gamma^* : M \text{ halts on } x\}.$$

A Turing machine M computes a partial function $f_M : \Gamma^* \to \Sigma^*$ with domain $L(M)$ such that $f_M(x) = y$ if and only if the computation of M on x ends in (q, p, y) for some $q \in F$, $y \in \Sigma^*$ and $p \in \mathbb{N}$.

Definition 1.4. A *Turing acceptor* is a Turing machine M with $F = F^+ \cup F^-$. We say that M *accepts* x if the computation of M on x ends in a state in F^+ and M *rejects* x if the computation of M on x ends in a state in F^-.

Definition 1.5.

- $L \subseteq \Gamma^*$ is *recursively enumerable (r.e.)* if there exists a TM M with $L(M) = L$.
- $L \subseteq \Gamma^*$ is *co-recursively enumerable (co-r.e.)* if $\overline{L} := \Gamma^* \setminus L$ is r.e..
- A (partial) function $f : \Gamma^* \to \Sigma^*$ is *(Turing) computable* if there is a TM M with $f_M = f$.
- $L \subseteq \Gamma^*$ is *decidable* (or *recursive*), if there is a Turing acceptor M such that for all $x \in \Gamma^*$

  $$x \in L \Rightarrow M \text{ accepts } x$$

  $$x \notin L \Rightarrow M \text{ rejects } x$$

  or, equivalently, if its characteristic function $\chi_L : \Gamma^* \to \{0, 1\}$ is Turing computable.

Theorem 1.6. A language $L \subseteq \Gamma^*$ is decidable if, and only if, L is r.e. and co-r.e.

Definition 1.7. Let $A \subseteq \Gamma^*, B \subseteq \Sigma^*$. We say that A is *(many-to-one) reducible* to B, $A \leq B$, if there is a total computable function $f : \Gamma^* \to \Sigma^*$ such that for all $x \in \Gamma^*$ we have $x \in A \Leftrightarrow f(x) \in B$.

Lemma 1.8.

- $A \leq B$, B decidable $\Rightarrow$ A decidable
- $A \leq B$, B r.e. $\Rightarrow$ A r.e.
- $A \leq B$, A undecidable $\Rightarrow$ B undecidable.

There surely are undecidable languages since there are only countably many Turing machines but uncountably many languages. Unfortunately, among these there are quite relevant classes of languages. For example we cannot decide whether a TM halts on a given input.

Definition 1.9 (Halting Problems)**.** The *general halting problem* is defined as

$$H := \{\rho(M)\#\rho(x) : M \text{ Turing machine}, x \in L(M)\}$$

where $\rho(M)$ and $\rho(x)$ are encodings of the TM M and the input x over a fixed alphabet $\{0,1\}$ such that the computation of M on x can be reconstructed from the encodings $\rho(M)$ and $\rho(x)$ in an effective way. This means that there is a universal TM U which, given $\rho(M)$ and $\rho(x)$, simulates the computation of M on x and halts if, and only if, M halts on x. Thus, $L(U) = H$ from which we conclude that H is r.e..

We introduce two special variants of the halting problem.

- *The self-application problem:* $H_0 := \{\rho(M) : \rho(M) \in L(M)\}$.
- *Halting on the empty word:* $H_\varepsilon := \{\rho(M) : \varepsilon \in L(M)\}$.

Theorem 1.10. H, H_0, and H_ε are undecidable.

Proof.

- H_0 is not co-r.e. and thus undecidable. Otherwise $\overline{H}_0 = L(M_0)$ for some TM M_0. Then

$$\rho(M_0) \in \overline{H}_0 \;\Leftrightarrow\; \rho(M_0) \in L(M_0) \;\Leftrightarrow\; \rho(M_0) \in H_0.$$

- H_0 is a special case of H, hence $H_0 \leq H$, and H is undecidable.
- We can reduce H to H_ε, thus H_ε is undecidable. Q.E.D.

We next establish the much more general result that in fact, no non-trivial semantic property of Turing machines can be decided algorithmically. In particular, for any fixed function, there is no algorithm that decides whether a given program computes precisely that function, i.e. we cannot algorithmically prove the correctness of a program. Note that this does not mean that we cannot prove the correctness of a single given program. Instead the statement is that we cannot do so algorithmically for all programs.

Theorem 1.11 (Rice)**.** Let $\mathcal{R}$ be the set of all computable functions and let $S \subseteq \mathcal{R}$ be a set of computable functions such that $S \neq \emptyset$ and $S \neq \mathcal{R}$. Then $\text{code}(S) := \{\rho(M) : f_M \in S\}$ is undecidable.

Proof. Let $\Uparrow$ be the everywhere undefined function, with domain $\text{Def}(\Uparrow) = \emptyset$. Obviously, $\Uparrow$ is computable. Assume that $\Uparrow \notin S$ (otherwise consider $\mathcal{R} \setminus S$ instead of S. Clearly if $\text{code}(\mathcal{R} \setminus S)$ is undecidable then so is $\text{code}(S)$.)

As $S \neq \emptyset$, there exists a function $f \in S$. Let M_f be a TM that computes f, i.e. $f_{M_f} = f$. We define a reduction $H_\varepsilon \leq \text{code}(S)$ by describing a total computable function $\rho(M) \mapsto \rho(M')$ such that

$$M \text{ halts on } \varepsilon \Leftrightarrow f_{M'} \in S.$$

Specifically, given $\rho(M)$, we construct the encoding of a TM M' which, given an input x, proceeds as follows:

- first simulate M on ε (i.e. apply the universal TM U to $\rho(M)\#\varepsilon$);
- then simulate M_f on x (i.e. apply the universal TM U to $\rho(M_f)\#\rho(x)$).

It is clear that the reduction function is computable. Furthermore, if M halts on ε then $f_{M'}(x) = f(x)$ for all inputs x, i.e. $f_{M'} = f$, so $f_{M'} \in S$. If M does not halt on ε then M' does not halt on x for any x, i.e. $f_{M'} = \Uparrow$, so $f_{M'} \notin S$. Q.E.D.

Definition 1.12 (Recursive inseparability)**.** Let $A, B \subseteq \Gamma^*$ be two disjoint

sets. We say that A and B are *recursively inseparable* if there exists no decidable set $C \subseteq \Gamma^*$ such that $A \subseteq C$ and $B \cap C = \emptyset$.
Example. $(A, \overline{A})$ are recursively inseparable if, and only if, A is undecidable.
Lemma 1.13. Let $A, B \subseteq \Gamma^*, A \cap B = \emptyset$ be recursively inseparable. Let $X, Y \subseteq \Sigma^*, X \cap Y = \emptyset$, and let f be a total computable function such that $f(A) \subseteq X$ and $f(B) \subseteq Y$. Then X and Y are recursively inseparable.

Proof. Assume there exists a decidable set $Z \subseteq \Sigma^*$ such that $X \subseteq Z$ and $Y \cap Z = \emptyset$. Consider $C = \{x \in \Gamma^* : f(x) \in Z\}$. C is decidable, $A \subseteq C, B \cap C = \emptyset$, thus C separates A, B. Q.E.D.

Notation: We write $(A, B) \leq (X, Y)$ if such a function f exists.
Example. $(A, \overline{A}) \leq (B, \overline{B}) \Leftrightarrow A \leq B$.

As a preparation for Trakhtenbrot's Theorem, we consider the following refinements of H_ε:

$$H_\varepsilon^+ := \{\rho(M) : M \text{ accepts } \varepsilon\}$$
$$H_\varepsilon^- := \{\rho(M) : M \text{ rejects } \varepsilon\}$$
$$H_\varepsilon^\infty := \{\rho(M) : \text{the computation of } M \text{ on } \varepsilon \text{ is infinite and does not cycle.}\}$$

H_0^+, H_0^-, H_0^∞ are defined analogously, with respect to self-application.
Theorem 1.14. $H_\varepsilon^+, H_\varepsilon^-$ and H_ε^∞ are pairwise recursively inseparable.

Proof. $(H_\varepsilon^+, H_\varepsilon^\infty)$: We show that every set C with $H_\varepsilon^+ \subseteq C$ and $H_\varepsilon^\infty \cap C = \emptyset$ is undecidable by reducing the halting problem H_ε to C. Define a reduction $\rho(M) \mapsto \rho(M')$ as follows. From a given code $\rho(M)$ construct the code of a TM M' that simulates M and simultaneously counts the number of computation steps since the start. If M halts (accepting or rejecting), M' accepts.

It is clear that the reduction function is computable. If M halts on ε then M' halts on ε as well and accepts, so $\rho(M') \in H_\varepsilon^+ \subseteq C$. If M does not halt on ε then M' does not halt either, and never cycles, so $\rho(M') \in H_\varepsilon^\infty$ and as $H_\varepsilon^\infty \cap C = \emptyset$, we have $\rho(M') \notin C$.

The statement for H_ε^- and H_ε^∞ is proven analogously.

$(H_\varepsilon^-, H_\varepsilon^+)$: Show that $(H_0^-, H_0^+) \leq (H_\varepsilon^-, H_\varepsilon^+)$ and that (H_0^-, H_0^+) are recursively inseparable.

- $(H_0^-, H_0^+) \leq (H_\varepsilon^-, H_\varepsilon^+)$:
  For a given input TM M construct a TM M' that ignores its own input and simulates M on $\rho(M)$. Obviously, M' can be constructed effectively, say by a computable function h. Now $h(M)$ accepts ε iff M accepts $\rho(M)$ and $h(M)$ rejects ε iff M rejects $\rho(M)$.
- (H_0^-, H_0^+) recursively inseparable:
  Assume there exists a decidable C with $H_0^- \subseteq C$ and $H_0^+ \subseteq \overline{C}$. Consider a machine M_0 that decides C. There are two cases:
  (1) M_0 accepts $\rho(M_0)$. Then $\rho(M_0) \in C$ by definition of M_0. Then $\rho(M_0) \notin H_0^+$ by definition of C. On the other hand, if M_0 accepts $\rho(M_0)$ then $\rho(M_0) \in H_0^+$ (by definition of H_0^+), a contradiction.
  (2) M_0 rejects $\rho(M_0)$. Then $\rho(M_0) \notin C$ by definition of M_0. Then $\rho(M_0) \notin H_0^-$ by definition of C. On the other hand, if M_0 rejects $\rho(M_0)$ then $\rho(M_0) \in H_0^-$ (by definition of H_0^-), a contradiction.

Q.E.D.

1.2 Trakhtenbrot's Theorem

In the following, we consider FO, more precisely first-order logic with equality. We restrict ourselves to a countable signature

$$\tau_\infty := \{R_j^i : i, j \in \mathbb{N}\} \cup \{f_j^i : i, j \in \mathbb{N}\}$$

where each R_j^i is a relation symbol of arity i and each f_j^i is a function symbol of arity i. We write formulae in $\mathrm{FO}(\tau_\infty)$ as words over the fixed finite alphabet

$$\Gamma := \{R, f, x, 0, 1, [,]\} \cup \{=, \neg, \wedge, \vee, \to, \leftrightarrow, \exists, \forall, ., (,)\},$$

using the following encoding of relation symbols, function symbols, and variables:

relation symbols:	R^i_j	$\longmapsto$	$R[\text{bin } i][\text{bin } j]$
function symbols:	f^i_j	$\longmapsto$	$f[\text{bin } i][\text{bin } j]$
variables:	x_j	$\longmapsto$	$x[\text{bin } j]$.

In this way, every formula $\varphi \in \text{FO}$ can be viewed as a word in Γ^*.

Let $X \subseteq \text{FO}$ be a class of formulae. We analyse the following decision problems:

$$\begin{aligned}
\mathit{Sat}(X) &:= \{\psi \in X : \psi \text{ has a model}\} \\
\mathit{Fin\text{-}Sat}(X) &:= \{\psi \in X : \psi \text{ has a finite model}\} \\
\mathit{Val}(X) &:= \{\psi \in X : \psi \text{ is valid}\} \\
\mathit{Non\text{-}Sat}(X) &:= X \setminus \mathit{Sat}(X) \\
\mathit{Inf\text{-}Axioms}(X) &:= \mathit{Sat}(X) \setminus \mathit{Fin\text{-}Sat}(X) \\
&= \{\psi \in X : \psi \text{ is an infinity axiom, i.e. } \psi \text{ has a} \\
&\qquad\qquad \text{model but no finite model}\}.
\end{aligned}$$

Theorem 1.15. Let $X \subseteq \text{FO}$ be decidable. Then

(1) $\mathit{Val}(X)$ is r.e.
(2) $\mathit{Non\text{-}Sat}(X)$ is r.e.
(3) $\mathit{Sat}(X)$ is co-r.e.
(4) $\mathit{Fin\text{-}Sat}(X)$ is r.e.
(5) $\mathit{Inf\text{-}Axioms}(X)$ is co-r.e.

Proof. (1) φ is valid $\Leftrightarrow \ \vdash \varphi$ (Completeness Theorem). Thus we can systematically enumerate all proofs and halt if a proof for φ is listed.
(2) φ valid $\Leftrightarrow \neg\varphi$ is not satisfiable.
(3) Follows from Item (2).
(4) Systematically generate all finite models and halt if a model of φ is found.
(5) $\text{FO} \setminus \mathit{Inf\text{-}Axioms}(X) = \mathit{Non\text{-}Sat}(X) \cup \mathit{Fin\text{-}Sat}(X)$ is r.e. Q.E.D.

Definition 1.16. A class $X \subseteq \text{FO}$ has the *finite model property* (FMP) if every satisfiable $\varphi \in X$ has a finite model, i.e. if $\mathit{Sat}(X) = \mathit{Fin\text{-}Sat}(X)$.

Theorem 1.17. Suppose that $X \subseteq \text{FO}$ is decidable and that X has the FMP. Then $\mathit{Sat}(X)$ is decidable.

Proof. $Sat(X)$ is co-r.e. and since $Sat(X) = Fin\text{-}Sat(X)$ and $Fin\text{-}Sat(X)$ is r.e. also $Sat(X)$ is r.e. Thus $Sat(X)$ is decidable. Q.E.D.

In this case also $Fin\text{-}Sat(X)$, $Non\text{-}Sat(X)$, $Val(X)$ are decidable and of course $Inf\text{-}Axioms(X) = \emptyset$ is decidable.

Theorem 1.18 (Trakhtenbrot)**.** There is a finite vocabulary $\tau \subseteq \tau_\infty$ such that $Fin\text{-}Sat(\mathrm{FO}(\tau)), Non\text{-}Sat(\mathrm{FO}(\tau))$ and $Inf\text{-}Axioms(\mathrm{FO}(\tau))$ are pairwise recursively inseparable and therefore undecidable.

The proof of Trakhtenbrot's theorem introduces a proof strategy that can be applied in many other undecidability proofs. (Do not focus on the technicalities but on the general idea to construct the reduction formulae.)

Proof. Let M be a deterministic Turing acceptor. We show that there is an effective reduction $\rho(M) \mapsto \psi_M$ such that

(1) M accepts $\varepsilon \implies \psi_M$ has a finite model.
(2) M rejects $\varepsilon \implies \psi_M$ is unsatisfiable.
(3) The computation of M on ε is infinite and non-periodic $\implies \psi_M$ is an infinity axiom.

Then the theorem follows by Lemma 1.13.

Let M be a Turing acceptor with states $Q = \{q_0, \ldots, q_r\}$, initial state q_0, alphabet $\Sigma = \{a_0, \ldots, a_s\}$ (where $a_0 = \square$), final states $F = F^+ \cup F^-$ and transition function δ.

ψ_M is defined over the vocabulary $\tau = \{0, f, q, p, w\}$ where 0 is a constant, f, q, p are unary functions and w is a binary function. Define the term $\boldsymbol{k}$ as $f^k 0$.

By constructing a formula we intend to have a model $\mathfrak{A}_M = (A, 0, f, q, p, w)$ describing a run of M on the input ε where

- universe $A = \{0, 1, 2, \ldots, n\}$ or $A = \mathbb{N}$;
- $f(t) = t + 1$ if $t + 1 \in A$ and $f(t) = t$, if t is the last element of A;
- $q(t) = i$ iff M is at time t in state q_i;
- $p(t)$ is the head position of M at time t;
- $w(s, t) = i$ iff symbol a_i is at time t on tape-cell s.

Note that we cannot enforce this model, but if ψ_M is satisfiable this one will be among its models.

$$\psi_M := \text{ START } \wedge \text{ COMPUTE } \wedge \text{ END}$$

$$\text{START } := (q0 = 0 \wedge p0 = 0 \wedge \forall x\, w(x,0) = 0).$$

[Enforces input configuration on ε at time 0]

$$\text{COMPUTE } := \text{NOCHANGE} \wedge \text{ CHANGE}$$

$$\text{NOCHANGE} := \forall x \forall y (py \neq x \to w(x, fy) = w(x,y))$$

[content of currently not visited tape cells does not change]

$$\text{CHANGE} := \bigwedge_{\delta:(q_i,a_j)\mapsto(q_k,a_\ell,m)} \forall y(\alpha_{i,j} \to \beta_{k,\ell,m})$$

where

$$\alpha_{ij} := (qy = \boldsymbol{i} \wedge w(py, y) = \boldsymbol{j})$$

[M is at time y in state q_i and reads the symbol a_j]

$$\beta_{k,\ell,m} := (qfy = \boldsymbol{k} \wedge w(py, fy) = \boldsymbol{\ell} \wedge \text{MOVE}_m)$$

and

$$\text{MOVE}_m := \begin{cases} pfy = py & \text{if } m = 0 \\ pfy = fpy & \text{if } m = 1 \\ \exists z(fz = py \wedge pfy = z) & \text{if } m = -1. \end{cases}$$

$$\text{END } := \bigwedge_{\substack{\delta(q_i,a_j) \text{ undef.} \\ q_i \notin F^+}} \forall y \neg \alpha_{ij}$$

[The only way the computation ends is in an accepting state]

Remark 1.19.

- $\rho(M) \mapsto \psi_M$ is an effective construction.
- If M accepts ε, the intended model is finite and is indeed a model $\mathfrak{A}_M \models \psi_M$, thus $\psi_M \in \mathit{Fin\text{-}Sat}(FO(\tau))$.
- If the computation of M on ε is infinite, the intended model is infinite and $\mathfrak{A}_M \models \psi_M$.

It remains to show that if M rejects ε, then ψ_M is unsatisfiable, and if the computation of M on ε is infinite and aperiodic, then ψ_M is an infinity axiom.

Suppose $\mathfrak{B} = (B, 0, f, q, p, w) \models \psi_M$.

Definition 1.20. $\mathfrak{B}$ *enforces* at time t the configuration (q_i, j, w) with $w = a_{i_0} \dots a_{i_m} \in \Sigma^*$ if

(1) $\mathfrak{B} \models q\boldsymbol{t} = \boldsymbol{i}$,
(2) $\mathfrak{B} \models p\boldsymbol{t} = \boldsymbol{j}$,
(3) for all $k \leq m$, $\mathfrak{B} \models w(\boldsymbol{k}, \boldsymbol{t}) = \boldsymbol{i_k}$ and for all $k > m$, $\mathfrak{B} \models w(\boldsymbol{k}, \boldsymbol{t}) = 0$.

Since $\mathfrak{B} \models \psi_M$, the following holds:

- $\mathfrak{B}$ enforces $C_0 = (q_0, 0, \varepsilon)$ at time 0 (since $\mathfrak{B} \models$ START.)
- If $\mathfrak{B}$ enforces at time t a non-final configuration C_t, then $\mathfrak{B}$ enforces the configuration $C_{t+1} = \mathrm{Next}(\mathrm{C_t})$ at time $t+1$.
- Especially, the computation of M cannot reach a rejecting configuration. It follows that if M rejects ε, then ψ_M is unsatisfiable.
  Consider an infinite and aperiodic computation of M, and assume $\mathfrak{B} \models \psi_M$ is finite. Since $\mathfrak{B}$ is finite, it enforces a periodic computation in contradiction to the assumption that the computation of M is aperiodic.

$$C_0 \vdash \dots \underbrace{\vdash C_r \vdash \dots \vdash C_{t-1}}$$

We have shown:

- If M accepts ε, then ψ_M has a finite model.
- If M rejects ε, then ψ_M is unsatisfiable.
- If the computation of M is infinite and aperiodic, then ψ_M is an infinity axiom. Q.E.D.

We now know that the sets of all finitely satisfiable, all unsatisfiable and all only infinitely satisfiable formulae are undecidable for $\mathrm{FO}(\tau)$ where τ consists of only three unary functions and one binary function. This raises a number of questions.

(1) For which other vocabularies σ do we have similar undecidability results for $\mathrm{FO}(\sigma)$?

(2) For which σ is satisfiability of $\mathrm{FO}(\sigma)$ decidable?
(3) Is there a complete classification? In this case, we want to find minimal vocabularies σ such that the above problems are undecidable, i.e. vocabularies such that any further restriction yields a class of formulae for which satisfiability is decidable.

We first define what it means that a fragment of FO is as hard for satisfiability as the whole FO.

Definition 1.21. $X \subseteq \mathrm{FO}$ is a *reduction class* if there exists a computable function $f : \mathrm{FO} \to X$ such that $\psi \in \mathit{Sat}(\mathrm{FO}) \Leftrightarrow f(\psi) \in \mathit{Sat}(X)$.

Let $X, Y \subseteq \mathrm{FO}$. A *conservative reduction of X to Y* is a computable function $f : X \to Y$ with

- $\psi \in \mathit{Sat}(X) \Leftrightarrow f(\psi) \in \mathit{Sat}(Y)$, and
- $\psi \in \mathit{Fin\text{-}Sat}(X) \Leftrightarrow f(\psi) \in \mathit{Fin\text{-}Sat}(Y)$.

X is a *conservative reduction class* if there exists a conservative reduction of FO to X.

Corollary 1.22. Let X be a conservative reduction class. Then $\mathit{Fin\text{-}Sat}(X)$, $\mathit{Inf\text{-}Axioms}(X)$ and $\mathit{Non\text{-}Sat}(X)$ are pairwise recursively inseparable, and thus $\mathit{Fin\text{-}Sat}(X)$, $\mathit{Sat}(X)$, $\mathit{Val}(X)$, $\mathit{Non\text{-}Sat}(X)$, $\mathit{Inf\text{-}Axioms}(X)$ are undecidable.

Proof. A conservative reduction from FO to X yields a uniform reduction from $\mathit{Fin\text{-}Sat}(\mathrm{FO})$, $\mathit{Inf\text{-}Axioms}(\mathrm{FO})$ and $\mathit{Non\text{-}Sat}(\mathrm{FO})$ to $\mathit{Fin\text{-}Sat}(X)$, $\mathit{Inf\text{-}Axioms}(X)$ and $\mathit{Non\text{-}Sat}(X)$, respectively. Q.E.D.

It is indeed possible to give a complete classification of those vocabularies σ such that $\mathrm{FO}(\sigma)$ is decidable.

Theorem 1.23. If $\sigma \subseteq \{P_0, P_1, \ldots\} \cup \{f\}$ consists of at most one unary function f and an arbitrary number of monadic predicates $P_0, P_1, \ldots$, then $\mathit{Sat}(\mathrm{FO}(\sigma))$ is decidable. In all other cases, $\mathit{Sat}(\mathrm{FO}(\sigma))$, $\mathit{Inf\text{-}Axioms}(\mathrm{FO}(\sigma))$ and $\mathit{Non\text{-}Sat}(\mathrm{FO}(\sigma))$ are pairwise recursively inseparable, and $\mathrm{FO}(\sigma)$ is a conservative reduction class.

A full proof of this classification theorem is rather difficult. In particular, the decidability of the monadic theory of one unary function, which implies the decidability part, is a difficult theorem due to Rabin.

On the other side, one has to show that Trakhtenbrot's theorem applies to the vocabularies

$\tau_1 = \{E\}$ where E is a binary relation,
$\tau_2 = \{f, g\}$ where f, g are unary functions,
$\tau_3 = \{F\}$ where F is a binary function,

and hence also to all extensions of τ_1, τ_2, τ_3.

Of course, one may also look at other syntactic restrictions besides restricting the vocabulary. One possibility is to restrict the number of variables. This is only interesting for relational formulae. If we have functions, satisfiability is undecidable even for formulae with only one variable, as we shall see later.

Define FO^k as first-order logic with relational symbols only and a fixed collection of k variables, say $x_1, \ldots, x_k$.

Theorem 1.24.

- FO^2 has the finite model property and is decidable (see Sect. 1.6).
- FO^3 is a conservative reduction class.

A further important possibility is to restrict the structure of quantifier prefixes of formulae in prenex normal form, and to combine this with restrictions on the vocabulary, and the presence or absence of equality. This leads to the notion of a *prefix-vocabulary class* in first-order logic, and indeed, also for these fragments of FO there is a complete classification of those with a solvable satisfiability problem, and those that are conservative reduction classes.

A full description of this classification exceeds the scope of this course by far (see E. Börger, E. Grädel, and Y. Gurevich, The Classical Decision Problem, 1997). Instead we shall present some of the fundamental methods for establishing such results, and illustrate these with applications to specific fragments of first-order logic.

1.3 Domino problems

Domino problems are a simple and yet general tool for proving undecidability results (and lower bounds in complexity theory) without the need of explicit encodings of Turing machine computations.

The informal idea is the following: a domino problem is given by a finite set of dominoes or tiles, each of them an oriented unit square with coloured edges; the question is whether it is possible to cover the first quadrant in the Cartesian plane by copies of these tiles, without holes and overlaps, such that adjacent dominoes have matching colours on their common edge. The set of tiles is finite, but there are infinitely many copies of each tile available; rotation of the tiles is not allowed. Variants of this problem require a tiling of a different geometric object (a finite square, a rectangle, or a torus) and/or that certain places (e.g. the origin, the bottom row or the diagonal) are tiled by specific tiles.

Here is a more abstract defintion.

Definition 1.25. A *domino system* is a structure $\mathcal{D} = (D, H, V)$ with

- a finite set D (of dominoes),
- horizontal and vertical compatibility relations $H, V \subseteq D \times D$.

The intuitive meaning of H and V is that

- $(d, d') \in H$ if the right colour of d is equal to the left colour of d',
- $(d, d') \in V$ if the top colour of d is equal to the bottom colour of d' (see Figure 1.1).

A *tiling* of $\mathbb{N} \times \mathbb{N}$ by $\mathcal{D}$ is a function $t : \mathbb{N} \times \mathbb{N} \to D$ such that for all $x, y \in \mathbb{N}$

- $(t(x, y), t(x + 1, y)) \in H$ and
- $(t(x, y), t(x, y + 1)) \in V$.

A periodic tiling of $\mathbb{N} \times \mathbb{N}$ by $\mathcal{D}$ is a tiling t for which there exist two integers $h, v \in \mathbb{N}$ such that $t(x, y) = t(x + h, y) = t(x, y + v)$ for all $x, y \in \mathbb{N}$.

The decision problem DOMINO is described as

$$\text{DOMINO} := \{\mathcal{D} : \text{there exists a tiling of } \mathbb{N} \times \mathbb{N} \text{ by } \mathcal{D}\}$$

Theorem 1.26 (Berger, Robinson)**.** DOMINO is co-r.e. and undecidable.

In this general form, this is quite a difficult result. A simpler variant is the so-called origin-constrained domino problem, that requires that a specific domino must be placed at the point $(0, 0)$. With this requirement, it is straightforward to encode Turing machine computations by domino

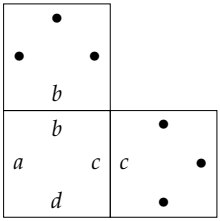


Figure 1.1. Domino adjacency condition

tilings (successive rows of the tiling correspond to successive configurations in the computation), and thus to reduce halting problems to tiling problems for domino systems. The origin constraint is used to encode the beginning of the computation (and to avoid that the entire space can be tiled by a domino corresponding to the blank symbol) Without an origin constraint, the problem is more difficult to handle; an essential part of the proof is the construction of a set of dominoes that admits only non-periodic tilings.

There are several extensions and variations of this result.

Theorem 1.27. A domino system $\mathcal{D}$ admits a tiling of $\mathbb{Z} \times \mathbb{Z}$ if, and only if, it admits a tiling of $\mathbb{N} \times \mathbb{N}$.

Proof. It is clear that a tiling of $\mathbb{Z} \times \mathbb{Z}$ also gives a tiling of $\mathbb{N} \times \mathbb{N}$. The converse is a nice application of König's Lemma. Suppose that t is a tiling of $\mathbb{N} \times \mathbb{N}$ by $\mathcal{D}$. There exists at least one domino d such that for all n there exist $i, j > n$ with $t(i, j) = d$. Fix such a d. Further, for every $k \in \mathbb{N}$, let S_k be the square $\{-k, \ldots, -1, 0, 1, \ldots, k\} \times \{-k, \ldots, -1, 0, 1, \ldots, k\}$.

We define a finitely branching tree whose nodes are the correct tilings t_k of S_k by $\mathcal{D}$ such that $t_k(0, 0) = d$. The root is the unique such tiling of S_0 and the children of a tiling t_k are the possible extensions to tilings t_{k+1} of S_{k+1}. This tree contains paths of any finite length. By König's Lemma it also contains an infinite path from the root, which means that $\mathcal{D}$ admits a tiling of $\mathbb{Z} \times \mathbb{Z}$. Q.E.D.

The undecidability result from Theorem 1.26 can be strengthened to a recursive inseparability result.

Theorem 1.28. The set of domino systems admitting a periodic tiling of $\mathbb{N} \times \mathbb{N}$, those that admit no tiling of $\mathbb{N} \times \mathbb{N}$ and those that admit a tiling but not a periodic one are pairwise recursively inseparable.

The proof of Theorem 1.28 reduces the halting problems $H_\varepsilon^+, H_\varepsilon^-, H_\varepsilon^\infty$, to the domino problems. There exists a recursive function that associates with every TM M a domino system $\mathcal{D}$ satisfying

- If $M \in H_\varepsilon^+$ then $\mathcal{D}$ admits a periodic tiling of $\mathbb{N} \times \mathbb{N}$.
- If $M \in H_\varepsilon^-$ then $\mathcal{D}$ admits no tiling of $\mathbb{N} \times \mathbb{N}$.
- If $M \in H_\varepsilon^\infty$ then $\mathcal{D}$ admits a tiling of $\mathbb{N} \times \mathbb{N}$ but no periodic one.

Definition 1.29. A computable function f is a *conservative reduction from domino systems to X* if, for all domino systems $\mathcal{D}$, $f(\mathcal{D}) = \varphi_\mathcal{D}$ is in X and the following holds:

- $\mathcal{D}$ admits a periodic tiling of $\mathbb{N} \times \mathbb{N} \Rightarrow \psi_\mathcal{D}$ has a finite model
- $\mathcal{D}$ admits no tiling of $\mathbb{N} \times \mathbb{N} \Rightarrow \psi_\mathcal{D}$ is unsatisfiable
- $\mathcal{D}$ admits a tiling of $\mathbb{N} \times \mathbb{N}$ but no periodic one $\Rightarrow \psi_\mathcal{D}$ is an infinity axiom.

Proposition 1.30. Let $X \in$ FO. If there exists a conservative reduction from domino systems to X then X is a conservative reduction class.

Proof. Since *Fin-Sat*(FO) and *Non-Sat*(FO) are recursively enumerable and *Inf-Axioms*(FO) is co-recursively enumerable, we can associate with every first-order formula ψ a Turing machine M such that

- $\psi \in$ *Fin-Sat*(FO) $\Rightarrow \rho(M) \in H_\varepsilon^+$,
- $\psi \in$ *Non-Sat*(FO) $\Rightarrow \rho(M) \in H_\varepsilon^-$,
- $\psi \in$ *Inf-Axioms*(FO) $\Rightarrow \rho(M) \in H_\varepsilon^\infty$.

According to the assumption, there is a reduction $\mathcal{D} \mapsto \varphi_\mathcal{D}$ from domino systems to X. Thus, the domino method yields a conservative reduction from FO to X.

Q.E.D.

1.4 Applications of the domino method

We now apply the domino method to obtain several reduction classes.

The Kahr-Moore-Wang class KMW is the class of all first-order sentences of form $\forall x \exists y \forall z \varphi$, where φ is a quantifier-free formula without equality, whose vocabulary contains only binary relation symbols.

Theorem 1.31. The Kahr-Moore-Wang class is a conservative reduction class.

Proof. It suffices to construct a conservative reduction from domino systems to KMW, i.e., a mapping $\mathcal{D} \mapsto \psi_{\mathcal{D}}$ over a vocabulary consisting of binary relation symbols $(P_d)_{d \in D}$ such that

(1) $\mathcal{D}$ admits a periodic tiling of $\mathbb{N} \times \mathbb{N} \Rightarrow \psi_{\mathcal{D}}$ has a finite model
(2) $\mathcal{D}$ admits no tiling of $\mathbb{N} \times \mathbb{N} \Rightarrow \psi_{\mathcal{D}}$ is unsatisfiable
(3) $\mathcal{D}$ admits a tiling of $\mathbb{N} \times \mathbb{N}$ but no periodic one $\Rightarrow \psi_{\mathcal{D}}$ is an infinity axiom

For a tiling $t : \mathbb{N} \times \mathbb{N} \to D$, an intended model of $\psi_{\mathcal{D}}$ is $\mathbb{N}$ with the interpretation $P_d = \{(i,j) \in \mathbb{N} \times \mathbb{N} : t(i,j) = d\}$ for all $d \in D$. We define $\psi_{\mathcal{D}}$ by

$$\psi_{\mathcal{D}} := \forall x \exists y \forall z \Big(\bigwedge_{d \neq d'} P_d xz \to \neg P_{d'} xz \\ \wedge \bigvee_{(d,d') \in H} (P_d xz \wedge P_{d'} yz) \wedge \bigvee_{(d,d') \in V} (P_d zx \wedge P_{d'} zy) \Big).$$

Obviously $\psi_{\mathcal{D}}$ is of the desired format, i.e. $\psi_{\mathcal{D}} \in$ KMW.

(1) Suppose that $\mathcal{D}$ admits a periodic tiling t of $\mathbb{N} \times \mathbb{N}$, such that $t(x,y) = t(x+h,y) = t(x,y+v)$ for all x, y. We construct a finite model of $\psi_{\mathcal{D}}$ as follows. Let $m := lcm(h,v)$ be the least common multiple of h and v. Then t induces a tiling

$$t' : \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z} \to D$$

with $t'(x,y) = t(x(\mod m), y(\mod m))$.

It follows that $\mathfrak{A} = (\mathbb{Z}/m\mathbb{Z}, (P_d)_{d \in D})$ with $P_d = \{(i,j) : t'(i,j) = d\}$ is a finite model for $\psi_{\mathcal{D}}$ (for x in $\mathbb{Z}/m\mathbb{Z}$ choose $y := x + 1 \pmod m$).

(2) By analogous arguments, it follows, that whenever $\mathcal{D}$ admits a tiling of $\mathbb{N} \times \mathbb{N}$, then $\psi_{\mathcal{D}}$ has a model over $\mathbb{N}$.

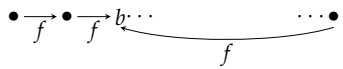
(3) Finally we prove that if $\psi_{\mathcal{D}}$ has a model, then $\mathcal{D}$ admits a tiling of $\mathbb{N} \times \mathbb{N}$, and if that model is finite, we even obtain a periodic tiling.

Consider the Skolem normal form $\varphi_{\mathcal{D}}$ of $\psi_{\mathcal{D}}$:

$$\varphi_{\mathcal{D}} := \forall x \forall z (\bigwedge_{d \neq d'} P_d xz \to \neg P_{d'} xz \\ \wedge \bigvee_{(d,d') \in H} (P_d xz \wedge P_{d'} fxz) \wedge \bigvee_{(d,d') \in V} (P_d zx \wedge P_{d'} zfx).$$

If $\psi_{\mathcal{D}}$ is satisfiable, then also $\varphi_{\mathcal{D}}$ has a model $\mathfrak{B} = (B, f, (P_d)_{d \in D})$. Define a tiling $t : \mathbb{N} \times \mathbb{N} \to D$ as follows: choose any $b \in B$, and for all $i, j \in \mathbb{N}$, set $t(i,j) := d$ for the unique $d \in D$ such that $\mathfrak{B} \models P_d(f^i b, f^j b)$. Since $\mathfrak{B} \models \varphi_{\mathcal{D}}$, it follows that t is a correct tiling.

Now suppose that $\mathfrak{B} \models \varphi_{\mathcal{D}}$ is finite.



Choose $b \in B$ such that, for some $n \geq 1$, $f^n b = b$. Then the defined tiling t is periodic. Q.E.D.

Corollary 1.32. FO^3 is a conservative reduction class.

Later we shall prove that FO^2 has the FMP.

Consider now formula classes $X \subseteq$ FO over functional vocabularies. One can prove that FO(τ) is a conservative reduction class if τ contains

- two unary functions or
- one binary function.

This is even true for sentences of the form $\forall x \varphi$ where φ is quantifier-free.

We stablish, again via a conservative reduction from domino problems, a weaker result from which the above mentioned ones can be obtained by interpretation arguments (see exercises).

Theorem 1.33. The class $\mathcal{F}$, consisting of all sentences $\forall x \varphi$ where φ is a quantifier-free formula whose vocabulary consists only of unary function symbols, is a conservative reduction classes.

Proof. We define a conservative reduction $\mathcal{D} = (D, H, V) \mapsto \psi_{\mathcal{D}}$ where $\psi_{\mathcal{D}} \in \mathcal{F}$ has the vocabulary $\{f, g, (h_d)_{d \in D}\}$ where all function symbols

are unary. The intended model is $\mathbb{N} \times \mathbb{N}$ with successor functions f and g. The subformula $\forall x(fgx = gfx)$ ensures that the models of $\psi_{\mathcal{D}}$ contain a two-dimensional grid. The fact that a position x is tiled by $d \in D$ is expressed by requiring that $h_d x = x$, i.e. that x is a fixed point of h_d.

$$\begin{aligned}\psi_{\mathcal{D}} := \forall x\big(fgx = gfx \wedge \bigwedge_{d \neq d'} (h_d x = x \to h_{d'} x \neq x) \\ \wedge \bigvee_{(d,d') \in H} (h_d x = x \wedge h_{d'} fx = fx) \\ \wedge \bigvee_{(d,d') \in V} (h_d x = x \wedge h_{d'} gx = gx)\big)\,.\end{aligned}$$

We claim that there exists a tiling $t : \mathbb{N} \times \mathbb{N} \to \mathcal{D}$ if and only if $\psi_{\mathcal{D}}$ is satisfiable.

" $\Rightarrow$ " Assume that t is a correct tiling. Construct the (intended) model $\mathfrak{A} = (\mathbb{N} \times \mathbb{N}, f, g, (h_d)_{d \in \mathcal{D}})$ with

- $f(i,j) = (i+1, j)$,
- $g(i,j) = (i, j+1)$,
- $h_d(i,j) \begin{cases} = (i,j) & \text{if } t(i,j) = d \\ \neq (i,j) & \text{otherwise.} \end{cases}$

Clearly $\mathfrak{A} \models \psi_{\mathcal{D}}$.

" $\Leftarrow$ " Consider $\mathfrak{B} = (B, f, g, (h_d)_{d \in \mathcal{D}}) \models \psi_{\mathcal{D}}$.

Choose an arbitrary $b \in B$ and define $t : \mathbb{N} \times \mathbb{N} \to D$ by

$$t(i,j) := d \text{ iff } \mathfrak{B} \models h_d f^i g^j b = f^i g^j b.$$

Note that every point in B is a fixed-point of exactly one of the functions h_d, and t is well-defined and a a correct tiling. Further, if $\mathfrak{B}$ is finite, then σ is periodic, and thus the reduction is conservative.

Q.E.D.

Exercise 1.1. Prove that the more restricted class $\mathcal{F}_2 \subseteq \mathcal{F}$ consisting of sentences in $\mathcal{F}$ that contain just two unary function symbols, is also a conservative reduction class.

Hint: Transform sentences $\forall x\varphi$ with unary function symbols $f_1, \ldots, f_m$ into sentences $\forall x\tilde{\varphi} := \forall x\varphi[x/hx, f_i/hg^i]$ where h, g are fresh unary function symbols.

1.5 The finite model property

We study the finite model property (FMP) for fragments of FO as a mean to show that these fragments are decidable, and also to better understand their expressive power and algorithmic complexity.

Recall that a class $X \subseteq$ FO has the *finite model property* if $Sat(X) = Fin\text{-}Sat(X)$. Since for any decidable class X, $Fin\text{-}Sat(X)$ is r.e. and $Sat(X)$ is co-r.e., it follows that $Sat(X)$ is decidable if X has the FMP. In many cases, the proof that a class has the finite model property provides a bound on the model's cardinality, and thus a complexity bound for the satisfiability problem. To prove completeness for complexity classes we make use of a bounded variant of the domino problem.

We shall illustrate the power of this method by a few examples.

Definition 1.34. The *atomic k-type* of $a_1, \ldots, a_k$ in $\mathfrak{A}$ is defined as

$$\begin{aligned}\mathrm{atp}_{\mathfrak{A}}(a_1, \ldots, a_k) := \{\gamma(x_1 \ldots, x_k) : \gamma \text{ atomic formula or negated} \\ \text{atomic formula such that } \mathfrak{A} \models \gamma(a_1, \ldots, a_k)\}.\end{aligned}$$

In the examples that we consider here, the structures contain unary or binary relations only. Hence, to describe a structure it suffices to define its universe and to specify the atomic 1-types and 2-types for all of its elements.

Example 1.35. Let $\mathfrak{A}$ be the structure $(A, E_1, \ldots, E_m)$ where the E_i are binary relations. Then for $a \in A$:

$$\mathrm{atp}_{\mathfrak{A}}(a) = \{E_i xx : \mathfrak{A} \models E_i aa\} \cup \{\neg E_i xx : \mathfrak{A} \models \neg E_i aa\}.$$

The *monadic class* (also called the Löwenheim class) is the class of first-order sentences over a vocabulary the contains only unary predicates.

Theorem 1.36. The monadic class has the FMP.

Proof. Let $\mathfrak{A} = (A, P_1^{\mathfrak{A}}, \ldots, P_n^{\mathfrak{A}}) \models \varphi$ where $\mathrm{qr}(\varphi) = m$. For each sequence of bits $\alpha = \alpha_1 \ldots \alpha_n \in \{0,1\}^n$ we define $P_\alpha^{\mathfrak{A}} = Q_1 \cap Q_2 \cap \ldots \cap Q_n$, where $Q_i = P_i^{\mathfrak{A}}$ if $\alpha_i = 1$ and $Q_i = A \setminus P_i^{\mathfrak{A}}$ if $\alpha_i = 0$. Notice that the sets $P_\alpha^{\mathfrak{A}}$ define a partition of A, and that α completely describes the atomic 1-type of any $a \in P_\alpha^{\mathfrak{A}}$.

We construct $\mathfrak{B}$ by taking $\min(|P_\alpha^{\mathfrak{A}}|, m)$ elements into each $P_\alpha^{\mathfrak{B}}$. Observe that $\mathfrak{B}$ is completly specified in this way, with $P_i^{\mathfrak{B}} = \bigcup_{\alpha|\alpha_i=1} P_\alpha^{\mathfrak{B}}$). We show that $\mathfrak{A} \equiv_m \mathfrak{B}$ using the Ehrenfeucht-Fraïssé Theorem.

The following is a winning strategy for Duplicator in the Ehrenfeucht-Fraïssé game with m moves on $(\mathfrak{A}, \mathfrak{B})$: Answer any element chosen by Spoiler by an element with the same atomic type in the other structure, respecting equalities and inequalities with previously chosen elements. Due to the construction it is certainly possible to do that for m moves, so Duplicator wins the game. Hence $\mathfrak{A} \equiv_m \mathfrak{B}$, and therefore $\mathfrak{B} \models \varphi$. Q.E.D.

From the proof we see that the constructed finite model $\mathfrak{B}$ is in fact a submodel of the arbitrary model $\mathfrak{A}$ that we started with. Thus we have in fact established a stronger result than the finite model property, namely the *finite submodel property* of the monadic class: every infinite model of a sentence in the monadic class has a finite substructure which is also a model of that sentence.

In general it need not be the case that classes with the FMP also have the finite submodel property.

1.6 The two-variable fragment of FO

We denote relational first-order logic over k variables by FO^k, i.e.

$$\mathrm{FO}^k := \{\varphi \in \mathrm{FO} : \varphi \text{ relational, } \varphi \text{ only contains } k \text{ variables}\}.$$

We have shown that the Kahr-Moore-Wang class KMW, and hence also FO^3, are conservative reduction classes. We now prove that FO^2 has the finite model property and is thus decidable. Note that FO^k formulae are not necessarily in prenex normal form. A further motivation for the study of FO^2 is that propositional modal logic can be viewed as a frag-

ment of FO^2 (in fact ML can be proven to be precisely the bisimulation invariant fragment of FO^2).

Before we proceed to prove the finite model property for FO^2, as a first step we establish a normal form for formulae in FO^2.

Lemma 1.37 (Scott)**.** For each sentence $\psi \in \mathrm{FO}^2$ one can construct in polynomial time a sentence $\varphi \in \mathrm{FO}^2$ of the form

$$\varphi := \forall x \forall y \alpha \wedge \bigwedge_{i=1}^{n} \forall x \exists y \beta_i$$

such that $\alpha, \beta_1, \ldots, \beta_n$ are quantifier free and such that ψ and φ are satisfiable over the same universe. Moreover, we have $|\varphi| = \mathcal{O}(|\psi| \cdot \log |\psi|)$.

Proof. First of all, we can assume that formulae $\varphi \in \mathrm{FO}^2$ only contain unary and binary relation symbols. This is no restriction since relations of higher arity can be substituted by introducing new binary and unary relation symbols. For example, if R is a relation of arity three, one could add a unary relation R_x and three binary relations $R_{x,x,y}$, $R_{x,y,x}$ and $R_{x,y,y}$ and replace each atom $R(x,x,x)$ (or $R(y,y,y)$) by $R_x(x)$ (or $R_x(y)$) and atoms as $R(x,x,y)$ or $R(x,y,x)$ by $R_{x,x,y}(x,y)$ and $R_{x,y,x}(x,y)$ respectively. By adding appropriate new subformulae one can ensure that the semantics are preserved, i.e. that the newly introduced relations partition a ternary relation in the intended sense. For example we would introduce as a new subformula $\forall x(R_x(x) \leftrightarrow R_{x,x,y}(x,x))$.

With ψ containing at most binary relations, we iterate the following steps until ψ has the desired form. We choose a subformula $Qy\eta$ of ψ ($Q \in \{\forall, \exists\}, \eta$ quantifier free) and add a new unary relation R:

$$\begin{aligned} \psi' &:= \psi[Qy\eta / Rx] \\ \psi &\mapsto \psi' \wedge \forall x(Rx \leftrightarrow Qy\eta). \end{aligned}$$

R captures those x that satisfy $Qy\eta$. The resulting formula φ is not yet of the desired form, but it is equivalent to the following:

(a) if $Q = \exists$, then

$$\varphi \equiv \psi' \wedge \forall x \forall y(\eta \to Rx) \wedge \forall x \exists y(Rx \to \eta)$$

(b) else if $Q = \forall$, then

$$\varphi \equiv \psi' \wedge \forall x \forall y(Rx \to \eta) \wedge \forall x \exists y(\eta \to Rx)$$

Now use that conjunctions of $\forall\forall$-formulae are equivalent to a $\forall\forall$-formula and obtain $\psi \equiv \forall x \forall y \alpha \wedge \bigwedge_{i=1}^{n} \forall x \exists y \beta_i$. Q.E.D.

Theorem 1.38. FO^2 has the finite model property. In fact, every satisfiable formula $\psi \in \mathrm{FO}^2$ has a model with at most $2^{|\psi|}$ elements.

Proof. The proof strategy is as follows: we start with a model $\mathfrak{A}$ of ψ and proceed by constructing a new model $\mathfrak{B}$ of ψ such that $|\mathfrak{B}| \leq 2^{\mathcal{O}(|\psi|)}$. For the construction the following definitions will be essential.

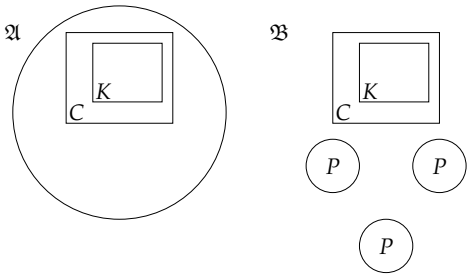
An element $a \in A$ is said to be a *king of* $\mathfrak{A}$ if its atomic 1-type is unique in $\mathfrak{A}$, i.e. if $\mathrm{atp}_{\mathfrak{A}}(b) \neq \mathrm{atp}_{\mathfrak{A}}(a)$ for all $b \neq a$. We let

- $K := \{a \in A : a \text{ is a king of } \mathfrak{A}\}$ be the set of kings of $\mathfrak{A}$, and
- $P := \{\mathrm{atp}_{\mathfrak{A}}(a) : a \in A, a \notin K\}$ be the set of atomic 1-types which are realized at least twice in $\mathfrak{A}$.

Since $\mathfrak{A} \models \forall x \exists y \beta_i$ for $i = 1, \ldots, n$, there exist (Skolem) functions $f_1, \ldots, f_n : A \to A$ such that $\mathfrak{A} \models \beta_i(a, f_i a)$ for all $a \in A$. The *court* of $\mathfrak{A}$ is defined as

$$C := K \cup \{f_i k : k \in K, i = 1, \ldots, n\}.$$

Let $\mathfrak{C}$ be the substructure of $\mathfrak{A}$ induced by C. We construct a model $\mathfrak{B} \models \psi$ with universe $B = C \cup (P \times \{1, \ldots, n\} \times \{0, 1, 2\})$.



To specify $\mathfrak{B}$ we set $\mathfrak{B}|_C = \mathfrak{C}$ and for all other elements we specify the 1- and 2-types (in this way fixing $\mathfrak{B}$ on the remaining part). However,

(1) This must be done consistently:
- $\mathrm{atp}_{\mathfrak{A}}(b, b')$ and $\mathrm{atp}_{\mathfrak{A}}(b, b'')$ must agree on $\mathrm{atp}_{\mathfrak{A}}(b)$, and
- $\gamma(x, y) \in \mathrm{atp}_{\mathfrak{B}}(b, b') \Leftrightarrow \gamma(y, x) \in \mathrm{atp}_{\mathfrak{B}}(b', b)$.

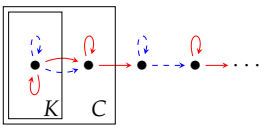
(2) Of course we have to ensure that $\mathfrak{B} \models \psi$.

We illustrate the construction with the following example.

Example 1.39. Consider the formula ψ over the signature $\tau = \{R, B\}$ (red edges and blue edges).

$$\begin{aligned}
\psi \;=\; & \exists x(Rxx \wedge Bxx) \\
\wedge\; & \forall x \forall y((Rxx \wedge Bxx \wedge Ryy \wedge Byy \to x = y) \\
& \wedge (Rxx \vee Bxx) \\
& \wedge (Rxy \wedge Ryx \to x = y) \\
& \wedge (Bxy \wedge Byx \to x = y) \\
& \wedge (Bxy \wedge x \neq y \to Ryy)) \\
\wedge\; & \forall x \exists y(x \neq y \wedge (Rxx \to Rxy) \\
\wedge\; & (Bxx \to Bxy)).
\end{aligned}$$

Let $\mathfrak{A} \models \psi$, then $\mathfrak{A}$ looks like follows:



In this case $P = \{\{Rxx, \neg Bxx\}, \{\neg Rxx, Bxx\}\}$ and the universe of $\mathfrak{B}$ is $B = C \cup (P \times \{1\} \times \{0,1,2\})$.

We proceed to construct $\mathfrak{B}$ by specifying the 1-types and 2-types of its elements as follows.

(1) The atomic 1-types of elements (p,i,j) are set to $\mathrm{atp}_{\mathfrak{B}}((p,i,j)) = p$.

(2) The atomic 2-types $\mathrm{atp}_{\mathfrak{B}}(b,b')$ will be set so that $\mathfrak{B} \models \forall x \exists y \beta_i$ for $i = 1, \ldots, m$.
Choose for each $p \in P$ an element $h(p) \in A$ with $\mathrm{atp}_{\mathfrak{A}}(h(p)) = p$.
Find for each $b \in \mathfrak{B}$ and each i a suitable element b' such that $\mathfrak{B} \models \beta_i(b,b')$ (by defining $\mathrm{atp}_{\mathfrak{B}}(b,b')$ appropriately).

(a) If b is a king, set $b' := f_i(b) \in C \subseteq B$. Then $\mathfrak{B} \models \beta_i(b,b')$.

(b) If $b \in C \setminus K$ (non-royal member of the court), distinguish:
- If $f_i(b) \in K$, then set $b' := f_i(b) \in K \subseteq B$.
- Otherwise it holds that $\mathrm{atp}_{\mathfrak{A}}(f_i(b)) = p \in P$.
  In this case, set $b' := (p,i,0)$. Now set $\mathrm{atp}_{\mathfrak{B}}(b,b') := \mathrm{atp}_{\mathfrak{A}}(b, f_i(b))$. Thus $\mathfrak{B} \models \beta_i(b,b')$ since $\mathfrak{A} \models \beta_i(b, f_i(b))$.

(c) If $b = (p,j,\ell)$ for some $p \in P, j \in \{1,\ldots,n\}, \ell \in \{0,1,2\}$, let $a := h(p)$ and consider $f_i(a)$.
If $f_i(a) \in K$, set $b' = f_i(a)$ and $\mathrm{atp}_{\mathfrak{B}}(b,b') := \mathrm{atp}_{\mathfrak{A}}(a,b')$.
If $f_i(a) \notin K$, then $\mathrm{atp}_{\mathfrak{A}}(f_i(a)) = p' \in P$.
Set $b' := (p', i, (\ell + 1) \pmod 3)$.
Then set $\mathrm{atp}_{\mathfrak{B}}(b,b') := \mathrm{atp}_{\mathfrak{A}}(a, f_i(a))$, and thus $\mathfrak{B} \models \beta_i(b,b')$.

To complete the construction of $\mathfrak{B}$, let $b_1, b_2 \in B$ be such that $\mathrm{atp}_{\mathfrak{B}}(b_1,b_2)$ is not yet specified. Choose $a_1, a_2 \in A$ so that

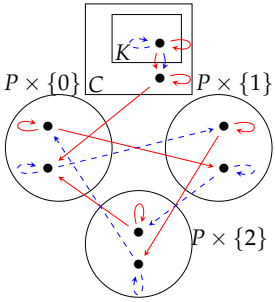
$$\begin{aligned} \mathrm{atp}_{\mathfrak{A}}(a_1) &= \mathrm{atp}_{\mathfrak{B}}(b_1) \text{ and} \\ \mathrm{atp}_{\mathfrak{A}}(a_2) &= \mathrm{atp}_{\mathfrak{B}}(b_2) \end{aligned}$$

and set

$$\mathrm{atp}_{\mathfrak{B}}(b_1,b_2) := \mathrm{atp}_{\mathfrak{A}}(a_1,a_2).$$

Since $\mathfrak{A} \models \alpha(a_1,a_2)$, also $\mathfrak{B} \models \alpha(b_1,b_2)$.

For the previously considered example, $\mathfrak{B}$ looks as follows:



Overall, we obtain $\mathfrak{B} \models \forall x \forall y \alpha \wedge \bigwedge_{i=1}^{n} \forall x \exists y \beta_i = \psi$, and the size of B is restricted by

$$|B| = \underbrace{|C|}_{\leq |K|(n+1)} + 3n|P| = \mathcal{O}(n \cdot \#\text{ (atomic 1-types)})\,.$$

For k relation symbols, there are 2^k atomic 1-types, hence $|B| = 2^{\mathcal{O}(|\psi|)}$.

Q.E.D.

This result implies that $Sat(\mathrm{FO}^2)$ is in NEXPTIME (indeed it is NEXPTIME-complete), since we can simply guess a finite structure $\mathfrak{A}$ of exponential size (in the length of ψ) and verify that $\mathfrak{A} \models \psi$.

Corollary 1.40. $Sat(\mathrm{FO}^2) \in \mathrm{NEXPTIME} = (\bigcup_k \mathrm{NTIME}(2^{n^k}))$.

This is a typical complexity level for decidable fragments of FO. In fact, $Sat(\mathrm{FO}^2)$ is even complete for NEXPTIME. For showing this, we reduce a bounded version of the domino problem to $Sat(\mathrm{FO}^2)$.

Definition 1.41. Let $\mathcal{D} = (D, H, V)$ be a domino system and let $Z(t)$ denote $\mathbb{Z}/t\mathbb{Z} \times \mathbb{Z}/t\mathbb{Z}$. For a word $w = w_0, \ldots, w_{n-1} \in D^n$ we say that $\mathcal{D}$ tiles $Z(t)$ with initial condition w if there is $\tau : Z(t) \to D$ such that

- if $\tau(x,y) = d$ and $\tau(x+1,y) = d'$ then $(d,d') \in H$ for all $(x,y) \in Z(t)$,
- if $\tau(x,y) = d, \tau(x,y+1) = d'$ then $(d,d') \in V$ for all $(x,y) \in Z(t)$ and
- $\tau(i,0) = w_i$ for all $i = 0, \ldots, n-1$.

Let $\mathcal{D}$ be a domino system and $T : \mathbb{N} \to \mathbb{N}$ a mapping. Define

$$\mathrm{DOMINO}(\mathcal{D}, T) := \{w \in D^* : \mathcal{D} \text{ tiles } Z(T(|w|)) \text{ with initial condition } w\}.$$

One can describe computations of a (in this case non-deterministic) Turing machine by domino tilings in such a way that the input condition of the domino problem relates to the initial configuration of the Turing machine. The restrictions on the size of the tiled rectangle correspond to the time and space restrictions of the Turing machine. To prove that a problem A is NEXPTIME-hard, it then suffices to show that $\mathrm{DOMINO}(\mathcal{D}, 2^n) \leq_p A$.

Our goal is to show that $\mathrm{DOMINO}(\mathcal{D}, 2^n)$ reduces to $Sat(X)$ for relatively simple classes $X \subseteq \mathrm{FO}$. Set

$$X = \{\varphi \in \mathrm{FO}^2 : \varphi = \forall x \forall y\, \alpha \wedge \forall x \exists y\, \beta, \text{ s.t. } \alpha, \beta \text{ quantifier-free, without } =, \text{ and with only monadic predicates}\}.$$

We show that $Sat(X)$ is NEXPTIME-complete and hence also $Sat(\mathrm{FO}^2)$ is NEXPTIME-complete.

Lemma 1.42. For each domino system $\mathcal{D} = (D, H, V)$ there exists a polynomial time reduction $w \in D^n \mapsto \psi_w \in X$ such that $\mathcal{D}$ tiles $Z(2^n)$ with initial condition w if and only if ψ_w is satisfiable.

Proof. The intended model of ψ_w is a description of a tiling $\tau : Z(2^n) \to D$ in the universe $Z(2^n)$.

Let $z = (a, b) \in Z(2^n)$ with $a = \sum_{i=0}^{n-1} a_i 2^i$ and $b = \sum_{i=0}^{n-1} b_i 2^i$. Encode the tuple as $(a_o, \ldots, a_{n-1}, b_0, \ldots, b_{n-1}) \in \{0,1\}^{2n}$.

To encode the tiling, we define ψ_w with the monadic predicates X_i, X_i^*, Y_i, Y_i^*, N_i for $0 \leq i < n$ and $P_d (d \in D)$ with the following intended meaning:

$$\begin{aligned} X_i z \quad &\text{iff} \quad a_i = 1. \\ X_i^* z \quad &\text{iff} \quad a_j = 1 \text{ for all } j < i. \\ Y_i z \quad &\text{iff} \quad b_j = 1. \end{aligned}$$

$$\begin{aligned} Y_i^* z \quad &\text{iff} \quad b_j = 1 \text{ for all } j < i. \\ N_i z \quad &\text{iff} \quad z = (i, 0). \\ P_d z \quad &\text{iff} \quad \tau(z) = d. \end{aligned}$$

ψ_w will have the form $\psi_w = \forall x \forall y \alpha \wedge \forall x \exists y \beta$, where β accounts for the correct interpretation of $X_i, X_i^*, Y_i, Y_i^*, N_i$ and ensures that every element has a successor, and α accounts for the description of a correct tiling.

Now β is the the following formula:

$$\begin{aligned} \beta = {} & X_0^* x \wedge Y_0^* x \\ \wedge \quad & \bigwedge_{i=1}^{n-1} X_i^* x \leftrightarrow (X_{i-1}^* x \wedge X_{i-1} x) \\ \wedge \quad & \bigwedge_{i=1}^{n-1} Y_i^* x \leftrightarrow (Y_{i-1}^* x \wedge Y_{i-1} x) \\ \wedge \quad & \bigwedge_{i=0}^{n-1} X_i y \leftrightarrow (X_i x \oplus X_i^* x) \\ \wedge \quad & \bigwedge_{i=0}^{n-1} Y_i y \leftrightarrow (Y_i x \oplus (Y_i^* x \wedge X_{n-1} x \wedge X_{n-1}^* x)) \\ \wedge \quad & N_0 x \leftrightarrow (\bigwedge_{i=0}^{n-1} \neg X_i x \wedge \neg Y_i x) \\ \wedge \quad & \bigwedge_{i=0}^{n-1} N_i x \leftrightarrow N_{i+1} y. \end{aligned}$$

We define the following shorthands for use in α:

$$\begin{aligned} H(x,y) \quad &:= \quad \bigwedge_{i=0}^{n-1} (Y_i y \leftrightarrow Y_i x) \wedge \bigwedge_{i=0}^{n-1} (X_i y \leftrightarrow (X_i x \oplus X_i^* x)) \\ V(x,y) \quad &:= \quad \bigwedge_{i=0}^{n-1} (X_i y \leftrightarrow X_i x) \wedge \bigwedge_{i=0}^{n-1} (Y_i y \leftrightarrow (Y_i x \oplus Y_i^* x)). \end{aligned}$$

Now α is defined to be

$$\begin{aligned}\alpha = & \bigwedge_{d \neq d'} \neg(P_d x \wedge P_{d'} x) \\ \wedge \quad & (H(x,y) \to \bigvee_{(d,d') \in H} (P_d x \wedge P_{d'} y)) \\ \wedge \quad & (V(x,y) \to \bigvee_{(d,d') \in V} (P_d x \wedge P_{d'} y)) \\ \wedge \quad & (\bigwedge_{i=i}^{n-1} (N_i x \to P_{w_i} x)).\end{aligned}$$

Claim 1.43. ψ_w is satisfiable if and only if $\mathcal{D}$ tiles $Z(2^n)$ with initial condition w.

Proof. We show both directions.

($\Leftarrow$) Consider the intended model, ψ_w holds in it.
($\Rightarrow$) Consider $\mathfrak{C} = (C, X_1, \ldots) \models \psi_w$ and define a mapping

$$\begin{aligned} f: \; C &\to Z(2^n) \\ c &\mapsto (a,b) \equiv (a_0, \ldots, a_{n-1}, b_0, \ldots, b_{n-1}) \end{aligned}$$

$$\begin{aligned} \text{with } a_i = 1 \quad &\text{iff} \quad \mathfrak{C} \models X_i c \quad \text{and} \\ b_i = 1 \quad &\text{iff} \quad \mathfrak{C} \models Y_i c. \end{aligned}$$

As $\mathfrak{C} \models \forall x \exists y \beta$, f is surjective. Choose for each $z \in Z(2^n)$ an element $c \in f^{-1}(z)$ and set $\tau(z) = d$ for the unique d that satisfies $\mathfrak{C} \models P_d c$. Then τ is a correct tiling with initial condition w. Q.E.D.

Since the length of ψ_w is $|\psi_w| = O(n \log n)$, the above claim completes the proof of the lemma. Q.E.D.

2 Descriptive Complexity

In this chapter we study the relationship between logical definability and computational complexity on finite structures. In contrast to the theory of computational complexity we do not measure resources as time and space required to decide a property but the logical resources needed to define it. The ultimate goal is to characterize the complexity classes known from computational complexity theory by means of logic.

We first define what it means for a logic to capture a complexity class. One of the main results is due to Fagin, stating that existential second order logic captures NP. At this point it is still unknown whether there exists a logic capturing PTIME on all finite structures. However, a deeper analysis of the proof of Fagin's Theorem shows that SO-HORN logic captures PTIME on all *ordered* finite structures.

2.1 Logics Capturing Complexity Classes

To measure the complexity of a property of finite τ-structures, (for instance, graph) we have to represent the structures by words over a finite alphabet Σ, so that they can serve as inputs for Turing machines. For graphs, a natural choice is to take an adjacency matrix, and write it, row after row, as binary string. Notice that one and the same graph can have many different adjacency matrices, and thus many different encodings. Moreover, it is an important open problem to decide efficiently (i.e. in polynomial time) whether two different matrices represent the same graph, up to isomorphism. The choice of an adjacency matrix means to fix an enumeration of the vertices, and thus an *ordering* of the graph. The same is true for encoding finite structures of any fixed finite vocabulary τ: to define an encoding it is necessary to fix an ordering on the universe.

By $\mathrm{Ord}(\tau)$ we denote the class of all finite structures $(\mathfrak{A}, <)$, where $\mathfrak{A}$ is a τ-structure and $<$ is a linear order on its universe. For any

structure $\mathfrak{A} \in \mathrm{Ord}(\tau)$ with universe of size n, and for any fixed k, we can identify A^k with the set $\{0, 1, \ldots, n^k - 1\}$. This is done by associating each k-tuple $\bar{a}$ with its rank in the lexicographic ordering induced by $<$ on A^k. When we talk about the $\bar{a}$-th element, we understand it in this sense.

Definition 2.1. An *encoding* is a function mapping ordered structures to words. An encoding $\mathrm{code}(\cdot) : \mathrm{Ord}(\tau) \to \Sigma^*$ is good if it identifies isomorphic structures, is polynomially bounded, first-order definable and allows to compute the values of atomic statements efficiently. Formally, the following abstract conditions must be satisfied.

- $\mathrm{code}(\mathfrak{A}, <) = \mathrm{code}(\mathfrak{B}, <)$ iff $(\mathfrak{A}, <) \cong (\mathfrak{B}, <)$.
- There is a fixed polynomial p such that $|\mathrm{code}(\mathfrak{A}, <)| \leq p(|A|)$ for all $(\mathfrak{A}, <) \in \mathrm{Ord}(\tau)$.
- For all $k \in \mathbb{N}$ and all $\sigma \in \Sigma$ there exists a first-order formula $\beta_\sigma(x_1, \ldots, x_k)$ of vocabulary $\tau \cup \{<\}$ so that for all $(\mathfrak{A}, <)$ and all $\bar{a} \in A^k$ it holds that

$$(\mathfrak{A}, <) \models \beta_\sigma(\bar{a}) \Leftrightarrow \text{ the } \bar{a}\text{-th symbol of } \mathrm{code}(\mathfrak{A}, <) \text{ is } \sigma.$$

- Given $\mathrm{code}(\mathfrak{A}, <)$ a relation symbol R of τ and a tuple $\bar{a}$ one can efficiently decide whether $\mathfrak{A} \models R\bar{a}$.

The meaning of "efficiently" in the last condition may depend on the context, here we understand it is as evaluated in linear time and logarithmic space.

Example 2.2. Let $\mathfrak{A} = (A, R_1, \ldots, R_m)$ be a structure with a linear order $<$ on A. Let $|A| = n$ and let s_i be the arity of R_i. Let ℓ be the maximal arity of $R_1, \ldots, R_m$. For each relation we define

$$\chi(R_j) = w_0 \ldots w_{n^{s_j}-1} 0^{n^\ell - n^{s_j}} \in \{0,1\}^{n^\ell},$$

where $w_i = 1$ if the i-th element of A^{s_j} is in R_j. Now

$$\mathrm{code}(\mathfrak{A}, <) := 1^n 0^{n^\ell - n} \chi(R_1) \ldots \chi(R_m).$$

When we say that an algorithm decides a class $\mathcal{K}$ of finite τ-structures

we actually mean that it decides

$$\text{code}(\mathcal{K}) = \{\text{code}(\mathfrak{A}, <) : \mathfrak{A} \in \mathcal{K}, < \text{ a linear order on } A\}.$$

Definition 2.3. A *model class* is a class $\mathcal{K}$ of structures of a fixed vocabulary τ that is closed under isomorphism, i.e. if $\mathfrak{A} \in \mathcal{K}$ and $\mathfrak{A} \cong \mathfrak{B}$, then $\mathfrak{B} \in \mathcal{K}$.

A *domain* is an isomorphism closed class $\mathcal{D}$ of structures where the vocabulary is not fixed. For a domain $\mathcal{D}$ and vocabulary τ, we write $\mathcal{D}(\tau)$ for the class of τ-structures in $\mathcal{D}$.

Definition 2.4. Let L be a logic, Comp a complexity class and $\mathcal{D}$ a domain of finite structures. L captures Comp on $\mathcal{D}$ if

(1) For every vocabulary τ and every (fixed) sentence $\psi \in L(\tau)$, the model-checking problem for ψ on $\mathcal{D}(\tau)$ is in Comp.
(2) For every vocabulary τ and any model class $\mathcal{K} \subseteq \mathcal{D}(\tau)$ whose membership problem is in Comp, there exists a sentence $\psi \in L(\tau)$ such that

$$\mathcal{K} = \{\mathfrak{A} \in \mathcal{D}(\tau) : \mathfrak{A} \models \psi\}.$$

Notice that first-order logic is very weak, in this sense. Indeed, for every fixed first-order sentence $\psi \in \text{FO}(\tau)$, it can be decided efficiently, with logarithmic space, whether a given finite τ-structure is a model for ψ. However, FO does *not* capture Logspace, not even on ordered structures. Indeed, the reachability problem on undirected graphs can be solved in Logspace, but it is not first-order expressible.

2.2 Fagin's Theorem

Existential second-order logic (Σ^1_1) is the fragment of second-order logic consisting of formulae of the form $\exists R_1 \ldots \exists R_m \varphi$ where $\varphi \in$ FO and $R_1, \ldots, R_m$ are relation symbols. As we will see in this chapter, the logic Σ^1_1 captures the complexity class NP on the domain of all finite structures.

Example 2.5. 3-Colourability of a graph $G = (V, E)$ is in NP and indeed there is a Σ^1_1-formula defining the class of graphs which possess a valid

3-colouring:

$$\begin{aligned} \exists R \exists B \exists Y \quad (\quad & \forall x (Rx \vee Bx \vee Yx) \\ \wedge \quad & \forall x \forall y (Exy \rightarrow \neg((Rx \wedge Ry) \vee (Bx \wedge By) \vee (Yx \wedge Yy)) \end{aligned}$$

Theorem 2.6 (Fagin)**.** Existential second-order logic captures NP on the domain of all finite structures.

Proof. The proof consists of two parts. First, let $\psi = \exists R_1 \ldots \exists R_m \varphi \in \Sigma^1_1$ be an existential second-order sentence. We show that it can be decided in non-deterministic polynomial time whether a given structure $\mathfrak{A}$ is a model of ψ.

In a first step, we guess relations $R_1, \ldots, R_m$ on A. Recall that relations can be identified with binary strings of length n^{s_i}, where s_i is the arity of R_i. Then we check whether $(\mathfrak{A}, R_1, \ldots, R_m) \models \varphi$ which can be done in LOGSPACE and hence in PTIME. Thus the computation consists of guessing a polynomial number of bits followed by a deterministic polynomial time computation, showing that the problem is in NP.

For the other direction, let $\mathcal{K}$ be an isomorphism-closed class of τ-structures and let M be a non-deterministic TM deciding code$(\mathcal{K})$ in polynomial time. We construct a sentence $\psi \in \Sigma^1_1$ such that for all finite τ-structure $\mathfrak{A}$ it holds that

$$\mathfrak{A} \models \psi \Leftrightarrow M \text{ accepts code}(\mathfrak{A}, <) \text{ for any linear order } < \text{ on } A.$$

Let $M = (Q, \Sigma, q_0, F^+, F^-, \delta)$ with accepting and rejecting states F^+ and F^- and $\delta : (Q \times \Sigma) \rightarrow \mathcal{P}(Q \times \Sigma \times \{0, 1, -1\})$ which, given an input code$(\mathfrak{A}, <)$, decides in non-deterministic polynomial time whether $\mathfrak{A}$ belongs to $\mathcal{K}$ or not. We assume that all computations of M reach an accepting or rejecting state after precisely n^k steps ($n := |A|$).

We encode a computation of M on code$(\mathfrak{A}, <)$ by relations $\overline{X}$ and construct a first-order sentence $\varphi_M \in \text{FO}(\tau \cup \{<\} \cup \{\overline{X}\})$ such that for every linear order $<$ there exists $\overline{X}$ with $(\mathfrak{A}, <, \overline{X}) \models \varphi_M$ if and only if code$(\mathfrak{A}, <) \in L(M)$. To this end we show that

- If $\overline{X}$ represents an accepting computation of M on code$(\mathfrak{A}, <)$ then $(\mathfrak{A}, <, \overline{X}) \models \varphi_M$.

- If $(\mathfrak{A}, <, \overline{X}) \models \varphi_M$ then $\overline{X}$ *contains* a representation of an accepting computation of M on code$(\mathfrak{A}, <)$.

Accordingly the desired formula ψ is then obtained via existential second-order quantification

$$\psi := (\exists <)(\exists \overline{X})(\text{“}< \text{ is a linear order ”} \wedge \varphi_M).$$

Details:

- We represent numbers up to n^k as tuples in A^k.
- For each state $q \in Q$ we introduce a predicate

$$X_q := \{\bar{t} \in A^k : \text{ at time } \bar{t} \text{ the TM } M \text{ is in state } q\}.$$

- For each symbol $\sigma \in \Sigma$ we define

$$Y_\sigma := \{(\bar{t}, \bar{a}) \in A^k \times A^k : \text{ at time } \bar{t} \text{ the cell } \bar{a} \text{ contains } \sigma\}.$$

- The head predicate is

$$Z := \{(\bar{t}, \bar{a}) \in A^k \times A^k : \text{ at time } \bar{t} \text{ the head of } M \text{ is at position } \bar{a}\}.$$

Now φ_M is the universal closure of START $\wedge$ COMPUTE $\wedge$ END.

$$\text{START} := X_{q_0}(\overline{0}) \wedge Z(\overline{0}, \overline{0}) \wedge \bigwedge_{\sigma \in \Sigma} (\beta_\sigma(\overline{x}) \to Y_\sigma(\overline{0}, \overline{x})).$$

Recall that β_σ states that the symbol at position $\overline{x}$ in code$(\mathfrak{A}, <)$ is σ. The existence of the formulae β_σ is guaranteed by the fact that code$(\cdot)$ is a good encoding. In what follows, we denote by $\overline{x} + 1$ and $\overline{x} - 1$ a first-order formula that defines the direct successor and predecessor of the tuple $\overline{x}$ (in the lexicographical ordering on tuples that is induced by the linear order $<$), respectively.

$$\text{COMPUTE} := \text{NOCHANGE} \wedge \text{CHANGE}.$$

$$\text{NOCHANGE} := \bigwedge_{\sigma\in\Sigma} (Y_\sigma(\bar{t},\bar{x}) \wedge Z(\bar{t},\bar{y}) \wedge \bar{y} \neq \bar{x} \wedge \bar{t}' = \bar{t}+1 \to Y_\sigma(\bar{t}',\bar{x})).$$

$$\text{CHANGE} := \bigwedge_{q\in Q,\sigma\in\Sigma} (\text{PRE}[q,\sigma] \to \bigvee_{(q',\sigma',m)\in\delta(q,\sigma)} \text{POST}[q',\sigma',m]),$$

where

$$\text{PRE}[q,\sigma] := X_q(\bar{t}) \wedge Z(\bar{t},\bar{x}) \wedge Y_\sigma(\bar{t},\bar{x}) \wedge \bar{t}' = \bar{t}+1,$$

$$\text{POST}[q',\sigma',m] := X_{q'}(\bar{t}') \wedge Y_{\sigma'}(\bar{t}',\bar{x}) \wedge \text{MOVE}_m[\bar{t}',\bar{x}],$$

and

$$\text{MOVE}_m[\bar{t}',\bar{x}] := \begin{cases} \exists\bar{y}(\bar{x}-1=\bar{y} \wedge Z(\bar{t}',\bar{y})), & m=-1 \\ Z(\bar{t}',\bar{x}), & m=0 \\ \exists\bar{y}(\bar{x}+1=\bar{y} \wedge Z(\bar{t}',\bar{y})), & m=1. \end{cases}$$

Finally, we let

$$\text{END} := \bigwedge_{q\in F^-} \neg X_q(\bar{t}).$$

It remains to show the following two claims.
Claim 1. If $\overline{X}$ represents an accepting computation of M on code$(\mathfrak{A},<)$ then $(\mathfrak{A},<,\overline{X}) \models \varphi_M$. This, however, follows immediately from the construction of φ_M.
Claim 2. If $(\mathfrak{A},<,\overline{X}) \models \varphi_M$, then $\overline{X}$ contains a representation of an accepting computation of M on code$(\mathfrak{A},<)$. We define

$$\text{CONF}[C,j] := X_q(\bar{j}) \wedge Z(\bar{j},\bar{p}) \wedge \bigwedge_{i=0}^{n^k-1} Y_{w_i}(\bar{j},\bar{i})$$

for configurations $C = (w_0 \dots w_{n^k-1}, q, p)$ (tape content $w_0 \dots w_{n^k-1}$, state q, head position p), i.e. the conjunction of the atomic statements

that hold for C at time j. Let C_0 be the input configuration of M on code$(\mathfrak{A}, <)$. Since $(\mathfrak{A}, <, \overline{X}) \models$ START it follows that

$$(\mathfrak{A}, <, \overline{X}) \models \text{CONF}[C_0, 0].$$

Since $(\mathfrak{A}, <, \overline{X}) \models$ COMPUTE and $(\mathfrak{A}, <, \overline{X}) \models \text{CONF}[C_i, t]$, for some $C_i \vdash C_{i+1}$ it holds that $(\mathfrak{A}, <, \overline{X}) \models \text{CONF}[C_{i+1}, t+1]$.

Finally, no rejecting configuration can be encoded in $\overline{X}$ because $(\mathfrak{A}, <, \overline{X}) \models$ END. Thus an accepting computation

$$C_0 \vdash C_1 \vdash \ldots \vdash C_{n^k-1}$$

of M on code$(\mathfrak{A}, <)$ exists, with $(\mathfrak{A}, <, \overline{X}) \models \text{CONF}[C_i, i]$ for all $i \leq n^k - 1$. This completes the proof of Fagin's Theorem. Q.E.D.

Theorem 2.7 (Cook, Levin)**.** SAT is NP-complete.

Proof. Obviously SAT $\in$ NP. We show that for any Σ^1_1-definable class $\mathcal{K}$ of finite structures the membership problem $\mathfrak{A} \in \mathcal{K}$ can be reduced to SAT. By Fagin's Theorem, there exists a first-order sentence ψ such that

$$\mathcal{K} = \{\mathfrak{A} \in \text{Fin}(\tau) : \mathfrak{A} \models \exists R_1 \ldots \exists R_m \psi\}.$$

Given $\mathfrak{A}$, construct a propositional formula $\psi_{\mathfrak{A}}$ as follows.

- replace $\exists x_i \varphi$ by $\bigvee_{a \in A} \varphi[x_i/a]$,
- replace $\forall x_i \varphi$ by $\bigwedge_{a \in A} \varphi[x_i/a]$,
- replace all closed τ-atoms $P\overline{a}$ in ψ with their truth values,
- replace all atoms $R\overline{a}$ with propositional variables $P_{R\overline{a}}$.

This is a polynomial transformation and it holds that

$$\mathfrak{A} \in \mathcal{K} \Leftrightarrow \mathfrak{A} \models \exists R_1 \ldots \exists R_m \psi \Leftrightarrow \psi_{\mathfrak{A}} \in \text{SAT}.$$

Q.E.D.

2.3 Second Order Horn Logic on Ordered Structures

The problem of whether there exists a logic capturing PTIME on all finite structures is still open. However, on *ordered* finite structures, there are several known logical characterizations of PTIME. The most famous result of this kind is the one is the Theorem by Immerman and Vardi which states that the least fixed-point logic LFP captures PTIME on the class of all ordered finite structures. We shall discuss this later. We here present a different characterization of PTIME, in terms of second-order Horn logic SO-HORN, which follows from a careful analysis of the proof of Fagin's Theorem. Indeed, the construction that we used in that proof is not the original one by Fagin, but an optimized version that has been tailored so that it can be adapted to a proof that SO-HORN captures PTIME on ordered structures.

Definition 2.8. *Second-order Horn logic*, denoted by SO-HORN, is the set of second-order sentences of the form

$$Q_1 R_1 \dots Q_m R_m \forall y_1 \dots \forall y_s \bigwedge_{i=1}^{t} C_i,$$

where $Q_i \in \{\exists, \forall\}$ and the C_i are Horn clauses, i.e. implications

$$\beta_1 \wedge \dots \wedge \beta_m \rightarrow H,$$

where each β_j is either a positive atom $R_k \bar{z}$ or an FO-formula that does not contain $R_1, \dots, R_m$. H is either a positive atom $R_j \bar{z}$ or the Boolean constant 0.

Σ_1^1-HORN denotes the existential fragment of SO-HORN, i.e. the set of SO-HORN sentences where all second-order quantifiers are existential.

Theorem 2.9. Every sentence $\psi \in$ SO-HORN is equivalent to a sentence $\psi' \in \Sigma_1^1$-HORN.

Proof. It suffices to prove the theorem for formulae of the form

$$\psi = \forall P \exists R_1 \dots \exists R_m \forall \bar{z} \varphi,$$

where φ is a conjunction of Horn clauses and $m \geq 0$ (for $m = 0$, the formula has the form $\forall P \forall \overline{z} \varphi$). Indeed we can then eliminate universal quantifiers beginning with the inner most one by considering only the part starting with that universal quantifier.

Lemma 2.10. A formula $\exists \overline{R} \forall \overline{z} \varphi(P, \overline{R}) \in \Sigma^1_1$-HORN holds for all relations P on a structure $\mathfrak{A}$ if and only if it holds for those P that are false at at most one point.

Proof. Let k be the arity of P. For every k-tuple $\overline{a}$, let $P^{\overline{a}} = A^k - \{\overline{a}\}$, i.e. the relation that is false at $\overline{a}$ and true at all other points. By assumption, there exist $\overline{R}^{\overline{a}}$ such that

$$(\mathfrak{A}, P^{\overline{a}}, \overline{R}^{\overline{a}}) \models \forall \overline{z} \varphi.$$

Now consider any $P \neq A^k$ and let $R_i := \bigcap_{\overline{a} \notin P} R_i^{\overline{a}}$. We show that $(\mathfrak{A}, P, \overline{R}) \models \forall \overline{z} \varphi$ where $\overline{R}$ is the tuple consisting of all R_i.

Suppose that this is false, then there exists a relation $P \neq A^k$, a clause C of φ and an assignment $\rho : \{z_1, \ldots, z_s\} \to A$ such that $(\mathfrak{A}, P, \overline{R}) \models \neg C[\rho]$. We proceed to show that in this case there exists a tuple $\overline{a}$ such that $(\mathfrak{A}, P^{\overline{a}}, \overline{R}^{\overline{a}}) \models \neg C[\rho]$ and thus

$$(\mathfrak{A}, P^{\overline{a}}, \overline{R}^{\overline{a}}) \models \neg \forall \overline{z} \varphi$$

which contradicts the assumption.

- If the head of $C[\rho]$ is $P\overline{a}$, then take $\overline{a} = \overline{u} \notin P$.
- If the head of $C[\rho]$ is $R_i \overline{u}$, then choose $\overline{a} \notin P$ such that $\overline{u} \notin R_i^{\overline{a}}$, which exists because $\overline{u} \notin R_i$.
- If the head is 0, take an arbitrary $\overline{a} \notin P$.

The head of $C[\rho]$ is clearly false in $(\mathfrak{A}, P^{\overline{a}}, \overline{R}^{\overline{a}})$. $P\overline{a}$ does not occur in the body of $C[\rho]$, because $\overline{a} \notin P$ and all atoms in the body of $C[\rho]$ are true in $(\mathfrak{A}, P, \overline{R})$. All other atoms of the form P_i that might occur in the body of the clause remain true for $P^{\overline{a}}$. Moreover, every atom $R_i \overline{v}$ in the body remains true if R_i is replaced by $R_i^{\overline{a}}$ because $R_i \subseteq R_i^{\overline{a}}$. This implies $(\mathfrak{A}, P^{\overline{a}}, \overline{R}^{\overline{a}}) \models \neg C[\rho]$. Q.E.D.

Using the above lemma, the original formula $\psi = \forall P \exists R_1 \ldots \exists R_m \forall \overline{z} \varphi$ is equivalent to

$$\exists \overline{R}\ \forall \overline{z} \varphi[P\overline{u}/\overline{u} = \overline{u}] \wedge \forall \overline{y}\ \exists \overline{R}\ \forall \overline{z} \varphi[P\overline{u}/\overline{u} \neq \overline{y}].$$

This formula can be converted again to Σ_1^1-HORN; in the second part we push the external first-order quantifiers inside while increasing the arity of quantified relations by $|\overline{y}|$ to compensate it, i.e. we get

$$\exists \overline{R'}\ \forall \overline{y}\, \overline{z} \varphi[P\overline{u}/\overline{u} \neq \overline{y}, R(\overline{x})/R'(\overline{x}, \overline{y})].$$

Q.E.D.

Theorem 2.11. If $\psi \in$ SO-HORN, then the set of finite models of ψ, $\mathrm{Mod}(\psi)$, is in PTIME.

Proof. Given $\psi' \in$ SO-HORN, transform it to an equivalent sentence $\psi = \exists R_1 \ldots \exists R_m \forall \overline{z} \bigwedge_i C_i$ in Σ_1^1-HORN. Given a finite structure $\mathfrak{A}$ reduce the problem of whether $\mathfrak{A} \models \psi$ to HORNSAT (as in the proof of the Theorem of Cook and Levin).

- Omit quantifiers $\exists R_i$.
- Replace the universal quantifiers $\forall z_i \eta(z_i)$ by $\bigwedge_{a \in A} \eta[z_i/a]$.
- If there is a clause that is already made false by this interpretation, i.e. $C = 1 \wedge \ldots \wedge 1 \rightarrow 0$, reject ψ. Else interpret atoms $R_i \overline{u}$ as propositional variables.

The resulting formula is a propositional Horn formula with length polynomially bounded in $|A|$ and which is satisfiable iff $\mathfrak{A} \models \psi$. The satisfiability problem HORNSAT can be solved in linear time. Q.E.D.

Theorem 2.12 (Grädel)**.** On ordered finite structures SO-HORN and Σ_1^1-HORN capture PTIME.

Proof. We analyze the formula φ_M constructed in the proof of Fagin's Theorem in the case of a deterministic TM M. Recall that φ_M is the universal closure of START $\wedge$ NOCHANGE $\wedge$ CHANGE $\wedge$ END. START, NOCHANGE and END are already in Horn form. CHANGE has the

form

$$\bigwedge_{q\in Q,\sigma\in\Sigma} (\mathrm{PRE}[q,\sigma] \to \bigvee_{(q',\sigma',m)\in\delta(q,\sigma)} \mathrm{POST}[q',\sigma',m]).$$

For a deterministic M for each (q,σ) there is a unique $\delta(q,\sigma) = (q',\sigma',m)$. In this case $\mathrm{PRE}[q,\sigma] \to \mathrm{POST}[q',\sigma',m]$ can be replaced by the conjunction of the Horn clauses

- $\mathrm{PRE}[q,\sigma] \to X_{q'}(\overline{t}')$
- $\mathrm{PRE}[q,\sigma] \to Y_{\sigma'}(\overline{t}',\overline{x})$
- $\mathrm{PRE}[q,\sigma] \wedge \overline{y} = \overline{x} + m \to Z(\overline{t}',\overline{y})$.

Q.E.D.

Remark 2.13. The assumption that a linear order is explicitly available cannot be eliminated, since linear orderings are not definable by Horn formulae.

3 Expressive Power of First-Order Logic

In the whole chapter we restrict ourselves to *finite* and *relational* vocabularies τ.

3.1 Ehrenfeucht-Fraïssé Theorem

Let $\mathfrak{A}$ and $\mathfrak{B}$ be τ-structures with $\overline{a} \in A^k$ and $\overline{b} \in B^k$ for some $k \geq 0$. Recall that we write $\mathfrak{A}, \overline{a} \equiv \mathfrak{B}, \overline{b}$ if no FO-formula can distinguish between $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{B}, \overline{b})$, that is if for all $\varphi(\overline{x}) \in \mathrm{FO}(\tau)$ we have

$$\mathfrak{A} \models \varphi(\overline{a}) \Leftrightarrow \mathfrak{B} \models \varphi(\overline{b}).$$

For $m \geq 0$ we write $\mathfrak{A}, \overline{a} \equiv_m \mathfrak{B}, \overline{b}$ if the same holds for all $\mathrm{FO}(\tau)$-formulas of quantifier rank at most m. We aim to develop an algebraic characterisation of $\equiv_m$ via *back-and-forth systems* and a game-theoretic characterisation via *Ehrenfeucht-Fraïssé games*.

Back-and-forth systems. A *partial isomorphism* between τ-structures $\mathfrak{A}$ and $\mathfrak{B}$ is a bijective function p with *finite* domain $\mathrm{dom}(p) \subseteq A$ and range $\mathrm{rg}(p) \subseteq B$ such that p is an isomorphism between the substructures of $\mathfrak{A}$ and $\mathfrak{B}$ induced on $\mathrm{dom}(p)$ and $\mathrm{rg}(p)$, respectively, that is

$$p : \mathfrak{A} \restriction \mathrm{dom}(p) \cong \mathfrak{B} \restriction \mathrm{rg}(p).$$

$\mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ denotes the set of partial isomorphism between $\mathfrak{A}$ and $\mathfrak{B}$. For all $\mathfrak{A}$ and $\mathfrak{B}$ we have $\emptyset \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$. For $p \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ we write $p = \overline{a} \to \overline{b}$ for $\overline{a} \in A^k$ and $\overline{b} \in B^k$ if $\mathrm{dom}(p) = \{a_1, \ldots, a_k\}$ and $\mathrm{rg}(p) = \{b_1, \ldots, b_k\}$ and if $p(a_i) = b_i$ for $1 \leq i \leq k$.

Definition 3.1. Let $I \subseteq \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ and $p \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$. Then p has *back-and-forth extensions* in I if

$$\forall a \in A \, \exists b \in B : p \cup \{(a, b)\} \in I \qquad \text{(forth)}$$
$$\forall b \in B \, \exists a \in A : p \cup \{(a, b)\} \in I \qquad \text{(back)}$$

Accordingly, for $I, J \subseteq \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ we say that I has *back-and-forth extensions* in J, if every $p \in I$ has back-and-forth extensions in J.

Definition 3.2. Let $m \geq 0$. A *back-and-forth system* for m-equivalence of $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{B}, \overline{b})$ is a sequence $(I_i)_{i \leq m}$ of sets of partial isomorphisms $I_i \subseteq \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ such that

- $\overline{a} \to \overline{b} \in I_m$, and
- for all $0 < i \leq m$, I_i has back-and-forth extensions in I_{i-1}.

If such a system $(I_i)_{i \leq m}$ for $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{B}, \overline{b})$ exists, then we write

$$(I_i)_{i \leq m} : (\mathfrak{A}, \overline{a}) \simeq_m (\mathfrak{B}, \overline{b}),$$

and we say that $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{B}, \overline{b})$ are *m-isomorphic*.

Lemma 3.3. For every $m \geq 0$, every τ-structure $\mathfrak{A}$ and every $\overline{a} \in A^k$, there exists an $\mathrm{FO}(\tau)$-formula $\chi^m_{\mathfrak{A},\overline{a}}(x_1, \ldots, x_k)$ of quantifier rank m such that for all $\mathfrak{B}$ and $\overline{b} \in B^k$ we have

$$\mathfrak{B} \models \chi^m_{\mathfrak{A},\overline{a}}(\overline{b}) \Leftrightarrow \mathfrak{A}, \overline{a} \simeq_m \mathfrak{B}, \overline{b}.$$

Moreover the number of different formulas $\chi^m_{\mathfrak{A},\overline{a}}$ only depends on m, τ, and k, and not on $\mathfrak{A}$ or $\overline{a}$ (up to logical equivalence).

Proof. The construction is by induction on $m \geq 0$ (for all $k \geq 0$, $\mathfrak{A}$, and $\overline{a} \in A^k$ at the same time).

$$\chi^0_{\mathfrak{A},\overline{a}}(x_1, \ldots, x_k) = \bigwedge \{\varphi(x_1, \ldots, x_k) : \varphi \text{ is an atomic or negated atomic } \mathrm{FO}(\tau)\text{-formula with } \mathfrak{A} \models \varphi(x_1, \ldots, x_k)\}$$

We have that $\mathfrak{A}, \overline{a} \simeq_0 \mathfrak{B}, \overline{b}$ if, and only if, $\overline{a} \to \overline{b} \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$ which means that $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{B}, \overline{b})$ satisfy the same atomic formulas. Note that

the number of different atomic formulas in k variables only depends on the vocabulary τ and on $k \geq 0$.

Now let $m > 0$. Then we set $\chi^m_{\mathfrak{A},\overline{a}}(x_1, \ldots, x_k) =$

$$\bigwedge_{a' \in A} \exists x \, \chi^{m-1}_{\mathfrak{A},\overline{a},a'}(x_1, \ldots, x_k, x) \wedge \forall x \bigvee_{a' \in A} \chi^{m-1}_{\mathfrak{A},\overline{a},a'}(x_1, \ldots, x_k, x).$$

Since the number of different formulas $\chi^{m-1}_{\mathfrak{A},\overline{a},a'}$ (up to equivalence) only depends on $m-1$ and $k+1$ (by the induction hypothesis), also the number of different formulas $\chi^m_{\mathfrak{A},\overline{a}}$ only depends on m and k (up to equivalence) and not on $\mathfrak{A}$ or $\overline{a}$. This is of particular importance if one of the structures is infinite, because it guarantees that the conjunction and the disjunction in $\chi^m_{\mathfrak{A},\overline{a}}$ are finite. It holds

$$\begin{aligned}
& (\mathfrak{A}, \overline{a}) \simeq_m (\mathfrak{B}, \overline{b}) \\
\iff \quad & \begin{cases} \forall a' \in A \, \exists b' \in B : (\mathfrak{A}, \overline{a}, a') \simeq_{m-1} (\mathfrak{B}, \overline{b}, b') \\ \forall b' \in B \, \exists a' \in A : (\mathfrak{A}, \overline{a}, a') \simeq_{m-1} (\mathfrak{B}, \overline{b}, b') \end{cases} \\
\iff \text{(by (IH))} \quad & \begin{cases} \forall a' \in A \, \exists b' \in B : \mathfrak{B} \models \chi^{m-1}_{\mathfrak{A},\overline{a},a'}(\overline{b}, b') \\ \forall b' \in B \, \exists a' \in A : \mathfrak{B} \models \chi^{m-1}_{\mathfrak{A},\overline{a},a'}(\overline{b}, b') \end{cases} \\
\iff \quad & \mathfrak{B} \models \chi^m_{\mathfrak{A},\overline{a}}(\overline{b}).
\end{aligned}$$

Q.E.D.

Ehrenfeucht-Fraïssé games. The Ehrenfeucht-Fraïssé game $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$ is played by two players according to the following rules.

The *arena* consists of the structures $\mathfrak{A}$ and $\mathfrak{B}$. We assume that $A \cap B = \emptyset$. The players are called *Spoiler* and *Duplicator*, and a play of $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$ consists of m moves.

The initial position is $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$. In the *i-th move*, $1 \leq i \leq m$, the play proceeds from the position

$$G_{m-i+1}(\mathfrak{A}, \overline{a}, c_1, \ldots, c_{i-1}, \mathfrak{B}, \overline{b}, d_1, \ldots, d_{i-1}).$$

Spoiler either chooses an element $c_i \in A$ or an element $d_i \in B$. Duplicator answers by choosing an element $c_i \in A$ or $d_i \in B$ in the other structure. The new position is $G_{m-i}(\mathfrak{A}, \overline{a}, c_1, \ldots, c_i, \mathfrak{B}, \overline{b}, d_1, \ldots, d_i)$. After m moves, elements $c_1, \ldots, c_m$ from $\mathfrak{A}$ and $d_1, \ldots, d_m$ from $\mathfrak{B}$ are chosen. Duplicator

wins at a final position $G_0(\mathfrak{A}, \overline{a}, c_1, \ldots, c_m, \mathfrak{B}, \overline{b}, d_1, \ldots, d_m)$ if $\mathfrak{A}, \overline{a}, \overline{c} \equiv_0 \mathfrak{B}, \overline{b}, \overline{d}$. Otherwise Spoiler wins.

A *winning strategy* of Spoiler is a function which determines, for every reachable position, a move such that Spoiler wins each play which is consistent with this strategy, no matter how Duplicator plays. Winning strategies for Duplicator are defined analogously. We say that *Spoiler (respectively, Duplicator) wins the game* $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$ if this player has a winning strategy for $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$. By induction on the number of moves it is easy to show that for every (sub)game exactly one of the two players has a winning strategy.

Theorem 3.4 (Ehrenfeucht, Fraïssé)**.** Let $\mathfrak{A}, \mathfrak{B}$ be τ-structures (recall, τ is finite and relational), let $\overline{a} \in A^k$ and $\overline{b} \in B^k$ and let $m \geq 0$. Then the following statements are equivalent:

(i) $\mathfrak{A}, \overline{a} \equiv_m \mathfrak{B}, \overline{b}$.
(ii) $\mathfrak{A}, \overline{a} \simeq_m \mathfrak{B}, \overline{b}$.
(iii) $\mathfrak{B} \models \chi^m_{\mathfrak{A},\overline{a}}(\overline{b})$.
(iv) Duplicator wins $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$.

Proof. Since $\mathfrak{A} \models \chi^m_{\mathfrak{A},\overline{a}}(\overline{a})$ and since $\mathrm{qr}(\chi^m_{\mathfrak{A},\overline{a}}) \leq m$, we have that $(i) \Rightarrow (iii)$. By Lemma 3.3, $(ii) \Leftrightarrow (iii)$. Recall from the introductory course that $(iv) \Rightarrow (ii)$. The proof strategy was to show, by induction on the quantifier rank $m \geq 0$, that if a formula $\varphi(\overline{x})$ of quantifier rank m can distinguish between $\mathfrak{A}, \overline{a}$ and $\mathfrak{B}, \overline{b}$, then we can extract a winning strategy for Spoiler from this formula for the game $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$.

Hence, it suffices to prove $(ii) \Rightarrow (iv)$. Let $(I_i)_{i \leq m} : (\mathfrak{A}, \overline{a}) \simeq_m (\mathfrak{B}, \overline{b})$. For $m = 0$ the claim holds, since $\overline{a} \to \overline{b} \in I_m \subseteq \mathrm{Part}(\mathfrak{A}, \mathfrak{B})$. For $m > 0$ assume that the Spoiler at position $G_m(\mathfrak{A}, \overline{a}, \mathfrak{B}, \overline{b})$ picks an element $a' \in A$. By the forth property Duplicator can pick $b' \in B$ such that $(\overline{a}, a') \to (\overline{b}, b') \in I_{m-1}$. Hence, $(I_i)_{i \leq m-1} : (\mathfrak{A}, \overline{a}, a') \simeq_{m-1} (\mathfrak{B}, \overline{b}, b')$. By the induction hypothesis, Duplicator wins $G_{m-1}(\mathfrak{A}, \overline{a}, a', \mathfrak{B}, \overline{b}, b')$. If Spoiler picks an element $b' \in B$ the reasoning is analogous using the back property. Q.E.D.

Corollary 3.5. For all $k \geq 0$, the relation $\equiv_m$ induces an equivalence relation on pairs $(\mathfrak{A}, \overline{a})$ of τ-structures $\mathfrak{A}$ and $\overline{a} \in A^k$ of finite index.

Corollary 3.6. A class $\mathcal{K}$ of τ-structures is FO-definable if, and only if, there exists $m \geq 0$ such that for all τ-structures $\mathfrak{A}$ and $\mathfrak{B}$ with $\mathfrak{A} \equiv_m \mathfrak{B}$ it holds that $\mathfrak{A} \in \mathcal{K} \Leftrightarrow \mathfrak{B} \in \mathcal{K}$.

3.2 Hanf's technique

Describing winning strategies in Ehrenfeucht-Fraïssé games can be difficult. In this section we want to establish sufficient criteria for structures $\mathfrak{A}$ and $\mathfrak{B}$ which guarantee that Duplicator has a winning strategy in the game $G_m(\mathfrak{A}, \mathfrak{B})$. The following approach goes back to Hanf who gave a similar criterion to characterise $\equiv$ (equivalance in full first-order logic). However, since we are mainly interested in properties of *finite* structures, $\equiv$ is far too powerful (two finite structures $\mathfrak{A}, \mathfrak{B}$ are isomorphic if, and only if, $\mathfrak{A} \equiv \mathfrak{B}$).

Gaifman graph. Let $\mathfrak{A}$ be a τ-structure. The *Gaifman-graph* $\mathcal{G}(\mathfrak{A}) = (V^{\mathcal{G}(\mathfrak{A})}, E^{\mathcal{G}(\mathfrak{A})})$ of $\mathfrak{A}$ is defined as the undirected graph over the vertex set $V^{\mathcal{G}(\mathfrak{A})} = A$ with the edge relation

$$E^{\mathcal{G}(\mathfrak{A})} = \{(a, b) : a \neq b \text{ and the elements } a, b \text{ occur together in some tuple } \bar{c} \in R^{\mathfrak{A}} \text{ for a relation } R \in \tau\}.$$

The Gaifman graph allows us to define a notion of distance between the elements of the structure $\mathfrak{A}$: we define $d^{\mathfrak{A}} : A^2 \to \mathbb{N} \cup \{\infty\}$ as the usual distance function in the Gaifman graph $\mathcal{G}(\mathfrak{A})$ of $\mathfrak{A}$.

Let $r \geq 0$. The *r-neighbourhood* of an element $a \in A$ is the set $N^r_{\mathfrak{A}}(a) = N^r(a) = \{b \in A : d^{\mathfrak{A}}(a, b) \leq r\}$. In particular, $N^0(a) = \{a\}$. For a tuple $\bar{a} = (a_1, \ldots, a_k) \in A^k$ we set

$$N^r(\bar{a}) = \bigcup_{1 \leq i \leq k} N^r(a_i).$$

The *r-isomorphism type* of an element $a \in A$ is the isomorphism type ι of the structure $(\mathfrak{A} \restriction N^r(a), a)$ (that is of the substructure of $\mathfrak{A}$ induced on the r-neighbourhood of a extended by a new constant symbol to distinguish the element a). This means that for τ-structures $\mathfrak{A}, \mathfrak{B}$, two

elements $a \in A$ and $b \in B$ have the same *r-isomorphism type* if there is an isomorphism $\pi : \mathfrak{A} \restriction N^r(a) \to \mathfrak{B} \restriction N^r(b)$ with $\pi(a) = b$.

Definition 3.7. Let $r \geq 0$ and $t \geq 0$. Two τ-structures $\mathfrak{A}$ and $\mathfrak{B}$ are (r,t)*-Hanf equivalent* if for all isomorphism types ι of structures $(\mathfrak{C}, c)$ (where $\mathfrak{C}$ is a τ-structure and $c \in C$ is a distinguished constant) the number of $a \in A$ with r-isomorphism type ι is the same as the number of $b \in B$ with r-isomorphism type ι or both numbers exceed the *threshold* t.

Remark 3.8. If $\mathfrak{A}$ and $\mathfrak{B}$ are (r,t)-Hanf equivalent, then they also are (r',t)-Hanf equivalent for all $r' \leq r$.

Theorem 3.9 (Hanf's Theorem)**.** Let $m \geq 0$ and let $\mathfrak{A}$ and $\mathfrak{B}$ be two τ-structures such that all 3^m-neighbourhoods in $\mathfrak{A}$ and $\mathfrak{B}$ have at most $e \geq 0$ many elements.

If $\mathfrak{A}$ and $\mathfrak{B}$ are $(3^m, m \cdot e)$-Hanf equivalent, then $\mathfrak{A} \equiv_m \mathfrak{B}$.

Proof. For $i \geq 0$ we obtain a back-and-forth system for m-equivalence of $\mathfrak{A}$ and $\mathfrak{B}$ by setting

$$I_{m-i} = \{\overline{a} \to \overline{b} \in \mathrm{Part}(\mathfrak{A}, \mathfrak{B}) : |\overline{a}| = |\overline{b}| = i, \\ \mathfrak{A} \restriction N^{3^{m-i}}(\overline{a}), \overline{a} \cong \mathfrak{B} \restriction N^{3^{m-i}}(\overline{b}), \overline{b}\}.$$

We have $I_m = \{\emptyset\}$, so let $i \geq 1$. Without loss of generality, it suffices to show that I_{m-i} has forth-extensions in I_{m-i-1}. Let $\overline{a} = (a_1, \ldots, a_i)$ and $\overline{b} = (b_1, \ldots, b_i)$ and ρ be such that $\rho : \mathfrak{A} \restriction N^{3^{m-i}}(\overline{a}), \overline{a} \cong \mathfrak{B} \restriction N^{3^{m-i}}(\overline{b}), \overline{b}$. Let $a \in A$. We have to find $b \in B$ such that $\mathfrak{A} \restriction N^{3^{m-i-1}}(\overline{a}, a), \overline{a}, a \cong \mathfrak{B} \restriction N^{3^{m-i-1}}(\overline{b}, b), \overline{b}, b$.

Case 1 (close to $\overline{a}$). If $a \in N^{2 \cdot 3^{m-i-1}}(\overline{a})$, then we choose $b = \rho(a) \in N^{2 \cdot 3^{m-i-1}}(\overline{b})$. This is a valid choice since we have $\rho : \mathfrak{A} \restriction N^{3^{m-i}}(\overline{a}), \overline{a}, a \cong \mathfrak{B} \restriction N^{3^{m-i}}(\overline{b}), \overline{b}, b$.

Case 2 (far from $\overline{a}$). If $a \notin N^{2 \cdot 3^{m-i-1}}(\overline{a})$, then $N^{3^{m-i-1}}(a) \cap N^{3^{m-i-1}}(a_j) = \emptyset$ for all $1 \leq j \leq i$. Hence, it suffices to find $b \in B$ with the same 3^{m-i-1}-isomorphism type as a (call this ι) and the property that $N^{3^{m-i-1}}(b) \cap N^{3^{m-i-1}}(b_j) = \emptyset$ for all $1 \leq j \leq i$.

We know that in $\mathfrak{A}$ and $\mathfrak{B}$ there are the same numbers of realisations of ι or more than $m \cdot e$ many. By our assumption, we know that in $N^{2 \cdot 3^{m-i-1}}(\overline{a})$ there are at most $m \cdot e$ realisations, and the same number of

realisations can be found in $N^{2\cdot 3^{m-i-1}}(\overline{b})$ (because of ρ). Hence, we can find a $b \in B$ as claimed. Q.E.D.

Corollary 3.10. Let $m \geq 0$ and let $\mathfrak{A}$ and $\mathfrak{B}$ be τ-structures such that the maximal degree in the Gaifman graphs $\mathcal{G}(\mathfrak{A})$ and $\mathcal{G}(\mathfrak{B})$ is $d \geq 0$. If $\mathfrak{A}$ and $\mathfrak{B}$ are $(3^m, m \cdot d^{3^m})$ equivalent, then $\mathfrak{A} \equiv_m \mathfrak{B}$.

Corollary 3.11. Connectivity of finite graphs is not definable in first-order logic.

Proof. Let $\mathfrak{A}_n$ be a cycle of length $2n$ and let $\mathfrak{B}_n$ be the disjoint union of two cycles of length n. For m we can set $n = 3^{m+1}$. Then $\mathfrak{A}_n$ and $\mathfrak{B}_n$ are $(3^m, \infty)$-Hanf equivalent but $\mathfrak{A}_n$ is connected while $\mathfrak{B}_n$ is not.

Q.E.D.

3.3 Gaifman's Theorem

Hanf's technique shows that first-order logic can essentially express local properties only: if two structures realise the same number of $f(m)$-neighbourhood types, then no first-order sentence with quantifier rank $\leq m$ can distinguish between both structures. Gaifman's Theorem makes this observation more precise by showing that every FO-sentence is equivalent to an FO-sentence which only speaks about neighbourhoods of elements of a bounded radius (and this semantic property is guaranteed by the syntactic structure of the sentence). To formally introduce this *Gaifman normal form* for first-order logic we first have to introduce the notions of *local formulas* and *local sentences*.

First of all, for every $r \geq 0$ we can find an FO-formula $\vartheta_{\leq r}(x, y)$ which defines in each structure $\mathfrak{A}$ the pairs of elements $(a, b) \in A^2$ whose distance in the Gaifman graph $\mathcal{G}(\mathfrak{A})$ of $\mathfrak{A}$ is at most r, that is

$$\vartheta^{\mathfrak{A}}_{\leq r} = \{(a, b) : d^{\mathfrak{A}}(a, b) \leq r\}.$$

In formulas we will usually write $d(x, y) \leq r$ as a shorthand for $\vartheta_{\leq r}(x, y)$. Also we write $d(\overline{x}, y) \leq r$ for a tuple of variables

$\overline{x} = (x_1, \ldots, x_k)$ to abbreviate the formula

$$d(\overline{x}, y) \leq r = \bigvee_{1 \leq i \leq k} d(x_i, y) \leq r.$$

Local formulas. A formula $\varphi(\overline{x})$ is r-local if its evaluation in a structure $\mathfrak{A}$ with respect to a tuple $\overline{a} \in A^k$ only depends on the r-neighbourhood of $\overline{a}$. To capture this formally, we inductively define the *relativisation* $\varphi^{N^r(\overline{x})}(\overline{x}, \overline{y})$ of a formula $\varphi(\overline{x}, \overline{y})$ to the r-neighbourhood $N^r(\overline{x})$ of $\overline{x}$ (for the construction we assume that no variable in $\overline{x}$ is bound in φ):

$$\begin{aligned}
\varphi^{N^r(\overline{x})} &= \varphi \quad \text{for atomic formulas } \varphi \\
\varphi^{N^r(\overline{x})} &= \psi^{N^r(\overline{x})} \circ \vartheta^{N^r(\overline{x})} \quad \text{for } \varphi = \psi \circ \vartheta, \circ \in \{\wedge, \vee\} \\
\varphi^{N^r(\overline{x})} &= \neg \psi^{N^r(\overline{x})} \quad \text{for } \varphi = \neg\psi \\
\varphi^{N^r(\overline{x})} &= \exists z (d(\overline{x}, z) \leq r \wedge \psi^{N^r(\overline{x})}) \quad \text{for } \varphi = \exists z \psi \\
\varphi^{N^r(\overline{x})} &= \forall z (d(\overline{x}, z) \leq r \rightarrow \psi^{N^r(\overline{x})}) \quad \text{for } \varphi = \forall z \psi
\end{aligned}$$

Lemma 3.12. For all $r \geq 0$, $\mathfrak{A}$, $\overline{a} \in A^k$ and $\overline{b} \in (N^r(\overline{a}))^\ell$ we have

$$\mathfrak{A} \restriction N^r(\overline{a}) \models \varphi(\overline{a}, \overline{b}) \quad \Leftrightarrow \quad \mathfrak{A} \models \varphi^{N^r(\overline{x})}(\overline{a}, \overline{b}).$$

Definition 3.13. A formula $\varphi(\overline{x})$ is called *r-local* if $\varphi(\overline{x}) \equiv \varphi^{N^r(\overline{x})}(\overline{x})$, that is if for all $\mathfrak{A}$ and $\overline{a} \in A^k$ we have

$$\mathfrak{A} \models \varphi(\overline{a}) \Leftrightarrow \mathfrak{A} \models \varphi^{N^r(\overline{x})}(\overline{a}) \Leftrightarrow \mathfrak{A} \restriction N^r(\overline{a}) \models \varphi(\overline{a}).$$

Note that r-locality is a semantic property of formulas. However, it is easy to see that all formulas $\varphi^{N^r(\overline{x})}(\overline{x})$ are r-local (in other words, the syntatic transformations guarantee that we obtain a local formula, but of course there are local formulas which do not have this syntactic form). Moreover, it is not hard to verify that every formula $\varphi(\overline{x})$ which is r-local is also r'-local for all $r' \geq r$. For a formula $\varphi(\overline{x})$ we write $\varphi^r(\overline{x}) = \varphi^{N^r(\overline{x})}(\overline{x})$ to denote the r-local version of the formula $\varphi(\overline{x})$.

Local sentences. An ℓ-tuple of elements $\bar{a} = (a_1, \ldots, a_\ell) \in A^\ell$ in a structure $\mathfrak{A}$ is called *r-scattered* if $d(a_i, a_j) > 2r$ for all a_i and a_j, $i \neq j$, that is if the r-neighbourhoods $N^r(a_i)$, $1 \leq i \leq \ell$, are pairwise disjoint. A *basic local sentence* of *Gaifman rank* (r, m, ℓ) is a sentence of the form

$$\exists x_1 \cdots \exists x_\ell \left(\bigwedge_{i \neq j} d(x_i, x_j) > 2r \wedge \bigwedge_i \psi^r(x_i) \right),$$

where $\mathrm{qr}(\psi) = m$, which expresses the existence of an r-scattered tuple of length ℓ such that every point in this tuple satisfies an r-local property which is specified by a formula ψ of quantifier-rank m. A *local sentence* is Boolean combination of basic local sentences.

Theorem 3.14 (Gaifman)**.** Every first-order sentence is equivalent to a local sentence.

To prove Gaifman's Theorem it suffices to show the following lemma.

Lemma 3.15. If $\mathfrak{A}$ and $\mathfrak{B}$ satisfy the same basic local sentences, then $\mathfrak{A} \equiv \mathfrak{B}$.

Proof (of Gaifman's Theorem using the preceeding lemma). Let Φ denote the set of all basic local sentences. Let φ be an FO-sentence and let $\mathcal{K} = \mathrm{Mod}(\varphi)$ be the class of models of φ. For $\mathfrak{A} \in \mathcal{K}$ we define

$$\Phi(\mathfrak{A}) = \{\varphi : \varphi \in \Phi, \mathfrak{A} \models \varphi\} \cup \{\neg\varphi : \varphi \in \Phi, \mathfrak{A} \models \neg\varphi\}$$

Then for all $\mathfrak{A} \in \mathcal{K}$ we have $\Phi(\mathfrak{A}) \models \varphi$, because if $\mathfrak{B} \models \Phi(\mathfrak{A})$, then $\mathfrak{A}$ and $\mathfrak{B}$ agree on all sentences from Φ and thus, by the preceeding lemma, we have that $\mathfrak{A} \equiv \mathfrak{B}$. By the compactness theorem, we can find finite sets $\Phi_0(\mathfrak{A}) \subseteq \Phi(\mathfrak{A})$ such that $\Phi_0(\mathfrak{A}) \models \varphi$ for all $\mathfrak{A} \in \mathcal{K}$.

We claim that for a finite subclass $\mathcal{K}_0 \subseteq \mathcal{K}$, the sentence φ is equivalent to $\bigvee_{\mathfrak{A} \in \mathcal{K}_0} \bigwedge \Phi_0(\mathfrak{A})$ (which is a local sentence). We know that $\bigvee_{\mathfrak{A} \in \mathcal{K}_0} \bigwedge \Phi_0(\mathfrak{A}) \models \varphi$, so assume that for every finite subclass of structures $\mathcal{K}_0 \subseteq \mathcal{K}$ the set $\{\varphi\} \cup \{\neg \bigwedge \Phi_0(\mathfrak{A}) : \mathfrak{A} \in \mathcal{K}_0\}$ would be satisfiable. Then, by compactness, also $\{\varphi\} \cup \{\neg \bigwedge \Phi_0(\mathfrak{A}) : \mathfrak{A} \in \mathcal{K}\}$ would be satisfiable which is impossible since $\mathfrak{A} \models \bigwedge \Phi_0(\mathfrak{A})$ for all $\mathfrak{A} \in \mathcal{K}$. Q.E.D.

Proof (of Lemma 3.15). For all $m \geq 0$, we prove by induction on $j =$

$m, \dots, 0$ that one can find values $g(0), g(1), \dots, g(m)$ such that

$$I_j = \{\overline{a} \to \overline{b} : |\overline{a}| = |\overline{b}| = m - j, (\mathfrak{A} \restriction N^{7^j}(\overline{a}), \overline{a}) \equiv_{g(j)} (\mathfrak{B} \restriction N^{7^j}(\overline{b}), \overline{b})\}$$

defines a back-and-forth system for m-equivalence of $\mathfrak{A}$ and $\mathfrak{B}$. Sufficient criteria for the values $g(0), \dots, g(m)$ are collected in the course of the proof (and it will be obvious that we can find values which satisfy all contraints). Note that $I_m = \{\emptyset\}$.

Let $0 \leq j < m$ and let $\overline{a} \to \overline{b} \in I_{j+1}$. Then we know that

$$(\mathfrak{A} \restriction N^{7^{j+1}}(\overline{a}), \overline{a}) \equiv_{g(j+1)} (\mathfrak{B} \restriction N^{7^{j+1}}(\overline{b}), \overline{b}).$$

By symmetry, it suffices to show that $\overline{a} \to \overline{b}$ has a forth-extension in I_j. Let $a \in A$. We have to find $b \in B$ such that

$$(\mathfrak{A} \restriction N^{7^j}(\overline{a}a), \overline{a}a) \equiv_{g(j)} (\mathfrak{B} \restriction N^{7^j}(\overline{b}b), \overline{b}b).$$

To this end we consider the $g(j)$-types of the 7^j-neighbourhoods of tuples in $\mathfrak{A}$ and $\mathfrak{B}$. Recall from Lemma 3.3 that we can describe these types by a first-order formula. More precisely, for a structure $\mathfrak{D}$ and a tuple $\overline{d}$ in $\mathfrak{D}$ we set

$$\psi^j_{\overline{d}}(\overline{x}) = \left[\chi^{g(j)}_{(\mathfrak{D} \restriction N^{7^j}(\overline{d}), \overline{d})}(\overline{x})\right]^{N^{7^j}(\overline{x})}.$$

Then $\psi^j_{\overline{d}}(\overline{x})$ is a 7^j-local formula such that $\mathfrak{C} \models \psi^j_{\overline{d}}(\overline{c})$ if the 7^j-neighbourhood of $\overline{c}$ in $\mathfrak{C}$ (with distinguished tuple $\overline{c}$) is $g(j)$-equivalent to the 7^j-neighbourhood of $\overline{d}$ in $\mathfrak{D}$ (with distinguished tuple $\overline{d}$). To find an appropriate $b \in B$ we distinguish between the following cases.

Case 1 (a is close to $\overline{a}$). Assume that $a \in N^{2 \cdot 7^j}(\overline{a})$. Then

$$(\mathfrak{A} \restriction N^{7^{j+1}}(\overline{a}), \overline{a}) \models \exists z (d(\overline{a}, z) \leq 2 \cdot 7^j \wedge \psi^j_{\overline{a}a}(\overline{a}, z)).$$

We assume that the quantifier rank of this formula, which only depends on j and $g(j)$, is at most $g(j+1)$ (this gives a first condition on $g(j+1)$). But then, by our precondition, we can find $b \in N^{2 \cdot 7^j}(\overline{b})$ such that

$$(\mathfrak{B} \restriction N^{7^j}(\bar{b})) \models \psi^j_{\bar{a}a}(\bar{b}, b),$$

which implies that $\bar{a}a \to \bar{b}b \in I_j$.

Case 2 (a is far from $\bar{a}$). Assume that $a \notin N^{2\cdot 7^j}(\bar{a})$. Then the 7^j-neighbourhoods of a and $\bar{a}$ are disjoint, i.e. $N^{7^j}(\bar{a}) \cap N^{7^j}(a) = \emptyset$. Hence it suffices to find a $b \in B$ whose 7^j-neighbourhood is disjoint with the 7^j-neighbourhood of $\bar{b}$ and such that the 7^j-neighbourhood of a in $\mathfrak{A}$ and of b in $\mathfrak{B}$ have the same $g(j)$-type. Formally the requirements for $b \in B$ are:

$$N^{7^j}(\bar{b}) \cap N^{7^j}(b) = \emptyset$$
$$\mathfrak{B} \restriction N^{7^j}(b) \models \psi^j_a(b).$$

For $s \geq 1$ we define a formula $\delta_s(x_1, \ldots, x_s)$ which expresses the existence of a $2 \cdot 7^j$-scattered tuple of elements whose 7^j-neighbourhood has the same $g(j)$-type as the 7^j-neighbourhood of a in $\mathfrak{A}$:

$$\delta_s(x_1, \ldots, x_s) = \bigwedge_{\ell \neq k} d(x_\ell, x_k) > 4 \cdot 7^j \wedge \bigwedge_k \psi^j_a(x_k).$$

We now determine the maximal lenght e of such tuples which are realised in $\mathfrak{A}$ and the maximal lenght i of such tuples which are realised in $\mathfrak{A} \restriction N^{2\cdot 7^j}(\bar{a})$, that is i and e are determined such that

$$(\mathfrak{A} \restriction N^{7^{j+1}}, \bar{a}) \models \exists x_1 \cdots \exists x_i \, (\bigwedge_k d(\bar{a}, x_k) \leq 2 \cdot 7^j \wedge \delta_i) \tag{3.1}$$

$$(\mathfrak{A} \restriction N^{7^{j+1}}, \bar{a}) \not\models \exists x_1 \cdots \exists x_{i+1} \, (\bigwedge_k d(\bar{a}, x_k) \leq 2 \cdot 7^j \wedge \delta_{i+1}) \tag{3.2}$$

$$\mathfrak{A} \models \exists x_1 \cdots \exists x_e \, \delta_e \tag{3.3}$$

$$\mathfrak{A} \not\models \exists x_1 \cdots \exists x_{e+1} \, \delta_{e+1}. \tag{3.4}$$

Of course, $i \leq e$. Moreover, $i \leq m - j = |\bar{a}| = |\bar{b}|$. We claim that the corresponding values determined in $\mathfrak{B}$ are the same. For 3.1 and 3.2 we guarantee this by choosing $g(j+1)$ large enough. Note that the quantifier rank of the formulas in 3.1 and 3.2 only depends on m (because i is bounded by m), j and $g(j)$ (we obtain a second condition on $g(j+1)$). For 3.3 and 3.4 this follows since these are basic local sentences and $\mathfrak{A}$

and $\mathfrak{B}$ satisfy the same basic local sentences by our assumption.
Case 2.1 ($i = e$). Then we claim that *all* $c \in A$ whose 7^j-neighbourhood has the same $g(j)$-type as a are contained in $N^{6\cdot 7^j}(\bar{a})$. Indeed, we could extend each $2 \cdot 7^j$-scattered tuple of such elements in $N^{2\cdot 7^j}(\bar{a})$ by each such element $c \in A$ with $d(\bar{a}, c) > 6 \cdot 7^j$. Since $a \notin N^{2\cdot 7^j}(\bar{a})$ we have

$$(\mathfrak{A} \restriction N^{7^{j+1}}(\bar{a}), \bar{a}) \models \exists z\, (2 \cdot 7^j < d(\bar{a}, z) \leq 6 \cdot 7^j \wedge \psi_a^j(z) \wedge \psi_{\bar{a}}^j(\bar{a})).$$

We assume that $g(j+1)$ is larger than the quantifier rank of this formula (this gives a third condition on $g(j+1)$). Then by our assumption we have that

$$(\mathfrak{B} \restriction N^{7^{j+1}}(\bar{b}), \bar{b}) \models \exists z\, (2 \cdot 7^j < d(\bar{b}, z) \leq 6 \cdot 7^j \wedge \psi_a^j(z) \wedge \psi_{\bar{a}}^j(\bar{b})).$$

This in turn shows that we can find an appropriate $b \in B$.

Case 2.2 ($i < e$). In this case we know that $\mathfrak{B} \models \exists x_1 \cdots \exists x_{i+1} \delta_{i+1}$ which implies that we can find $b \in B$ such that $N^{7^j}(\bar{b}) \cap N^{7^j}(b) = \emptyset$ and such that $\mathfrak{B} \models \psi_a^j(b)$. Q.E.D.

3.4 Lower bound for the size of local sentences

Gaifman's Theorem states that for every FO-sentence there is an equivalent local one. In the following we show that the local sentence can be much longer than the original one, as captured by

Theorem 3.16. For every $h \geq 1$ there is an FO(E)-sentence $\varphi_h \in \mathcal{O}(h^4)$ such that every FO(E)-sentence in Gaifman normal form, i.e. every local sentence, that is equivalent to φ_h has size at least $\mathit{Tower}(h)$.

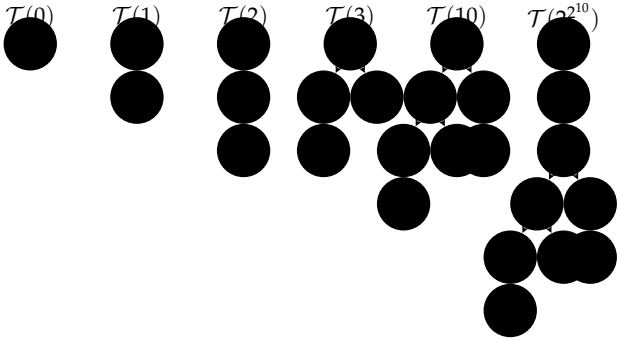
Here, $\mathit{Tower}\colon \mathbb{N} \to \mathbb{N}$ is the function defined by $\mathit{Tower}(0) := 1$ and $\mathit{Tower}(n) := 2^{\mathit{Tower}(n-1)}$ for $n > 0$. In order to prove this theorem we first introduce and analyse an encoding of natural numbers by trees.

Definition 3.17. For natural numbers i, n we write $\mathit{bit}(i, n)$ to denote the i-th bit in the binary representation of n, i.e., $\mathrm{bit}(i, n) = 0$ if $\lfloor \frac{n}{2^i} \rfloor$ is even, and $\mathrm{bit}(i, n) = 1$ if $\lfloor \frac{n}{2^i} \rfloor$ is odd. Inductively we define a directed and rooted tree $\mathcal{T}(n)$ for each natural number n as follows:

- $\mathcal{T}(0)$ is the one-node tree.

- For $n > 0$ the tree $\mathcal{T}(n)$ is obtained by creating a new root and attaching to it all trees $\mathcal{T}(i)$ for all i such that $\text{bit}(i, n) = 1$.

  The following figure illustrates these trees.



It is straightforward to see that

$$\text{for all } h, n \geq 0, \quad \text{height}(\mathcal{T}(n)) \leq h \iff n < \mathit{Tower}(h).$$

Recall that the height of a tree is the length of its longest path.

For a graph $G = (V, E)$ and some node $v \in V$, let G_v be the subgraph induced on the set of nodes reachable from v. Now, we show that important properties of these tree encodings of natural numbers can be expressed by small FO(E)-formulas in the sense of the following three Lemmata.

Lemma 3.18. For each $h \geq 0$ there is a formula $eq_h(x, y) \in \text{FO}(E)$ of length $\mathcal{O}(h)$ such that for all graphs $G = (V, E)$ we have that: if there are $u, v \in V$ and $m, n < \mathit{Tower}(h)$ with $G_u \cong \mathcal{T}(n)$ and $G_v \cong \mathcal{T}(m)$, then $G \models eq_h(u, v) \Leftrightarrow n = m$.

Proof.
- If $h = 0$, set $eq_h(x, y) := \mathit{true}$.
- If $h > 0$, $eq_h(x, y)$ has to be equivalent to

$$\forall z(Exz \to \exists w(Eyw \wedge eq_{h-1}(z, w))) \wedge$$
$$\forall w(Eyw \to \exists z(Exz \wedge eq_{h-1}(z, w))).$$

The length of the formula we get by this recursive definition would

be exponential in h. However, we can rewrite it as follows:

$$\begin{aligned} eq_h(x,y) :=& (\exists z Exz \leftrightarrow \exists w Eyw) \wedge \\ & \forall z(Exz \rightarrow \exists w(Eyw \wedge \forall w'(Eyw' \rightarrow \exists z'(Exz' \wedge \\ & \forall u \forall v((u = z \wedge v = w) \vee (u = z' \wedge v = w') \rightarrow \\ & eq_{h-1}(u,v)))))). \end{aligned}$$

Q.E.D.

Lemma 3.19. For $h \geq 0$ there is a formula $code_h(x) \in \mathrm{FO}(E)$ of length $\mathcal{O}(h^2)$ such that for all graphs $G = (V, E)$ and $v \in V$:

$$G \models code_h(v) \iff G_v \cong \mathcal{T}(i) \text{ for some } i < Tower(h).$$

Proof.
- If $h = 0$, set $code_h(x) := \neg \exists y Exy$.
- If $h > 0$, set

$$\begin{aligned} code_h(x) :=& \forall y(Exy \rightarrow code_{h-1}(y)) \wedge \\ & \forall y_1 \forall y_2(Exy_1 \wedge Exy_2 \wedge eq_{h-1}(y_1, y_2) \rightarrow y_1 = y_2). \end{aligned}$$

Observe that

$$\begin{aligned} \|code_h(x)\| &= \|code_{h-1}(x)\| + \|eq_{h-1}(x,y)\| + \mathcal{O}(1) \\ &\leq c \cdot (1 + 2 + \cdots + h) \text{ for some } c \geq 1, \end{aligned}$$

implying that $\|code_h(x)\| \in \mathcal{O}(h^2)$.

Q.E.D.

Lemma 3.20. For $h \geq 0$ there are formulas

(1) $bit_h(x,y)$ of length $\mathcal{O}(h)$,
(2) $less_h(x,y)$ of length $\mathcal{O}(h^2)$,
(3) $min(x)$ of length $\mathcal{O}(1)$,
(4) $succ_h(x,y)$ of length $\mathcal{O}(h^3)$,
(5) $max_h(x)$ of length $\mathcal{O}(h^4)$,

such that for all $G = (V, E)$ and nodes $u, v \in V$ with $G_u \cong \mathcal{T}(m)$ and $G_v \cong \mathcal{T}(n)$, where $m, n < Tower(h)$:

(1) $G \models bit_h(u,v) \iff bit(m,n) = 1$,
(2) $G \models less_h(u,v) \iff m < n$,
(3) $G \models min(u) \iff m = 0$,
(4) $G \models succ_h(u,v) \iff m+1 = n$,
(5) $G \models max_h(u) \iff m = Tower(h) - 1$.

Proof. (1) $bit_h(x,y) := \exists z(Eyz \wedge eq_h(x,z))$,

(2)
- If $h = 0$, set $less_h(x,y) := false$.
- If $h > 0$, set

$$\begin{aligned} less_h(x,y) :=& \exists y'(Eyy' \wedge \forall x'(Exx' \to \neg eq_{h-1}(x',y')) \wedge \\ & \forall x''(Exx'' \wedge less_{h-1}(y',x'') \to \\ & \exists y''(Eyy'' \wedge eq_{h-1}(y'',x''))) \end{aligned}$$

(3) $min(x) := \neg\exists yExy$.

(4)
- If $h = 0$, set $succ_h(x,y) := false$.
- If $h > 0$, set

$$\begin{aligned} succ_h(x,y) =& \exists y'(Eyy' \wedge \\ & \forall y''(Eyy'' \wedge y' \neq y'' \to less_{h-1}(y',y'') \wedge \\ & \forall x'(Exx' \to \neg eq_{h-1}(x',y') \wedge \\ & \forall y''(Eyy'' \wedge less_{h-1}(y',y'') \to \\ & \exists x''(Exx'' \wedge eq_{h-1}(y'',x'))) \wedge \\ & \forall x''(Exx'' \wedge less_{h-1}(y',x'') \to \\ & \exists y''(Eyy'' \wedge eq_{h-1}(y'',x''))) \wedge \\ & \neg min(y') \to (\exists x'(Exx' \wedge min(x')) \wedge \\ & \forall x'(Exx' \wedge less_{h-1}(x',y') \to \\ & \exists z(succ_{h-1}(x',z) \wedge (z = y' \vee Exz)))). \end{aligned}$$

(5)
- If $h = 0$, set $max_h(x) := \neg\exists yExy$.
- If $h > 0$, set

$$\begin{aligned} max_h(x) :=& \exists y(Exy \wedge min(y)) \wedge \forall y(Exy \to \\ & (max_{h-1}(y) \vee \exists z(Exz \wedge succ_{h-1}(y,z))). \end{aligned}$$

This formula is correct since $x = Tower(h) - 1 = 2^{Tower(h-1)} - 1$ implies that $\mathcal{T}(Tower(h) - 1)$ has a subtree $\mathcal{T}(i)$ for any $i \leq Tower(h-1) - 1$.

Q.E.D.

Finally, we use these three lemmata to prove a last lemma of which Theorem 3.16 is a corollary.

Lemma 3.21. For all $h \geq 1$ there is a formula $\varphi_h \in \mathrm{FO}(E)$ with $\|\varphi_h\| \in \mathcal{O}(h^4)$ such that every local sentence ψ which is equivalent to φ_h on the class of forests of height less or equal to h has size $\|\psi\| \geq Tower(h)$.

Proof. Let F_h be the forest consisting of all trees $\mathcal{T}(i)$ with $0 \leq i < Tower(h)$ and let F_h^{-i} be the forest F_h without the tree $\mathcal{T}(i)$ for some $0 \leq i < Tower(h)$. Furthermore, $root(x) := \neg\exists y Eyx$. Now, define

$$\begin{aligned}\varphi_h :=& \exists x(root(x) \wedge min(x)) \wedge \\ & \forall x(root(x) \wedge \neg max_h(x) \rightarrow \exists y(root(y) \wedge succ_h(x,y))).\end{aligned}$$

Observe that $\|\varphi_h\| \in \mathcal{O}(h^4)$ and $F_h \models \varphi_h$ as well as $F_h^{-i} \not\models \varphi_h$ for each $0 \leq i < Tower(h)$.

Let ψ be a local sentence which is equivalent to φ_h on the class of all forests of height less or equal to h. We want to show that $\|\psi\| \geq Tower(h)$.

ψ is a Boolean combination of basic local sentences $\chi_1, \ldots, \chi_L$ with

$$\chi_\ell = \exists x_1 \ldots \exists x_{k_\ell}(\bigwedge_{i \neq j} d(x_i, x_j) > 2 \cdot r_\ell \wedge \bigwedge_i \psi_\ell^{r_\ell}(x_i)).$$

W.l.o.g. there is some $m \leq L$ such that $F_h \models \chi_\ell$ for all $\ell \leq m$ and $F_h \not\models \chi_\ell$ for all $m < \ell \leq L$. Hence we can find for all $\ell \leq m$ nodes $u_{\ell,1}, \ldots, u_{\ell,k_\ell}$ in F_h such that $F_h \models d(u_{\ell,i}, u_{\ell,j}) > 2 \cdot r_\ell \wedge \psi_\ell^{r_\ell}(u_{\ell,i})$ for all $i \neq j$. The set U consisting of all these nodes contains at most $k_1 + \cdots + k_m \leq \|\psi\|$ many nodes.

Towards a contradiction assume that $\|\psi\| < Tower(h)$. Since F_h contains $Tower(h)$ many disjoint trees, there is at least one $j < Tower(h)$ such that $\mathcal{T}(j)$ in F_h contains no U-node. We claim that $F_h^{-j} \models \psi$ (which would yield the desired contradiction).

- $F_h^{-j} \models \chi_\ell$ where $l \leq m$: the local properties around the nodes $u_{\ell,1}, \ldots, u_{\ell,k_\ell}$ also hold in F_h^{-j} since the neighbourhoods are not changed by removing the tree $T(j)$.
- $F_h^{-j} \models \chi_\ell$ where $m < \ell \leq L$: clear, since F_h^{-j} is a substructure of F_h.

Q.E.D.

4 Zero-one laws

4.1 Random graphs

We consider the class $\mathcal{G}_n$ of (undirected) graphs over $\{0, \ldots, n-1\}$, i.e.

$$\mathcal{G}_n := \{G = (V, E) : G \text{ graph}, V = \{0, \ldots, n-1\}\},$$

In order to introduce *random graphs* we consider a sequence of probability distributions $\overline{\mu} = (\mu_1, \mu_2, \ldots)$ on $(\mathcal{G}_1, \mathcal{G}_2, \ldots)$, i.e. $\mu_n : \mathcal{G}_n \to [0,1]$ and $\sum_{G \in \mathcal{G}_n} \mu(G) = 1$ for all $n \geq 1$. This defines a sequence of probability spaces $(\mathcal{G}_1, \mu_1), (\mathcal{G}_2, \mu_2), \ldots$ on classes of graphs of increasing size.
Example 4.1.

(1) The *uniform distribution* μ_n assigns equal probability to each graph:

$$\mu_n(G) = \frac{1}{2^{\binom{n}{2}}}.$$

(2) Let $p : \mathbb{N} \to [0,1]$ be an arbitrary mapping. Then the probability space $\mathcal{G}_{n,p} = (\mathcal{G}_n, \mu_{p,n})$ is defined by the following random experiment: determine for every pair (u, v) with $0 \leq u < v < n$ whether $(u, v) \in E$ using a random variable X taking values $0, 1$ (False and True) with $\Pr[X = 1] = p(n)$ and $\Pr[X = 0] = (1 - p(n))$. Observe that for $p = \frac{1}{2}$ one obtains the uniform distribution.

We make the following convention: unless otherwise stated, μ_n denotes the uniform distribution. For a class $\mathcal{K}$ of graphs we set

$$\mu_n(\mathcal{K}) := \mu_n(\mathcal{K} \cap \mathcal{G}_n) = \sum_{G \in \mathcal{K} \cap \mathcal{G}_n} \mu_n(G).$$

This definition formalises what it means that a random graph $G \in \mathcal{G}_n$ has a certain property $\mathcal{K}$. However, in what follows, we are not interested

in random graphs of some fixed size $n \in \mathbb{N}$ but much more in the behaviour of the probability $\mu_n(K)$ if we increase the size of graphs, i.e. if we let n approach infinity.

Definition 4.2. The *asymptotic probability* of a class $\mathcal{K}$ of graphs (with respect to $\overline{\mu}$) is defined as

$$\mu(\mathcal{K}) := \lim_{n\to\infty} \mu_n(\mathcal{K}),$$

in the case that this sequence has a limit. In particular, if ψ is a sentence over vocabulary $\{E\}$ in some logic $\mathcal{L}$, then the *asymptotic probability* of ψ (with respect to $\overline{\mu}$) is defined as

$$\mu(\psi) := \lim_{n\to\infty} \mu_n(\{G \in \mathcal{G}_n : G \models \psi\},$$

again only for the case that the limit exists.

Example 4.3.

(1) Let $\mathcal{K} = \{G : G \text{ is a clique}\}$. Then

$$\lim_{n\to\infty} \mu_n(\mathcal{K}) = \lim_{n\to\infty} \frac{1}{2^{\binom{n}{2}}} = 0.$$

(2) Let H be a graph and let $\mathcal{K}_H = \{G : G \text{ contains } H \text{ as subgraph}\}$. For $n > k \cdot |H|$ we have

$$\mu_n(\mathcal{K}_H) \geq 1 - (1 - (2^{-|E(H)|}))^k,$$

hence $\mu(K_H) = 1$ since $k \to \infty$ for $n \to \infty$.

(3) Let $\mathcal{K} = \{G : G \text{ is three-colourable}\}$. Then

$$\lim_{n\to\infty} \mu_n(\mathcal{K}) \leq 1 - \lim_{n\to\infty} \mu_n(\{G \in \mathcal{G}_n : G \text{ contains } K_4\}) = 0.$$

(4) Recall that we have $\lim_{n\to\infty} \mu_n(\{G : (3,17) \in E\}) = \frac{1}{2}$.

(5) The asymptotic probability is not defined for every class of graphs. For instance, consider $\mathcal{K} = \{G : G \text{ has an even number of nodes}\}$. Then the sequence $(\mu_n(\mathcal{K}))_{n\geq 1} = (0, 1, 0, 1, \dots)$ has no limit.

4.2 Zero-one law for first-order logic

In this section we prove the *zero-one law* for first-order logic:
Theorem 4.4. For sentences $\psi \in \mathrm{FO}$ (over relational vocabulary) we have

$$\mu(\psi) = 0 \quad \text{or} \quad \mu(\psi) = 1.$$

To put it in words, every first-order definable property of graphs either holds *almost never* or *almost surely* on random graphs of increasing size.
Definition 4.5. An *atomic graph k-type* is a maximal consistent set t of $\mathrm{FO}(\{E\})$-literals in variables $x_1, \ldots, x_k$, i.e. $Ex_ix_j, \neg Ex_ix_j, x_i = x_j, x_i \neq x_j$, which is consistent with the graph axioms $(\forall x \neg Exx, \forall x \forall y(Exy \leftrightarrow Eyx))$. Furthermore, for a graph $G = (V, E)$ and $\bar{a} \in V^k$ we define the *atomic graph k-type of* $\bar{a}$ by

$$t_G(\bar{a}) := \{\varphi(x_i, x_j) : \varphi \text{ an } \mathrm{FO}(\{E\})\text{-literal such that } G \models \varphi(a_i, a_j)\}.$$

Formally, an atomic k-type t is a set but we frequently identify it with the formula $t(\bar{x}) = \bigwedge_{\varphi \in t} \varphi(\bar{x})$ (this formula is an FO-formula, since there are only finitely many $\{E\}$-literals in k variables).

In what follows, let $s(\bar{x})$ and $t(\bar{x})$ denote atomic graph types of tuples of distinct elements, i.e. $s, t \models \bigwedge_{i<j\leq k} x_i \neq x_k$. We say that an atomic $(m+1)$-type $t(x_1, \ldots, x_m, x_{m+1})$ *extends* an atomic m-type $s(x_1, \ldots, x_m)$ if $s \subseteq t$, or equivalently, if $t \models s$.
Definition 4.6. Let $s(x_1, \ldots, x_m)$ and $t(x_1, \ldots, x_m, x_{m+1})$ be atomic types such that $s \subseteq t$. We define the *extension axiom* $\sigma_{s,t}$ by

$$\sigma_{s,t} := \forall x_1 \cdots \forall x_m (s(\bar{x}) \rightarrow \exists x_{m+1} t(\bar{x}, x_{m+1})).$$

Furthermore, we let T be the set of all extension axioms together with the graph axioms.

The proof of the zero-one law for FO relies on the following properties of the extension axioms and the set T:

(1) $\mu(\sigma_{s,t}) = 1$ for all $\sigma_{s,t} \in T$.
(2) T is ω-categorical, i.e. there is, up to isomorphism, only one countable model of T. This structure is known as the *Rado graph*.

(3) T is complete, i.e. for all $\psi \in \mathrm{FO}$ either $T \models \psi$ or $T \models \neg\psi$.

We proceed to establish these three properties.

Lemma 4.7. Let $\sigma_{s,t} \in T$ be an extension axiom. Then $\mu(\sigma_{s,t}) = 1$.

Proof. Let $\sigma_{s,t} := \forall x_1 \cdots \forall x_m (s(\overline{x}) \rightarrow \exists x_{m+1} t(\overline{x}, x_{m+1}))$. For every $i = 1, \ldots, m$ we have $t \models Ex_i x_{m+1}$ or $t \models \neg Ex_i x_{m+1}$. Let $G \in \mathcal{G}_n$ be a random graph and $a_1, \ldots, a_m \in \{0, \ldots, n-1\}$. For every fixed $a_{m+1} \in V \setminus \{a_1, \ldots, a_m\}$, the experiments $G \models Ea_i a_{m+1}$ are stochastically independent and have probability $\frac{1}{2}$. Hence

$$\Pr[G \models t(\overline{a}, a_{m+1}) | G \models s(\overline{a})] = \frac{1}{2^m}.$$

Thus, probability that *no* element $a_{m+1} \in V \setminus \{a_1, \ldots, a_m\}$ extends a realisation $\overline{a}$ of s to a realisation of $(\overline{a}, a_{m+1})$ of t is $(1 - \frac{1}{2^m})^{n-m}$. In conclusion, we obtain

$$\begin{aligned} \mu_n(\neg\sigma_{s,t}) &= \mu_n(\exists x_1 \cdots \exists x_n (s(\overline{x}) \wedge \forall x_{m+1} \neg t(\overline{x}, x_{m+1}))) \\ &\leq n^m \cdot (1 - \frac{1}{2^m})^{n-m} \quad \overset{\text{exp. fast}}{\longrightarrow} \quad 0, \end{aligned}$$

and thus $\mu(\sigma_{s,t}) = 1$. Q.E.D.

The compactness theorem implies that also every logical consequence of the extensions axioms almost surely holds in a random graph.

Corollary 4.8. If $T \models \psi$ then $\mu(\psi) = 1$, and the set T is satisfiable.

Proof. If $T \models \psi$, then by the compactness theorem there is a finite set $T_0 \subseteq T$ such that $T_0 \models \psi$. Hence, we have $\mu_n(\psi) \geq \mu_n(\bigwedge T_0)$. Observe that $\mu_n(\neg\varphi) = 1 - \mu_n(\varphi)$ and $\mu_n(\varphi_1 \vee \varphi_2) \leq \mu_n(\varphi_1) + \mu_n(\varphi_2)$ are true for every sentences $\varphi, \varphi_1, \varphi_2$. Furthermore, by Lemma 4.7, it follows that $\mu_n(\neg\sigma) = 1 - \mu_n(\sigma) \to 0$ for $n \to \infty$. Putting everything together, we obtain

$$\mu_n(\neg\psi) \leq \mu_n\big(\neg \bigwedge T_0\big) = \mu_n\Big(\bigvee_{\sigma \in T_0} \neg\sigma \Big) \leq \sum_{\sigma \in T_0} \mu_n(\neg\sigma)$$

and the sum on the right converges to 0 for $n \to \infty$, which implies that $\mu_n(\psi)$ converges to 1 or, to put it differently, $\mu(\psi) = 1$. Q.E.D.

Interestingly, one can give explicit description of models of T and we present two different possibilities here. However, as we show later that T is ω-categorical, these models are isomorphic.

Definition 4.9 (Rado graph)**.** The following graphs are models of T.

(1) Let p_i denote the i-th prime number. We define $G = (\mathbb{N}, E)$ with

$$E := \{(i,j) \in \mathbb{N} \times \mathbb{N} : p_i \mid j \text{ or } p_j \mid i.\}$$

We claim that $G \models T$. To see this, we choose an arbitrary extension axiom $\sigma_{s,t} := \forall x_1 \cdots \forall x_m (s(\overline{x}) \to \exists x_{m+1} t(\overline{x}, x_{m+1})) \in T$.
Let $I \cup J = \{1, \ldots, m\}$ be the partition defined by t with respect to the following condition

- If $t \models Ex_i x_{m+1}$ then $i \in I$, and
- if $t \models \neg Ex_i x_{m+1}$ then $i \in J$.

Let $a_1, \ldots, a_k \in A$ such that $G \models s(a_1, \ldots, a_k)$. We set $a_{m+1} := \prod_{i \in I} p_{a_i} q$ where q is a prime number with $q > p_{a_1} \cdots p_{a_m}$. Then it is easy to check that $G \models Ea_i a_{m+1}$ for all $i \in I$ and $G \models \neg Ea_j a_{m+1}$ for all $j \in J$.

(2) The set HF of *heriditarily finite sets* is defined by:

- $\emptyset \in \mathrm{HF}$
- If $a_1, \ldots, a_k \in \mathrm{HF}$, then also $\{a_1, \ldots, a_k\} \in \mathrm{HF}$.

Let $G = (\mathrm{HF}, E)$ with $E := \{(a,b) : a \in b \text{ or } b \in a\}$. Similarly as above, one can show that $G \models T$.

Theorem 4.10. Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be two countable models of T. Then $G \cong H$. The unique countable model of T is known as the *Rado graph* $\mathcal{R}$.

Proof. First of all, it is clear that T has no finite models, hence G and H are infinite graphs. We fix two enumerations of V_G and V_H and inductively construct a sequence of partial isomorphism $p_0, p_1, p_2, \ldots$ between G and H such that $p_0 \subseteq p_1 \subseteq p_2 \subseteq \cdots$. For the base case, we set $p_0 := \emptyset$. For the induction step let $p_i = \{(a_1, b_1), \ldots, (a_i, b_i)\} \in \mathrm{Loc}(G, H)$ be already defined. We distinguish between the following two cases:

- If i is even, choose $a_{i+1} \in V_G$ to be the minimal element (with respect to the enumeration of V_G) which is not in the domain of p_i, i.e. $a_{i+1} \notin \{a_1, \ldots, a_i\}$. Let $s := t_G(a_1, \ldots, a_i)$ and $t := t_G(a_1, \ldots, a_{i+1})$. Since p_i is a partial isomorphism we know that $H \models s(b_1, \ldots, b_i)$. Since $H \models \sigma_{s,t}$ there exists an element $b_{i+1} \in V_H$ such that $H \models t(b_1, \ldots, b_{i+1})$. We set $p_{i+1} := p_i \cup \{(a_{i+1}, b_{i+1})\}$ and obtain a partial isomorphism extending p_i.
- If i is odd, we proceed analogously, but this time we let $b_{i+1} \in V_H$ be the minimal element (with respect to the enumeration of V_H) which is not in the image of p_i, i.e. $b_{i+1} \notin \{b_1, \ldots, b_i\}$. For $s := t_H(b_1, \ldots, b_i)$ and $t := t_H(b_1, \ldots, b_{i+1})$, the same reasoning as above yields an element $a_{i+1} \in V_G$ such that $G \models t(a_1, \ldots, a_{i+1}$. Again we obtain an extended partial isomorphism by setting $p_{i+1} := p_i \cup \{(a_{i+1}, b_{i+1})\}$.

Finally we let $p := \bigcup_{i \geq 0} p_i$. By construction we have that $\mathrm{dom}(p) = V_G$ and $\mathrm{im}(p) = V_H$, hence $p : G \xrightarrow{\sim} H$. Q.E.D.

In particular, ω-categorical theories are complete:

Theorem 4.11. *T axiomatises a complete theory, i.e. for all sentences $\psi \in \mathrm{FO}(\{E\})$ we have $T \models \psi$ or $T \models \neg\psi$.*

Proof. Assume for some sentence $\psi \in \mathrm{FO}(\{E\})$ it holds that $T \not\models \psi$ and $T \not\models \neg\psi$. Then by the downwards Löwenheim-Skolem theorem, there exist two countable graphs G and H with $G \models T \cup \{\psi\}$ and $H \models T \cup \{\neg\psi\}$. In particular this implies $G \ncong H$, which contradicts Theorem 4.10. Q.E.D.

Theorem 4.12. [Glebskiĭ et al., R. Fagin] *For all $\psi \in \mathrm{FO}(\{E\})$ it holds:*

$$\mu(\psi) = 0 \quad \text{or} \quad \mu(\psi) = 1.$$

Proof. If $T \models \psi$, then $\mu(\psi) = 1$. Otherwise, $T \models \neg\psi$, and hence $\mu(\psi) = 1 - \mu(\neg\psi) = 0$. Q.E.D.

In particular, we can give a precise characterisation of those first-order properties which hold almost surely in random graphs.

Corollary 4.13. Let $\psi \in \mathrm{FO}(\{E\})$. Then

$$\mu(\psi) = 1 \quad \text{iff} \quad T \models \psi \quad \text{iff} \quad \mathcal{R} \models \psi.$$

4.2.1 Applications

We can use Theorem 4.12 to show that certain classes of graphs are not definable in first-order logic: if a class $\mathcal{K}$ of graphs has undefined asymptotic probability or an asymptotic probability different from 0 and 1, then clearly $\mathcal{K}$ cannot be defined in first-order logic. More generally, this method yields non-definability of $\mathcal{K}$ for *every* logic that has a 0-1-law, e.g. for $L^{\omega}_{\infty\omega}$ as we see later. For instance, consider the class EvenV $= \{G = (V, E) : |V| \text{ is even}\}$ with undefined asymptotic probability or the class EvenE $= \{G = (V, E) : |E| \text{ is even}\}$ with $\mu(\text{EvenE}) = \frac{1}{2}$. Moreover, we can use our results as a convenient method to determine the asymptotic probability for many natural classes of graphs.

(1) We want to determine $\mu(\text{Con})$ where Con denotes the class of connected graphs. Let s be an atomic 2-type in variables x, y containing $\neg Exy$ and let t be the atomic 3-type in variables x, y, z which extends s and contains $Exz \wedge Eyz$. Then $G \models \sigma_{s,t}$ iff G has diameter at most 2. Hence, $G \models \sigma_{s,t}$ implies $G \in \text{Con}$, which means that $\mu(\text{Con}) = 1$.

(2) Let $\mathcal{K}$ be any class of graphs which exclude a forbidden subgraph $H = (\{v_1, \ldots, v_k\}, E)$. Then $\mu(\mathcal{K}) = 0$. To see this, we set $s_i(x_1, \ldots, x_i) := t_H(v_1, \ldots, v_i)$ for $i \leq k$ and consider the extension axioms $\sigma_{s_i s_{i+1}}$. Then clearly $\psi := \bigwedge_{i<k} \sigma_{s_i s_{i+1}}$ is a logical consequence of T, which means that $\mu(\psi) = 1$. Moreover, if $G \models \psi$, then G contains H as an induced subgraph. We conclude that $\mu(\mathcal{K}) \leq 1 - \mu(\psi) = 0$. As an application, consider the class of planar graphs which exclude K_5 (the complete graph on 5 vertices) and the class of k-colourable graphs which exclude K_{k+1} (where k is fixed). To put it in words, a random graph is almost never planar nor k-colourable.

4.3 Generalised zero-one laws

In this section we want to generalise our considerations in two different ways. Firstly, instead of restricting ourselves to graphs, we want to work on more general classes of structures and analyse whether the zero-one-law for FO still holds. Secondly, as FO has rather limited expressive power, we look for zero-one laws for more powerful logics as well.

Let τ be an arbitrary vocabulary (not necessarily relational). By $\mathrm{Str}_n(\tau)$ we denote the set of all τ-structures over the universe $\{0, \dots, n-1\}$. As before we define a sequence $\overline{\mu} = (\mu_1, \mu_2, \dots)$ of uniform probability distributions $\mu_n : \mathrm{Str}_n(\tau) \to [0,1]$, i.e. for every $\mathfrak{A} \in \mathrm{Str}_n(\tau)$ we set

$$\mu_n(\mathfrak{A}) = \frac{1}{|\mathrm{Str}_n(\tau)|}.$$

We claim that $\mathrm{FO}(\tau)$ has a zero-one law if, and only if, τ contains no function symbols. To this end, we first consider the case where τ contains function symbols:

(1) Assume $\{P, c\} \subseteq \tau$ where c is a constant symbol and P a monadic relation. Then for $\psi := Pc$ we have $\mu_n(\psi) = \frac{1}{2}$ for all $n \geq 1$, hence $\mu(\psi) = \frac{1}{2}$, i.e. the zero-one law does hold in this case.

(2) Assume $f \in \tau$ where f is a unary function symbol. Consider the $\mathrm{FO}(\tau)$-sentence $\psi := \exists x(fx = x)$ stating that f has a fixed point. For $n \geq 1$ we have

$$\mu_n(\psi) = 1 - \prod_{i=0}^{n-1} \underbrace{\left(\frac{n-1}{n}\right)}_{=\Pr[f(i) \neq i]} = 1 - \left(1 - \frac{1}{n}\right)^n.$$

Since $\left(1 - \frac{1}{n}\right)^n \longrightarrow e^{-1}$ for $n \to \infty$, the zero-one law does not hold in this case either.

For the other direction, let τ be purely relational, $\tau = \{R_1, \dots, R_k\}$. The proof strategy we used over graphs generalises for this general in a straightforward way:

- An *atomic τ-type in k variables* is a maximal, consistent set of τ-

literals over variables $x_1, \dots, x_k$. For a τ-structure $\mathfrak{A}$ and $\overline{a} \in \mathfrak{A}$ we set $t_{\mathfrak{A}}(\overline{a}) = \{\varphi(\overline{x}) : \varphi \text{ a } \tau\text{-literal with } \mathfrak{A} \models \varphi(\overline{a})\}$.

- The *τ-extension axiom* $\sigma_{s,t}$ for two atomic τ-types s and t (in k and $k+1$ variables, respectively) with $s \subseteq t$ is defined as

  $$\sigma_{s,t} := \forall \overline{x}(s(\overline{x}) \to \exists x_{k+1} t(\overline{x}, x_{k+1})).$$

  As before, we let T denote the set of all τ-extension axioms
- Again we can show that $\mu(\sigma_{s,t}) = 1$ for all $\sigma_{s,t} \in T$. Let r denote the number of literals in t which contain x_{m+1}. Then, for a random structure $\mathfrak{A} \in \mathrm{Str}_n(\tau)$, $\overline{a} \in A$ and a_{m+1} it holds

  $$\Pr[\mathfrak{A} \models t(\overline{a}, a_{m+1}) \mid \mathfrak{A} \models s(\overline{a})] = 2^{-r}.$$

  Thus

  $$\begin{aligned} \mu_n(\neg\sigma_{s,t}) &= \mu_n(\exists \overline{x}(s(\overline{x}) \wedge \forall x_{m+1} \neg t(\overline{x}, x_{m+1}))) \\ &\leq n^m (1 - 2^{-r})^{n-m} \quad \overset{\text{exp. fast}}{\longrightarrow} \quad 0. \end{aligned}$$

- T is ω-categorical: analogously!

Our analysis raises the question why even basic functions but not arbitrary relations inhibit a zero-one law. The reason is that atomic experiments are not longer stochastically independent. For instance, consider the experiments $f(a) = b$ and $f(a) = c$ (for $b \neq c$), then $\Pr[f(a) = c \mid f(a) = b] = 0 \neq \Pr[f(a) = c]$.

4.3.1 Zero-one law for $L^{\omega}_{\infty\omega}$

We proceed to show that the zero-one law holds for $L^{\omega}_{\infty\omega}$ as well (restricted to relational vocabularies). In particular, since LFP $\leq L^{\omega}_{\infty\omega}$, this means that a random graph either almost surely has an LFP-definable property or almost never does. With FO^k we denote the k-variable fragment of FO, i.e. $\mathrm{FO}^k = \mathrm{FO} \cap L^k_{\infty\omega} = \{\varphi \in \mathrm{FO} : \varphi \text{ only contains variables } x_1, \dots, x_k\}$. If we restrict the set of extension axioms T to FO^k we obtain finite sets of approximations of T which are

again sentences in FO^k; more specifically, we set

$$\Theta_k := \bigwedge T \cap \mathrm{FO}^k = \bigwedge \{\sigma_{s,t} : \sigma_{s,t} \in T \cap \mathrm{FO}^k\} \in \mathrm{FO}^k.$$

The central property of these approximations for T is stated in the following theorem: in models of Θ_k, every $L^k_{\infty\omega}$-formula is equivalent to a simple Boolean combinations of atomic k-types. In particular, every $L^k_{\infty\omega}$-sentence is either true or false in all models of Θ_k.

Theorem 4.14. Let $m \leq k$, $s(x_1, \ldots, x_m)$ an atomic m-type and $\varphi(x_1, \ldots, x_m) \in L^k_{\infty\omega}$. Then

$$\begin{array}{ll} \text{either} & \Theta_k \models \forall \overline{x}(s(\overline{x}) \to \varphi(\overline{x})) \\ \text{or} & \Theta_k \models \forall \overline{x}(s(\overline{x}) \to \neg\varphi(\overline{x})). \end{array}$$

Proof. We proceed by induction on φ and simultaneously show the claim for all $m \leq k$ and atomic types s. If φ is atomic, then either $\varphi \in s$ or $\neg\varphi \in s$. If $\varphi = \neg\psi$, the claim directly follows.

Let $\varphi = \bigwedge \Psi$, $\Psi \subseteq L^k_{\infty\omega}$. By induction hypothesis for all $\psi \in \Psi$

$$\begin{array}{ll} \text{either} & \Theta_k \models \forall \overline{x}(s(\overline{x}) \to \psi(\overline{x})) \\ \text{or} & \Theta_k \models \forall \overline{x}(s(\overline{x}) \to \neg\psi(\overline{x})). \end{array}$$

If $\Theta_k \models \forall \overline{x}(s(\overline{x}) \to \psi(\overline{x}))$ for all $\psi \in \Psi$, then $\Theta_k \models \forall \overline{x}(s(\overline{x}) \to \bigwedge \Psi(\overline{x}))$. Otherwise, $\Theta_k \models \forall \overline{x}(s(\overline{x}) \to \neg \bigwedge \Psi(\overline{x}))$.

Let $\varphi(\overline{x}) = \exists y \psi(\overline{x}, y)$ and assume that $\Theta_k \not\models \forall \overline{x}(s(\overline{x}) \to \neg\varphi(\overline{x}))$. Choose a structure $\mathfrak{A} \models \Theta_k$ with $\mathfrak{A} \models \exists \overline{x}(s(\overline{x}) \wedge \exists y \psi(\overline{x}, y))$ and consider the following two cases

- If $y \notin \{x_1, \ldots, x_m\}$, i.e. $y \in \{x_{m+1}, \ldots, x_k\}$; let $a_1, \ldots, a_m, b \in A$ such that $\mathfrak{A} \models s(\overline{a}) \wedge \psi(\overline{a}, b)$. We define the atomic type $t(x_1, \ldots, x_m, y) := t_{\mathfrak{A}}(\overline{a}, b)$ with $s \subseteq t$. In particular,

  $$\mathfrak{A} \models \exists \overline{x} \exists y (t(\overline{x}, y) \wedge \psi(\overline{x}, y)).$$

  By induction hypothesis we know that

  $$\mathfrak{A} \models \forall \overline{x} \forall y (t(\overline{x}, y) \to \psi(\overline{x}, y)),$$

and since $\sigma_{s,t} = \forall\overline{x}(s(x) \to \exists y t(\overline{x}, y))$ is an extension axiom contained in Θ_k we finally obtain

$$\mathfrak{A} \models \forall\overline{x}(s(\overline{x}) \to \exists y \psi(\overline{x}, y)).$$

- If $y \in \{x_1, \ldots, x_m\}$, i.e. $y = x_j$ for $j \leq m$; let $\overline{a} \in A$ such that $\mathfrak{A} \models s(\overline{a}) \wedge \exists x_j \psi(\overline{a})$, and let $\overline{x}^*$ and $\overline{a}^*$ denote the tuples $\overline{x}$ and $\overline{a}$ without the j-th componenent, i.e.

$$\overline{x}^\star := x_1 \cdots x_{j-1} x_{j+1} \cdots x_k$$
$$\overline{a}^\star := a_1 \cdots a_{j-1} a_{j+1} \cdots a_k.$$

Similarly, let $s^\star(\overline{x}^\star) := t_{\mathfrak{A}}(\overline{a}^\star)$ be the atomic type of $\overline{a}^\star$ in $\mathfrak{A}$. Then $s^\star \subseteq s$ and there is $b \in A$ such that

$$\mathfrak{A} \models s^\star(\overline{a}^\star) \wedge \psi(\overline{a}\frac{b}{a_j}), \quad \text{where } \overline{a}\frac{b}{a_j} := a_1 \cdots a_{j-1} b a_{j+1} \cdots a_m.$$

For $t^\star(\overline{x}) := t_{\mathfrak{A}}(\overline{a}\frac{b}{a_j})$ we thus have $\mathfrak{A} \models \exists(t^\star(\overline{x}) \wedge \psi(\overline{x}))$, and the induction hypothesis yields

$$\Theta_k \models \forall\overline{x}(t^\star(\overline{x}) \to \psi(\overline{x})).$$

As above, since $s^\star \subseteq t^\star$, it holds that $\Theta_k \models \forall\overline{x}^\star(s^\star(\overline{x}^\star) \to \exists x_j t^\star(\overline{x}))$, and altogether we obtain

$$\Theta_k \models \forall\overline{x}(s(\overline{x}) \to \exists x_j \psi(\overline{x})).$$

Q.E.D.

Corollary 4.15. For every $L^k_{\infty\omega}$-sentence ψ we either have $\Theta_k \models \psi$ or $\Theta_k \models \neg\psi$.

Corollary 4.16. If $\mathfrak{A} \models \Theta_k$ and $\mathfrak{B} \models \Theta_k$, then $\mathfrak{A} \equiv_{L^k_{\infty\omega}} \mathfrak{B}$.

Corollary 4.17 (Kolaitis, Varidi 1992)**.** For every sentence $\psi \in L^\omega_{\infty\omega}$ (over a relational signature) we have $\mu(\psi) = 0$ or $\mu(\psi) = 1$.

Proof. Let $\psi \in L^k_{\infty\omega}$ for some $k \geq 1$. Then by Corollary 4.15 we have

$\Theta_k \models \psi$ or $\Theta_k \models \neg\psi$. Since $\Theta_k \subseteq T$ is finite, we have $\mu(\Theta_k) = 1$ and thus the claim follows. Q.E.D.

5 Modal, Inflationary and Partial Fixed Points

In finite model theory, a number of other fixed-point logics, in addition to LFP, play an important role. The structure, expressive power, and algorithmic properties of these logics have been studied intensively, and we review these results in this chapter.

5.1 The Modal μ-Calculus

A fragment of LFP that is of fundamental importance in many areas of computer science (e.g. controller synthesis, hardware verification, and knowledge representation) is the modal μ-calculus (L_μ). It is obtained by adding least and greatest fixed points to propositional modal logic (ML). In this way L_μ relates to ML in the same way as LFP relates to FO.

Definition 5.1. The *modal μ-calculus* L_μ extends ML (including propositional variables $X, Y, \ldots$, which can be viewed as monadic second-order variables) by the following rule for building fixed point formulae: If ψ is a formula in L_μ and X is a propositional variable that only occurs positively in ψ, then $\mu X.\psi$ and $\nu X.\psi$ are also L_μ-formulae.

The semantics of these fixed-point formulae is completely analogous to that for LFP. The formula ψ defines on G (with universe V, and with interpretations for other free second-order variables that ψ may have besides X) the monotone operator $F_\psi : \mathcal{P}(V) \to \mathcal{P}(V)$ assigning to every set $X \subseteq V$ the set $\psi^G(X) := \{v \in V : (G, X), v \models \psi\}$. The semantics of fixed-points is defined by

$$G, v \models \mu X.\psi \text{ iff } v \in \mathrm{lfp}(F_\psi)$$
$$G, v \models \nu X.\psi \text{ iff } v \in \mathrm{gfp}(F_\psi).$$

Example 5.2. The formula $\mu X.(\varphi \vee \langle a \rangle X)$ asserts that there exists a path along a-transitions to a node where φ holds.

The formula $\psi := \nu X.\Big((\bigvee_{a \in A} \langle a \rangle \mathit{true}) \wedge (\bigwedge_{a \in A} [a] X)\Big)$ expresses the assertion that the given transition system is deadlock-free. In other words, $G, v \models \psi$ if no path from v in G reaches a dead end (i.e. a node without outgoing transitions).

Finally, the formula $\nu X.\mu Y.\Big(\langle a \rangle \big((\varphi \wedge X) \vee Y\big)\Big)$ says that there exists a path from the current node on which φ holds infinitely often.

The embedding from ML into FO is readily extended to a translation from L_μ into LFP, by inductively replacing formulas of the form $\mu X.\varphi$ by $[\mathrm{lfp}\, Xx.\varphi^*](x)$.

Proposition 5.3. Every formula $\psi \in L_\mu$ is equivalent to a formula $\psi^*(x) \in \mathrm{LFP}$.

Further the argument proving that LFP can be embedded into SO also shows that L_μ is a fragment of MSO.

As for LFP, a fixed μ-calculus formula can be evaluated on a structure $\mathfrak{A}$ in time polynomial in $|\mathfrak{A}|$. The question whether evaluating μ-calculus formulas on a structure when both the formula and the structure are part of the input is in PTIME is a major open problem. On the other hand, it is not difficult to see that the μ-calculus does not suffice to capture PTIME, even in very restricted scenarios such as word structures. Indeed, as L_μ is a fragment of MSO, it can only define *regular languages*, and of course, not all PTIME-languages are regular. However, we shall see in Section 5.5 that there is a multidimensional variant of L_μ that captures the *bisimulation-invariant* fragment of PTIME. Before we do this, let us first show that L_μ is itself invariant under bisimulation. To this end, we translate L_μ formulas into formulas of *infinitary modal logic* $ML_{\infty\omega}$, similar to the embedding of LFP into $L_{\infty\omega}$.

5.1.1 Infinitary Modal Logic and Bisimulation Invariance

Infinitary modal logic extends ML in an analogous way as how infinitary first-order logic extends FO.

Definition 5.4. Let $\kappa \in \mathrm{Cn}^\infty$ be an infinite cardinal number. The *infinitary logic* $ML_{\kappa\omega}$ is inductively defined as follows.

- Predicates P_i are in $ML_{\kappa\omega}$.
- If $\varphi \in ML_{\kappa\omega}$, then also $\neg\varphi$, $\Box\varphi$, $\Diamond\varphi \in ML_{\kappa\omega}$.
- If $\Phi \subseteq ML_{\kappa\omega}$ is a set of formulae with $|\Phi| < \kappa$, then $\bigvee \Phi$, $\bigwedge \Phi \in ML_{\kappa\omega}$.

Further, we write $ML_{\infty\omega}$ to denote $\bigcup_{\kappa \in \mathrm{Cn}^\infty} ML_{\kappa\omega}$.

The semantics of $ML_{\infty\omega}$ on Kripke structures is defined analogously to the semantics of ML, with the following obvious extension for the case of infinite disjuntions and conjunctions.

- $\mathcal{K}, v \models \bigwedge \Phi$ if and only if $\mathcal{K}, v \models \varphi$ for all $\varphi \in \Phi$.
- $\mathcal{K}, v \models \bigvee \Phi$ if and only if there exists a $\varphi \in \Phi$ such that $\mathcal{K}, v \models \varphi$.

The same proof that shows invariance of ML under bisimulation works for $ML_{\infty\omega}$, because the introduction of infinite conjunctions and disjunctions does not interfere with the arguments in the proof at all.

Theorem 5.5. The logic $ML_{\infty\omega}$ is invariant under bisimulation, i.e. if $\varphi \in ML_{\infty\omega}$ is a formula and $\mathcal{K}, v \sim \mathcal{K}', v'$ are two bisimilar Kripke structures, then

$$\mathcal{K}, v \models \varphi \text{ iff } \mathcal{K}', v' \models \varphi.$$

Similarly, the proof of Theorem 5.6 can be adapted to give a translation from L_μ formulas to $ML_{\infty\omega}$, as stated below.

Theorem 5.6. Let $\kappa \in \mathrm{Cn}^\infty$. For each formula $\varphi \in L_\mu$ there exists a formula $\widehat{\varphi} \in ML_{\kappa\omega}$ such that for all transition systems $\mathcal{K}$ with $|\mathcal{K}| < \kappa$ and all $v \in \mathcal{K}$, we have $\mathcal{K}, v \models \varphi$ if and only if $\mathcal{K}, v \models \widehat{\varphi}$.

Combining these two theorems, we get bisimulation invariance of L_μ.

Corollary 5.7. The logic L_μ is invariant under bisimulation.

5.2 Inflationary Fixed-Point Logic

LFP is only one instance of a logic with an explicit operator for forming fixed points. A number of other fixed-point extensions of first-order logic (or fragments of it) have been extensively studied in finite model theory. These include inflationary, partial, non-deterministic, and alternating fixed-point logics. All of these have in common that they allow

the construction of fixed points of operators that are not necessarily monotone.

An operator $G : \mathcal{P}(B) \to \mathcal{P}(B)$ is called *inflationary* if $G(X) \supseteq X$ for all $X \subseteq B$. With any operator F one can associate an inflationary operator G, defined by $G(X) := X \cup F(X)$. In particular, inflationary operators are inductive, so iterating G yields a fixed point, called the *inflationary fixed point* of F.

To be more precise, the inflationary fixed-point of any operator $F\colon \mathcal{P}(B) \to \mathcal{P}(B)$ is defined as the limit of the increasing sequence of sets (R^α) defined as $R^0 := \emptyset$, $R^{\alpha+1} := R^\alpha \cup F(R^\alpha)$, and $R^\lambda := \bigcup_{\alpha<\lambda} R^\alpha$ for limit ordinals λ. The *deflationary fixed point* of F is constructed in the dual way starting with B as the initial stage and taking intersections at successor and limit ordinals.

Remark 5.8.

(1) Monotone operators need not be inflationary, and inflationary operators need not be monotone.

(2) An inflationary operator need not have a least fixed point.

(3) The least fixed point of an inflationary operator (if it exists) may be different from the inductive fixed point.

(4) However, if F is a monotone operator, then its inflationary fixed point and its least fixed point coincide.

The logic IFP is defined with a syntax similar to that of LFP, but without the requirement that the fixed-point variable occurs only positively in the formula defining the operator, and with semantics given by the associated inflationary operator.

Definition 5.9. IFP is the extension of first-order logic by the following fixed-point formation rules. For every formula $\psi(R, \overline{x})$, every tuple $\overline{x}$ of variables, and every tuple $\overline{t}$ of terms (such that the lengths of $\overline{x}$ and $\overline{t}$ match the arity of R), we can build formulas $[\mathrm{ifp}\, R\overline{x}\ .\ \psi](\overline{t})$ and $[\mathrm{dfp}\, R\overline{x}\ .\ \psi](\overline{t})$.

Semantics. For a given structure $\mathfrak{A}$, we have that $\mathfrak{A} \models [\mathrm{ifp}\, R\overline{x}\ .\ \psi](\overline{t})$ and $\mathfrak{A} \models [\mathrm{dfp}\, R\overline{x}\ .\ \psi](\overline{t})$ if $\overline{t}^{\mathfrak{A}}$ is contained in the inflationary and deflationary fixed point of F_ψ, respectively.

By the last item of Remark 5.8, least and inflationary inductions are equivalent for positive formulae, and hence IFP is at least as expressive as LFP. On finite structures, inflationary inductions reach the fixed point after a polynomial number of iterations, hence every IFP-definable class of finite structures is decidable in polynomial time.

Proposition 5.10. IFP captures PTIME on ordered finite structures.

5.2.1 Least Versus Inflationary Fixed-Points

As both logics capture PTIME, IFP and LFP are equivalent on ordered finite structures. What about unordered structures? It was shown by Gurevich and Shelah that the equivalence of IFP and LFP holds on all finite structures. Their proof does not work on infinite structures, and indeed there are some important aspects in which least and inflationary inductions behave differently. For instance, there are first-order operators (on arithmetic, say) whose inflationary fixed point is not definable as the least fixed point of a first-order operator. Further, the alternation hierarchy in LFP is strict, whereas IFP has a positive normal form (see Proposition 5.17 below). Hence it was conjectured by many that IFP might be more powerful than LFP. However, Kreutzer showed recently that IFP is equivalent to LFP on arbitrary structures. Both proofs, by Gurevich and Shelah and by Kreutzer, rely on constructions showing that the *stage comparison relations* of inflationary inductions are definable by lfp inductions.

Definition 5.11. For every inductive operator $F : \mathcal{P}(B) \to \mathcal{P}(B)$, with stages X^α and an inductive fixed point X^∞, the *F-rank* of an element $b \in B$ is $|b|_F := \min\{\alpha : b \in X^\alpha\}$ if $b \in X^\infty$, and $|b|_F = \infty$ otherwise. The *stage comparison relations* of G are defined by

$$\begin{aligned} a \leq_F b \quad &\text{iff} \quad |a|_F \leq |b|_F < \infty \\ a \prec_F b \quad &\text{iff} \quad |a|_F < |b|_F. \end{aligned}$$

Given a formula $\varphi(R, \overline{x})$, we write $\leq_\varphi$ and $\prec_\varphi$ for the stage comparison relations defined by the operator F_φ (assuming that it is indeed inductive), and $\leq_\varphi^{\text{inf}}$ and $\prec_\varphi^{\text{inf}}$ for the stage comparison relations of the associated inflationary operator $G_\varphi : R \mapsto R \cup \{\overline{a} : \mathfrak{A} \models \varphi(R, \overline{a})\}$.

Example 5.12. For the formula $\varphi(T,x,y) := Exy \vee \exists z(Exz \wedge Tzy)$ the relation $\prec_\varphi$ on a graph (V,E) is distance comparison:

$$(a,b) \prec_\varphi (c,d) \text{ iff } \mathrm{dist}(a,b) < \mathrm{dist}(c,d).$$

Stage comparison theorems are results about the definability of stage comparison relations. For instance, Moschovakis proved that the stage comparison relations $\leq_\varphi$ and $\prec_\varphi$ of any positive first-order formula φ are definable by a simultaneous induction over positive first-order formulae. For results on the equivalence of IFP and LFP one needs a stage comparison theorem for IFP inductions.

We first observe that the stage comparison relations for IFP inductions are easily definable in IFP. For any formula $\varphi(T,\bar{x})$ with free variables $\bar{x}$ and free occuring predicate T, the stage comparison relation $\prec_\varphi^{\mathrm{inf}}$ is defined by the formula

$$\psi(\bar{x}'\bar{y}') = [\mathrm{ifp}\,\overline{w} \prec \bar{z}\,.\,\varphi[T\overline{u}/\overline{u} \prec \overline{w}](\overline{w}) \wedge \neg\varphi[T\overline{u}/\overline{u} \prec \bar{z}](\bar{z})](\bar{x}',\bar{y}').$$

Here we syntactically substitute $T,\overline{u}$ by $\overline{u} \prec \overline{w}$ in $\varphi(T\bar{x})$ and, additionally, free variables again by $\overline{w}$. (Note that $\overline{u}$ may contain free variables.) In $\neg\varphi(T,\bar{x})$, we substitute $T,\overline{u}$ by $\overline{u} \prec \bar{z}$ and, additionally, free variables again by $\bar{z}$. Thus free variables become parameter variables of the fixed-point. Now, for the first iteration, T_0 is empty as well as $\prec_0$, so the formula $\varphi(T_0,\overline{w})$ is satisfied by the same $\bar{a}$ as $\varphi(\prec_0,\overline{w})$. So in the first interation, the first components of $\prec_1$ contain the same elements as T_1. The second components of $\prec_1$ contain all other elements. In general, in the i-th iteration, $\prec_i$ consists of pairs $(\bar{a},\bar{b})$ such that $\bar{a} \in T_i$ and $\bar{b} \notin T_i$. In the next step, precisely those $\bar{a}$ satisfy $\varphi[T\overline{u}/\overline{u} \prec \overline{w}](\prec_i)$ that satisfy $\varphi(T_i)$ (instead of $\varphi[T,\overline{u}]$ we now have $\varphi[\overline{u} \prec \overline{w}]$, i.e. $T\bar{a}$ holds if and only if $\overline{u} \prec \bar{a}$ holds if and only if $\bar{a}$ has come to T in the previous steps). So those $\bar{b}$ that do not satisfy $\varphi[T\overline{u}/\overline{u} \prec \overline{w}](\prec_i)$, satisfy $\neg\varphi[T\overline{u}/\overline{u} \prec \overline{w}](\prec_i)$. Summing up, pairs $\bar{a},\bar{b}$ are included to $\prec_{i+1}$ if and only if $\bar{a}$ is included into T_{i+1}, but not earlier, and $\bar{b}$ is not in T_{i+1}.

However, what we need to show is that the stage comparison relation for IFP inductions is in fact LFP-definable.

Theorem 5.13 (Inflationary Stage Comparison)**.** For any formula $\varphi(R,\bar{x})$

in FO or LFP, the stage comparison relation $\prec_\varphi^{\inf}$ is definable in LFP. On finite structures, it is even definable in positive LFP.

From this result, the equivalence of LFP and IFP follows easily.

Theorem 5.14 (Kreutzer)**.** For every IFP-formula, there is an equivalent LFP-formula.

Proof. For any formula $\varphi(R, \overline{x})$,

$$[\text{ifp}\, R\overline{x}\ .\ \varphi](\overline{x}) \equiv \varphi(\{\overline{y} : \overline{y} \prec_\varphi^{\inf} \overline{x}\}, \overline{x}).$$

This holds because, by definition, an inductive fixed-point can only increase. Thus a tuple is added to it if and only if there is a stage, at which the relation R contains all previously added elements (thus $R = \{\overline{y} : \overline{y} \prec_\varphi^{\inf} \overline{x}\}$), and at that stage $\varphi(R, \overline{x})$ holds. Due to Theorem 5.13, the relation $\{\overline{y} : \overline{y} \prec_\varphi^{\inf} \overline{x}\}$ is definable in LFP, so the statement follows directly. Q.E.D.

Positive LFP. While LFP and the modal μ-calculus allow arbitrary nesting of least and greatest fixed points, and arbitrary interleaving of fixed points with Boolean operations and quantifiers, we can also ask about their more restricted forms. Let LFP_1 (sometimes also called positive LFP) be the extension of first-order logic that is obtained by taking least fixed points of positive first-order formulae (without parameters) and closing them under disjunction, conjunction, and existential and universal quantification, but *not* under negation. LFP_1 can be conveniently characterized in terms of simultaneous least fixed points, defined in the next chapter.

Theorem 5.15. A relation is definable in LFP_1 if and only if it is definable by a formula of the form $[\text{lfp}\, R : S](\overline{x})$, where S is a system of update rules $R_i\overline{x} := \varphi_i(\overline{R}, \overline{x})$ with first-order formulae φ_i. Moreover, we can require, without diminishing the expressive power, that each of the formulae φ_i in the system is either a purely existential formula or a purely universal formula.

One interesting consequence of the stage comparison theorems is that on finite structures, greatest fixed points (i.e. negations of least fixed

points) can be expressed in positive LFP. This gives a normal form for LFP and IFP.

Theorem 5.16 (Immerman)**.** On finite structures, every LFP-formula (and hence also every IFP-formula) is equivalent to a formula in LFP_1.

This result fails on infinite structures. On infinite structures, there exist LFP formulae that are not equivalent to positive formulae, and in fact the alternation hierarchy of least and greatest fixed points is strict. This is not the case for IFP.

Proposition 5.17. It can be proven that every IFP-formula is equivalent to one that uses ifp-operators only positively.

Proof. Assume that structures contain at least two elements and that a constant 0 is available. Then a formula $\neg[\mathrm{ifp}\, R\overline{x}\ .\ \psi(R,\overline{x})]$ is equivalent to an inflationary induction on a predicate $T\overline{x}\,y$ which, for $y \neq 0$, simulates the induction defined by ψ, checks whether the fixed point has been reached, and then makes atoms $T\overline{x}0$ true if $\overline{x}$ is not contained in the fixed point. Q.E.D.

In finite model theory, owing to the Gurevich-Shelah Theorem, the two logics LFP and IFP have often been used interchangeably. However, there are significant differences that are sometimes overlooked. Despite the equivalence of IFP and LFP, inflationary inductions are a more powerful concept than monotone inductions. The translation from IFP-formulae to equivalent LFP-formulae can make the formulae much more complicated, requires an increase in the arity of fixed-point variables and, in the case of infinite structures, introduces alternations between least and greatest fixed points. Therefore it is often more convenient to use inflationary inductions in explicit constructions, the advantage being that one is not restricted to inductions over positive formulae. For an example, see the proof of Theorem 5.29 below. Furthermore, IFP is more robust, in the sense that inflationary fixed points remain well defined even when other non-monotone operators (e.g. generalized quantifiers) are added to the language.

5.3 Simultaneous Inductions

A more general variant of LFP permits simultaneous inductions over several formulae. A simultaneous induction is based on a system of operators of the form

$$\begin{aligned} F_1 : \mathcal{P}(B_1) \times \cdots \times \mathcal{P}(B_m) &\to \mathcal{P}(B_1) \\ &\vdots \\ F_m : \mathcal{P}(B_1) \times \cdots \times \mathcal{P}(B_m) &\to \mathcal{P}(B_m), \end{aligned}$$

forming together an operator

$$F = (F_1, \ldots, F_m) : \mathcal{P}(B_1) \times \cdots \times \mathcal{P}(B_m) \to \mathcal{P}(B_1) \times \cdots \times \mathcal{P}(B_m).$$

Inclusion on the product lattice $\mathcal{P}(B_1) \times \cdots \times \mathcal{P}(B_m)$ is componentwise. Accordingly, F is monotone if, whenever $X_i \subseteq Y_i$ for all i, then also $F_i(\overline{X}) \subseteq F_i(\overline{Y})$ for all i.

Everything said above about least and greatest fixed points carries over to simultaneous induction. In particular, a monotone operator F has a least fixed point $\mathrm{lfp}(F)$ which can be constructed inductively, starting with $\overline{X}^0 = (\emptyset, \ldots, \emptyset)$ and iterating F until a fixed point $\overline{X}^\infty$ is reached.

One can extend the logic LFP by a simultaneous fixed point formation rule.

Definition 5.18. *Simultaneous least fixed-point logic*, denoted by S-LFP, is the extension of first-order logic by the following rule.

Syntax. Let $\psi_1(\overline{R}, \overline{x}_1), \ldots, \psi_m(\overline{R}, \overline{x}_m)$ be formulae of vocabulary $\tau \cup \{R_1, \ldots, R_m\}$, with only positive occurrences of $R_1, \ldots, R_m$, and, for each $i \leq m$, let $\overline{x}_i$ be a sequence of variables matching the arity of R_i. Then

$$S := \begin{cases} R_1\overline{x}_1 & := \quad \psi_1 \\ & \vdots \\ R_m\overline{x}_m & := \quad \psi_m \end{cases}$$

is a *system of update rules*, which is used to build formulae $[\mathrm{lfp}\ R_i : S](\bar{t})$ and $[\mathrm{gfp}\ R_i : S](\bar{t})$ (for any tuple $\bar{t}$ of terms whose length matches the arity of R_i).

Semantics. On each structure $\mathfrak{A}$, S defines a monotone operator $S^{\mathfrak{A}} = (S_1, \ldots, S_m)$ mapping tuples $\overline{R} = (R_1, \ldots, R_m)$ of relations on A to $S^{\mathfrak{A}}(\overline{R}) = (S_1(\overline{R}), \ldots, S_m(\overline{R}))$ where $S_i(\overline{R}) := \{\overline{a} : (\mathfrak{A}, \overline{R}) \models \psi_i(\overline{R}, \overline{a})\}$. As the operator is monotone, it has a least fixed point $\mathrm{lfp}(S^{\mathfrak{A}}) = (R_1^\infty, \ldots, R_m^\infty)$. Now $\mathfrak{A} \models [\mathrm{lfp}\, R_i : S](\overline{a})$ if $\overline{a} \in R_i^\infty$. Similarly for greatest fixed points.

As in the case of LFP, one can also extend IFP and PFP (defined in the next section) by simultaneous inductions over several formulae. In all of these cases, simultaneous fixed-point logics S-LFP, S-IFP and S-PFP are not more expressive than their simple variants. This can be proven easily by taking a fixed-point over a relation R with bigger arity, e.g. one higher than the maximum arity of $R_1, \ldots, R_m$. The atoms $R_i(\overline{x})$ can then be replaced by $R(c_i, \overline{x})$ for chosen m constants $c_1, \ldots, c_m$. The fixed-point of R is then sufficient to describe the simultaneous fixed-point of S, yielding the following.

Theorem 5.19. For every formula $\varphi \in$ S-LFP ($\varphi \in$ S-IFP,S-PFP) there exists an equivalent formula $\varphi \in$ LFP ($\varphi \in$ IFP,PFP).

5.4 Partial Fixed-Point Logic

Another fixed-point logic that is relevant to finite structures is the partial fixed-point logic (PFP). Let $\psi(R, \overline{x})$ be an arbitrary formula defining on a finite structure $\mathfrak{A}$ a (not necessarily monotone) operator $F_\psi : R \mapsto \{\overline{a} : \mathfrak{A} \models \psi(R, \overline{a})\}$, and consider the sequence of its finite stages $R^0 := \emptyset$, $R^{m+1} = F_\psi(R^m)$.

This sequence is not necessarily increasing. Nevertheless, as $\mathfrak{A}$ is finite, the sequence either converges to a fixed point, or reaches a cycle with a period greater than one. We define the *partial fixed point* of F_ψ as the fixed point that is reached in the former case, and as the empty relation otherwise. The logic PFP is obtained by adding to first-order logic the *partial-fixed-point formation rule*, which allows us to build from any formula $\psi(R, \overline{x})$ a formula $[\mathrm{pfp}\ R\overline{x}\ .\ \psi(R, \overline{x})](\overline{t})$, saying that $\overline{t}$ is contained in the partial fixed point of the operator F_ψ.

Note that if R occurs only positively in ψ, then

$$[\mathrm{lfp}\ R\overline{x}\ .\ \psi(R, \overline{x})](\overline{t}) \equiv [\mathrm{pfp}\ R\overline{x}\ .\ \psi(R, \overline{x})](\overline{t}),$$

so we have that LFP $\leq$ PFP. However, PFP seems to be much more powerful than LFP. For instance, while a least-fixed-point induction on finite structures always reaches the fixed point in a polynomial number of iterations, a partial-fixed-point induction may need an exponential number of stages.

Example 5.20. Consider the sequence of stages R^m defined by the formula

$$\psi(R,x) := \Big(Rx \wedge \exists y(y < x \wedge \neg Ry)\Big) \vee \Big(\neg Rx \wedge \forall y(y < x \to Ry)\Big) \vee \forall y Ry$$

on a finite linear order $(A, <)$. It is easily seen than the fixed point reached by this induction is the set $R = A$, but before this fixed point is reached, the induction goes in lexicographic order through all possible subsets of A. Hence the fixed point is reached at stage $2^n - 1$, where $n = |A|$.

Complexity. Although a PFP induction on a finite structure may go through exponentially many stages (with respect to the cardinality of the structure), each stage can be represented with polynomial storage space. As first-order formulae can be evaluated efficiently, it follows by a simple induction that PFP-formulae can be evaluated in polynomial space.

Proposition 5.21. For every formula $\psi \in$ PFP, the set of finite models of ψ is in PSPACE; in short: PFP $\subseteq$ PSPACE.

On ordered structures, one can use techniques similar to those used in previous capturing results, to simulate polynomial-space-bounded computation by PFP-formulae.

Theorem 5.22 (Abiteboul, Vianu, and Vardi)**.** On ordered finite structures, PFP captures PSPACE.

Proof. It remains to prove that every class $\mathcal{K}$ of finite ordered structures that is recognizable in PSPACE, can be defined by a PFP-formula.

Let M be a polynomially space-bounded deterministic Turing machine with state set Q and alphabet Σ, recognizing (an encoding of) an ordered structure $(\mathfrak{A}, <)$ if and only if $(\mathfrak{A}, <) \in \mathcal{K}$. Without loss of generality, we can make the following assumptions. For input structures of cardinality n, M requires space less than $n^k - 2$, for some fixed k. For

any configuration C of M, let $\mathrm{Next}(\mathrm{C})$ denote its successor configuration. The transition function of M is adjusted so that $\mathrm{Next}(\mathrm{C}) = \mathrm{C}$ if, and only if, C is an accepting configuration.

We represent any configuration of M with a current state q, tape inscription $w_1 \cdots w_m$, and head position i, by the word $\#w_1 \cdots w_{i-1}(qw_i)w_{i+1} \cdots w_{m-1}\#$ over the alphabet $\Gamma := \Sigma \cup (Q \times \Sigma) \cup \{\#\}$, where $m = n^k$ and # is merely used as an end marker to make the following description more uniform. When moving from one configuration to the next, Turing machines make only local changes. We can therefore associate with M a function $f : \Gamma^3 \to \Gamma$ such that, for any configuration $C = c_0 \cdots c_m$, the successor configuration $\mathrm{Next}(\mathrm{C}) = \mathrm{c}'_0 \cdots \mathrm{c}'_\mathrm{m}$ is determined by the rules

$$c'_0 = c'_m = \# \quad \text{and} \quad c'_i = f(c_{i-1}, c_i, c_{i+1}) \text{ for } 1 \leq i \leq m-1.$$

Recall that we encode structures so that there exist first-order formulae $\beta_\sigma(\overline{y})$ such that $(\mathfrak{A}, <) \models \beta_\sigma(\overline{a})$ if and only the $\overline{a}$th symbol of the input configuration of M for input $\mathrm{code}(()\mathfrak{A}, <)$ is σ. We now represent any configuration C in the computation of M by a tuple $\overline{C} = (C_\sigma)_{\sigma\in\Gamma}$ of k-ary relations, where

$$C_\sigma := \{\overline{a} : \text{ the } \overline{a}\text{-th symbol of } C \text{ is } \sigma\}.$$

The configuration at time t is the stage $t+1$ of a simultaneous pfp induction on $(\mathfrak{A}, <)$, defined by the rules

$$C_\#\overline{y} := \forall z(\overline{y} \leq \overline{z}) \vee \forall \overline{z}(\overline{z} \leq \overline{y})$$

and, for all $\sigma \in \Gamma - \{\#\}$,

$$C_\sigma\overline{y} := \Big(\beta_\sigma(\overline{y}) \wedge \bigwedge_{\gamma\in\Gamma} \forall\overline{x}\neg C_\gamma\overline{x}\Big) \vee \exists\overline{x}\exists\overline{z}\Big(\overline{x}+1 = \overline{y} \wedge \overline{y}+1 = \overline{z} \wedge \bigvee_{f(\alpha,\beta,\gamma)=\sigma} C_\alpha\overline{x} \wedge C_\beta\overline{y} \wedge C_\gamma\overline{z})\Big)$$

The first rule just says that each stage represents a word starting and ending with #. The other rules ensure that (1) if the given sequence

$\overline{C}$ contains only empty relations (i.e. if we are at stage 0), then the next stage represents the input configuration, and (2) if the given sequence represents a configuration, then the following stage represents its successor configuration.

By our convention, M accepts its input if and only the sequence of configurations becomes stationary (i.e. reaches a fixed point). Hence M accepts code$(()\mathfrak{A}, <)$ if and only if the relations defined by the simultaneous pfp induction on $\mathfrak{A}$ of the rules described above are non-empty. Hence $\mathcal{K}$ is PFP-definable. Q.E.D.

5.4.1 Least Versus Partial Fixed-Point Logic

From the capturing results for PTIME and PSPACE we immediately obtain the result that PTIME = PSPACE if, and only if, LFP = PFP on ordered finite structures. The natural question arises of whether LFP and PFP can be separated on the domain of all finite structures. For a number of logics, separation results on arbitrary finite structures can be established by relatively simple methods, even if the corresponding separation on ordered structures would solve a major open problem in complexity theory. For instance, we have proved by quite a simple argument that DTC $\subsetneq$ TC, and it is also not very difficult to show that TC $\subsetneq$ LFP (indeed, TC is contained in stratified Datalog, which is also strictly contained in LFP). Further, it is trivial that LFP is less expressive than Σ^1_1 on all finite structures. However the situation is different for LFP vs. PFP.

Theorem 5.23 (Abiteboul and Vianu)**.** LFP and PFP are equivalent on finite structures if, and only if, PTIME = PSPACE.

5.5 Capturing PTIME up to Bisimulation

In mathematics, we consider isomorphic structures as identical. Indeed, it almost goes without saying that relevant mathematical notions do not distinguish between isomorphic objects. As classical algorithmic devices work on ordered *representations of structures* rather than the structures themselves, our capturing results rely on an ability to reason

about canonical ordered representations of isomorphism classes of finite structures.

However, in many application domains of logic, structures are distinguished only up to equivalences coarser than isomorphism. Perhaps the best-known example is the modelling of the computational behaviour of (concurrent) programs by transition systems. The meaning of a program is usually not captured by a unique transition system. Rather, transition systems are distinguished only up to appropriate notions of behavioural equivalence, the most important of these being *bisimulation*.

In such a context, the idea of a logic capturing PTIME gets a new twist. One would like to express in a logic precisely those properties of structures that are

(1) decidable in polynomial time, and

(2) invariant under the notion of equivalence being studied.

A class S of rooted transition systems or Kripke structures is *invariant under bisimulation* if, whenever $\mathcal{K}, v \in S$ and $\mathcal{K}, v \sim \mathcal{K}', v'$, then also $\mathcal{K}', v' \in S$. We say that a class S of finite rooted transition systems is in *bisimulation-invariant PTIME* if it is invariant under bisimulation, and if there exists a polynomial-time algorithm deciding whether a given pair $\mathcal{K}, v$ belongs to S. A logic L is invariant under bisimulation if all L-definable properties of rooted transition systems are.

Clearly, $L_\mu \subseteq$ bisimulation-invariant PTIME. However, as pointed out in Section 5.1, L_μ is far too weak to *capture* this class, mainly because it is essentially a monadic logic. Instead, we have to consider a *multidimensional* variant L_μ^ω of L_μ.

But before we define this logic, we should explain the main technical step, which relies on definable canonization, but of course with respect to bisimulation rather than isomorphism. For simplicity of notation, we consider only Kripke structures with a single transition relation E. The extension to the case of several transition relations E_a is straightforward.

With a rooted Kripke structure $\mathcal{K} = (V, E, (P_b)_{b \in B}), u$, we associate a new transition system

$$\mathcal{K}_u^\sim := (V_u^\sim, E^\sim, (P_b^\sim)_{b \in B}),$$

where $V_u^\sim$ is the set of all $\sim$-equivalence classes $[v]$ of nodes $v \in V$ that are reachable from u. More formally, let $[v]$ denote the bisimulation equivalence class of a node $v \in V$. Then

$$V_u^\sim := \{[v] : \text{there is a path in } G \text{ from } u \text{ to } v\}$$
$$P_b^\sim := \{[v] \in V_u^\sim : v \in P_b\}$$
$$E^\sim := \{([v],[w]) : (v,w) \in E\}.$$

The pair $\mathcal{K}_u^\sim, [u]$ is, up to isomorphism, a *canonical representant* of the bisimulation equivalence class of $\mathcal{K}, u$. To see this one can prove that (1) $(\mathcal{K},u) \sim (\mathcal{K}_u^\sim,[u])$, and (2) if $(\mathcal{K},u) \sim (\mathcal{G},v)$, then $(\mathcal{K}_u^\sim,[u]) \cong (\mathcal{G}_v^\sim,[v])$.

It follows that a class S of rooted transition systems is bisimulation-invariant if and only if $S = \{(\mathcal{K},u) : (\mathcal{K}_u^\sim,[u]) \in S\}$. Let $\mathcal{CR}^\sim$ be the domain of canonical representants of finite transition systems, i.e.

$$\mathcal{CR}^\sim := \{\mathcal{K},u \mid (\mathcal{K}_u^\sim,[u]) \cong (\mathcal{K},u)\}.$$

Proposition 5.24. $\mathcal{CR}^\sim$ admits LFP-definable linear orderings, i.e. for every vocabulary $\tau = \{E\} \cup \{P_b : b \in B\}$, there exists a formula $\psi(x,y) \in \mathrm{LFP}(\tau)$ which defines a linear order on every transition system in $\mathcal{CR}^\sim(\tau)$.

Proof. Recall that bisimulation equivalence on a transition system is a greatest fixed point. Its complement, bisimulation inequivalence, is a least fixed point, which is the limit of an increasing sequence $\not\sim_i$ defined as follows: $u \not\sim_0 v$ if u and v do not have the same atomic type, i.e. if there exists some b such that one of the nodes u,v has the property P_b and the other does not. Further, $u \not\sim_{i+1} v$ if the sets of $\sim_i$-classes that are reachable in one step from u and v are different. The idea is to refine this inductive process, by defining relations $\prec_i$ that order the $\sim_i$-classes. On the transition system itself, these relations are pre-orders. The inductive limit $\prec$ of the pre-orders $\prec_i$ defines a linear order of the bisimulation equivalence classes. But in transition systems in $\mathcal{CR}^\sim$, bisimulation classes have only one element, so $\prec$ actually defines a linear order on the set of nodes.

To make this precise, we choose an order on B and define $\prec_0$ by

enumerating the $2^{|B|}$ atomic types with respect to the propositions P_b, i.e.

$$x \prec_0 y := \bigvee_{b \in B} \Big(\neg P_b x \wedge P_b y \wedge \bigwedge_{b' < b} P_{b'} x \leftrightarrow P_{b'} y \Big).$$

In other words, there is some b such that P_b separates x from y and for the least such b, P_b holds on y and not on x.

In what follows, $x \sim_i y$ can formally be taken as an abbreviation for $\neg(x \prec_i y \vee y \prec_i x)$, and similarly for $x \sim y$. We define $x \prec_{i+1} y$ by the condition that either $x \prec_i y$, or $x \sim_i y$ and the set of $\sim_i$-classes reachable from x is lexicographically smaller than the set of $\sim_i$-classes reachable from y. Note that this inductive definition of $\prec$ is not monotone, so it cannot be directly captured by an LFP-formula. However, as we know that LFP $\equiv$ IFP, we can use an IFP-formula instead. Explicitly, $\prec$ is defined by $[\mathrm{ifp}\, x \prec y \,.\, \psi(\prec, x, y)](x, y)$, where

$$\begin{aligned} \psi(\prec, x, y) := x \prec_0 y \vee \Big(& x \sim y \wedge \\ & (\exists y' \,.\, Eyy') \Big((\forall x' \,.\, Exx') x' \not\sim y' \wedge \\ & \qquad (\forall z . z \prec y') \big(\exists x'' (Exx'' \wedge x'' \sim z) \leftrightarrow \\ & \qquad\qquad \exists y'' (Eyy'' \wedge y'' \sim z) \big) \Big) \Big). \end{aligned}$$

Q.E.D.

Corollary 5.25. On the domain $\mathcal{CR}^{\sim}$, LFP captures PTIME.

Since LFP is not invariant under bisimulation, we will strengthen the above result and capture bisimulation-invariant PTIME in terms of a natural logic, the multidimensional μ-calculus L_μ^ω.

Definition 5.26. The syntax of the *k-dimensional μ-calculus* L_μ^k (for transition systems $\mathcal{K} = (V, E, (P_b)_{b \in B})$) is the same as the syntax of the usual μ-calculus L_μ with modal operators $\langle i \rangle$, $[i]$, and $\langle \sigma \rangle$, $[\sigma]$ for every substitution $\sigma : \{1, \ldots, k\} \to \{1, \ldots, k\}$. Let $S(k)$ be the set of all these substitutions.

The semantics is different, however. A formula ψ of L_μ^k is interpreted on a transition system $\mathcal{K} = (V, E, (P_b)_{b\in B})$ at node v by evaluating it as a formula of L_μ on the modified transition system

$$\mathcal{K}^k = (V^k, (E_i)_{1\le i\le k}, (E_\sigma)_{\sigma\in S(k)}, (P_{b,i})_{b\in B, 1\le i\le k})$$

at node $\underline{v} := (v, v, \dots, v)$. Here $V^k = V \times \cdots \times V$ and

$$\begin{aligned} E_i &:= \{(\overline{v}, \overline{w}) \in V^k \times V^k : (v_i, w_i) \in E \text{ and } v_j = w_j \text{ for } j \neq i\} \\ E_\sigma &:= \{(\overline{v}, \overline{w}) \in V^k \times V^k : w_i = v_{\sigma(i)} \text{ for all } i\} \\ P_{b,i} &:= \{\overline{v} \in V^k : v_i \in P_b\} \end{aligned}$$

That is, $\mathcal{K}, v \models_{L_\mu^k} \psi$ iff $\mathcal{K}^k, (v, \dots, v) \models_{L_\mu} \psi$. The *multidimensional μ-calculus* is $L_\mu^\omega = \bigcup_{k<\omega} L_\mu^k$.

Remark. Instead of evaluating a formula $\psi \in L_\mu^k$ at single nodes v of G, we can also evaluate it at k-tuples of nodes: $\mathcal{K}, \overline{v} \models_{L_\mu^k} \psi$ iff $\mathcal{K}^k, \overline{v} \models_{L_\mu} \psi$.

Example 5.27. Bisimulation is definable in L_μ^2 (in the sense of the remark just made). Let

$$\psi^\sim := \nu X \,.\, \Big(\bigwedge_{b\in B} (P_{b,1} \leftrightarrow P_{b,2}) \wedge [1]\langle 2\rangle X \wedge [2]\langle 1\rangle X \Big).$$

For every transition system $\mathcal{K}$, we have that $\mathcal{K}, v_1, v_2 \models \psi^\sim$ if, and only if, v_1 and v_2 are bisimilar in $\mathcal{K}$. Further, we have that

$$\mathcal{K}, v \models \mu Y \,.\, \langle 2\rangle(\psi^\sim \vee \langle 2\rangle Y)$$

if, and only if, there exists in $\mathcal{K}$ a point w that is reachable from v (by a path of length ≥ 1) and bisimilar to v.

One can see that L_μ^ω is invariant under bisimulation (because if $\mathcal{K}, v_i \sim \mathcal{G}, u_i$ for all i then also $\mathcal{K}^k, \overline{v} \sim \mathcal{G}, \overline{u}$) and that L_μ^ω can be embedded in LFP. This establishes the easy direction of the desired result: $L_\mu^\omega \subseteq$ bisimulation-invariant PTIME.

For the converse, it suffices to show that LFP and L_μ^ω are equivalent on the domain $\mathcal{CR}^\sim$. Let S be a class of rooted transition systems in

bisimulation-invariant PTIME. For any $\mathcal{K}, u$, we have that $\mathcal{K}, u \in S$ if its canonical representant $\mathcal{K}_u^\sim, [u] \in S$. If LFP and L_μ^ω are equivalent on $\mathcal{CR}^\sim$, then there exists a formula $\psi \in L_\mu^\omega$ such that $\mathcal{K}_u^\sim, [u] \models \psi$ iff $\mathcal{K}_u^\sim, [u] \in S$. By the bisimulation invariance of ψ, it follows that $\mathcal{K}, u \models \psi$ iff $\mathcal{K}, u \in S$.

The *width* of an LFP-formula φ is the maximal number of free variables occuring in a subformula of φ.

Proposition 5.28. On the domain $\mathcal{CR}^\sim$, LFP $\leq L_\mu^\omega$. More precisely, for each formula $\psi(x_1, \ldots, x_k) \in$ LFP of width $\leq k$, there exists a formula $\psi^* \in L_\mu^{k+1}$ such that for each $\mathcal{K}, u \in \mathcal{CR}^\sim$, we have that $\mathcal{K} \models \psi(u, \overline{v})$ iff $\mathcal{K}, u, \overline{v} \models \psi^*$.

Note that although, ultimately, we are interested only in formulae $\psi(x)$ with just one free variable, we need more general formulae, and evaluation of L_μ^k-formulae over k-tuples of nodes, for the inductive treatment. In all formulae, we shall have at least x_1 as a free variable, and we always interpret x_1 as u (the root of the transition system). We remark that, by an obvious modification of the formula given in Example 5.27, we can express in L_μ^k the assertion that $x_i \sim x_j$ for any i, j.

Atomic formulae are translated from LFP to L_μ^ω according to

$$\begin{aligned}(x_i = x_j)^* &:= x_i \sim x_j\\ (P_b x_i)^* &:= P_{b,i}\overline{x}\\ (Ex_ix_j)^* &:= \langle i\rangle x_i \sim x_j\\ (Xx_{\sigma(1)} \cdots x_{\sigma(r)})^* &:= \langle\sigma\rangle X.\end{aligned}$$

Boolean connectives are treated in the obvious way, and *quantifiers* are translated by use of fixed points. To find a witness x_j satisfying a formula ψ, we start at u (i.e. set $x_j = x_1$), and search along transitions (i.e. use the μ-expression for reachability). That is, let $j/1$ be the substitution that maps j to 1 and fixes the other indices, and translate $\exists x_j \psi(\overline{x})$ into

$$\langle j/1\rangle \mu Y \,.\, \psi^* \vee \langle j\rangle Y.$$

Finally, *fixed points* are first brought into normal form so that variables appear in the right order, and then they are translated literally, i.e.

$[\mathrm{lfp}\, X\overline{x} \,.\, \psi](\overline{x})$ translates into $\mu X \,.\, \psi^*$.

The proof that the translation has the desired property is a straightforward induction, which we leave as an exercise. Altogether we have established the following result.

Theorem 5.29 (Otto)**.** The multidimensional μ-calculus captures bisimulation-invariant PTIME.

Otto has also established capturing results with respect to other equivalences. For finite structures $\mathfrak{A}, \mathfrak{B}$, we say that $\mathfrak{A} \equiv_k \mathfrak{B}$ if no first-order sentence of width k can distinguish between $\mathfrak{A}$ and $\mathfrak{B}$. Similarly, $\mathfrak{A} \equiv_k^C \mathfrak{B}$ if $\mathfrak{A}$ and $\mathfrak{B}$ are indistinguishable by first-order sentences of width k with counting quantifiers of the form $\exists^{\geq i} x$, for any $i \in \mathbb{N}$.

Theorem 5.30 (Otto)**.** There exist logics that effectively capture $\equiv_2$-invariant PTIME and $\equiv_2^C$-invariant PTIME on the class of all finite structures.