Kapitel 4: Fixpunktlogiken

~~5.4 2.2 Deskriptive Komplexität~~

4.1 ~~2.2.1~~ Etwas Fixpunkttheorie

Wir schreiben $\text{Pot}(A)$, um die Potenzmenge einer Menge A zu bezeichnen, d.h.

$$\text{Pot}(A) = \{X : X \subseteq A\}.$$

Insbesondere gilt natürlich für jedes *endliche* A, dass $|\text{Pot}(A)| = 2^{|A|}$.

4.1 ~~2.13~~ Definition. Sei A eine Menge und $F : \text{Pot}(A) \to \text{Pot}(A)$ eine Abbildung.

(a) F heißt *monoton*, falls für alle Mengen $P, Q \in \text{Pot}(A)$ gilt:

$$P \subseteq Q \implies F(P) \subseteq F(Q).$$

(b) F heißt *inflationär*, falls für alle $P \in \text{Pot}(A)$ gilt: $P \subseteq F(P)$.

(c) Eine Menge $P \in \text{Pot}(A)$ heißt *Fixpunkt* von F, falls $F(P) = P$.

(d) Eine Menge $P \in \text{Pot}(A)$ heißt *kleinster Fixpunkt* von F, falls P ein Fixpunkt von F ist und für jeden Fixpunkt Q von F gilt: $P \subseteq Q$.

4.2 ~~2.14~~ Proposition.

(a) Es gibt Abbildungen, die weder monoton noch inflationär sind.

(b) Es gibt Abbildungen, die monoton, aber nicht inflationär sind.

(c) Es gibt Abbildungen, die inflationär, aber nicht monoton sind.

(d) Es gibt Abbildungen, die keinen Fixpunkt besitzen.

(e) Für jede Abbildung $F : \text{Pot}(A) \to \text{Pot}(A)$ gilt:
Falls es einen kleinsten Fixpunkt von F gibt, so ist dieser eindeutig bestimmt.

Beweis: *Übung.* □

Der folgende Satz charakterisiert die kleinsten Fixpunkte *monotoner* Abbildungen:

4.3 ~~2.15~~ Satz (Knaster und Tarski).
Sei A eine Menge und $F : \text{Pot}(A) \to \text{Pot}(A)$ eine monotone Abbildung.
Dann hat F einen kleinsten Fixpunkt, der $\textbf{lfp}(F)$ genannt wird, und es gilt:[3]

$$\textbf{lfp}(F) = \bigcap\{X \subseteq A : F(X) = X\} = \bigcap\{X \subseteq A : F(X) \subseteq X\}.$$

[3] Für eine Menge M, deren Elemente selbst Mengen sind, schreiben wir $\bigcap M$, um die Schnittmenge $\bigcap_{X \in M} X$ zu bezeichnen.

Beweis: Sei

$$M := \{X \subseteq A : F(X) \subseteq X\} \qquad \text{und} \qquad P := \bigcap M.$$

Schritt 1: *Wir zeigen zunächst, dass P ein Fixpunkt von F ist:*

"$F(P) \subseteq P$": Da $P = \bigcap M$, gilt für jedes $X \in M$, dass $P \subseteq X$. Da F monoton ist, folgt $F(P) \subseteq F(X)$. Wegen $X \in M$ gilt $F(X) \subseteq X$. Somit gilt für alle $X \in M$, dass $F(P) \subseteq X$. Daher gilt also:

$$F(P) \subseteq \bigcap M \stackrel{\text{def}}{=} P.$$

"$P \subseteq F(P)$": Da (wie wir gerade gezeigt haben) $F(P) \subseteq P$ ist, folgt aus der Monotonie von F, dass auch $F\big(F(P)\big) \subseteq F(P)$. Somit gilt $F(P) \in M$. Wegen $P = \bigcap M$ folgt daher $P \subseteq F(P)$.

Schritt 2: *Für jeden Fixpunkt Q von F gilt $P \subseteq Q$, denn:*

Für jeden Fixpunkt Q von F gilt $F(Q) \subseteq Q$. Daher ist $Q \in M$, und gemäß Definition von P daher $P \subseteq Q$.

Schritt 3: *Abschluss des Beweises:*

Aus den Schritten 1 und 2 folgt, dass P der kleinste Fixpunkt von F ist.
Außerdem gilt für die Menge

$$M' := \{X \subseteq A : F(X) = X\},$$

dass $P \in M'$ (da $F(P) = P$) und $M' \subseteq M$. Somit folgt

$$P \underset{\text{(Def.)}}{=} \bigcap M \underset{(M \supseteq M')}{\subseteq} \bigcap M' \underset{(P \in M')}{\subseteq} P.$$

Daher gilt für $\mathbf{lfp}(F) := P$, dass

$$\mathbf{lfp}(F) = \bigcap \{X \subseteq A : F(X) = X\} = \bigcap \{X \subseteq A : F(X) \subseteq X\}$$

der kleinste Fixpunkt von F ist. □

Eine andere Charakterisierung der kleinsten Fixpunkte monotoner Funktionen, die auch gleich ein Verfahren zum *Ausrechnen* des kleinsten Fixpunkts liefert, ergibt sich durch die Betrachtung der einzelnen *Induktionsstufen*:

44
~~2.16~~ Definition (Induktionsstufen).
Sei A eine *endliche* Menge und $F : \text{Pot}(A) \to \text{Pot}(A)$ eine Abbildung.

138

(a) Wir definieren induktiv eine Sequenz von Mengen $R^0, R^1, R^2, \ldots$ durch

$$R^0 := \varnothing \qquad \text{und} \qquad R^{i+1} := F\big(R^i\big) \qquad \text{(für alle } i \in \mathbb{N}\text{)}.$$

Die Menge R^i heißt *i-te Induktionsstufe* (oder auch: i-te Stufe).

(Es gilt also für alle $i \in \mathbb{N}$: $R^i = F^i(\varnothing)$.)

(b) F heißt *induktiv*, falls für alle $i \in \mathbb{N}$ gilt: $R^i \subseteq R^{i+1}$.

4.5 ~~2.17~~ **Proposition.**

(a) Jede monotone oder inflationäre Abbildung ist induktiv.

(b) Es gibt Abbildungen, die induktiv, aber weder monoton noch inflationär sind.

Beweis: *Übung.* □

4.6 ~~2.18~~ **Proposition.** *Für jede endliche Menge A und jede induktive Abbildung $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$ wird die Sequenz der Induktionsstufen stationär, d.h. es gibt eine Zahl $i \leqslant |A|$, so dass $R^i = R^{i+1} \stackrel{\text{def}}{=} F(R^i)$, und somit $R^i = R^{i+n}$ für alle $n \geqslant 0$.*

Beweis: Da F induktiv ist, gilt

$$\varnothing = R^0 \subseteq R^1 \subseteq R^2 \subseteq \cdots \subseteq R^j \subseteq R^{j+1} \subseteq \cdots \subseteq A.$$

Da A nur $|A|$ viele Elemente besitzt, kann *nicht* für alle $j \in \{0, 1, .., |A|\}$ gelten, dass $R^j \subsetneq R^{j+1}$ (denn sonst würde in jeder der Stufen $1, 2, .., |A|, |A|+1$ mindestens ein neues Element hinzukommen, in A gibt es aber nur $|A|$ viele Elemente).
Somit muss es also ein $i \in \{0, 1, .., |A|\}$ geben, so dass $R^i = R^{i+1}$. Gemäß der Definition der Stufen ($R^{j+1} := F(R^j)$) folgt daraus, dass

$$R^i = F(R^i) = F(F(R^i)) = F(F(F(R^i))) = \cdots,$$

also

$$R^i = R^{i+1} = R^{i+2} = R^{i+3} = \cdots.$$

□

4.7 ~~2.19~~ **Definition.** Sei A eine endliche Menge und $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$ induktiv.

(a) Die kleinste Zahl i, für die $R^i = R^{i+1}$ (d.h. $F^i(\varnothing) = F^{i+1}(\varnothing) = F^{i+n}(\varnothing)$, für alle $n \geqslant 0$), heißt das *Abschlussordinal* von F, oder kurz: $\mathrm{cl}(F)$.

(b) Die Menge $R^\infty := R^{\mathrm{cl}(F)}$ heißt *induktiver Fixpunkt* von F.

4.8
~~2.20~~ Satz. *Sei A eine endliche Menge.*
Für jede monotone Abbildung $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$ gilt:

$$\mathbf{lfp}(F) \;=\; R^{\infty}.$$

D.h.: Der induktive Fixpunkt einer monotonen Abbildung ist gerade der kleinste Fixpunkt.

Beweis: "$\subseteq$": Gemäß der Definition des inflationären Fixpunkts gilt $R^{\infty} = F(R^{\infty})$, R^{∞} ist also ein Fixpunkt von F. Insbesondere gilt

$$\mathbf{lfp}(F) \subseteq R^{\infty},$$

da der kleinste Fixpunkt gemäß Definition in jedem anderen Fixpunkt enthalten ist.
"$\supseteq$": Per Induktion zeigen wir, dass $R^i \subseteq \mathbf{lfp}(F)$ für alle $i \in \mathbb{N}$.
$i = 0$: Klar, da $R^0 = \varnothing \subseteq \mathbf{lfp}(F)$.
$i \to i{+}1$: Per Induktion gilt $R^i \subseteq \mathbf{lfp}(F)$. Da F monoton ist, folgt, dass

$$R^{i+1} \;\stackrel{\text{def}}{=}\; F(R^i) \;\subseteq\; F(\mathbf{lfp}(F)) \;=\; \mathbf{lfp}(F).$$

□

Den kleinsten Fixpunkt einer monotonen Funktion kann man leicht berechnen, indem man nach und nach die einzelnen Induktionsstufen berechnet und abbricht, sobald diese stationär werden:

4.9
~~2.21~~ Proposition. *Ist $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$ monoton und in polynomieller Zeit berechenbar, so kann $\mathbf{lfp}(F)$ in polynomieller Zeit berechnet werden.*

Beweis: Es gibt höchstens $|A|$ Induktionsstufen, und jede davon kann in polynomieller Zeit aus der vorangegangenen berechnet werden. □

4.2
~~2.2.2~~ Die kleinste Fixpunktlogik

4.10
~~2.22~~ Definition (Operator $F_{\varphi,\mathfrak{A}}$).
Sei σ eine Signatur, $k \in \mathbb{N}_{\geqslant 1}$, R eine k-stellige Relationsvariable und $\vec{x} = x_1, \ldots, x_k$ ein Tupel aus k verschiedenen Variablen erster Stufe.
Jede $\mathrm{FO}[\sigma\dot{\cup}\{R\}]$-Formel $\varphi(R, \vec{x})$ definert in jeder σ-Struktur $\mathfrak{A}$ eine Abbildung $F_{\varphi,\mathfrak{A}}$ wie folgt:

$$\begin{array}{rrcl} F_{\varphi,\mathfrak{A}}: & \mathrm{Pot}(A^k) & \to & \mathrm{Pot}(A^k) \\ & P & \mapsto & \{\vec{a} \in A^k : \mathfrak{A} \models \varphi[P, \vec{a}]\}. \end{array}$$

Zum Beispiel kann man sich φ als Datenbankanfrage vorstellen und $F_{\varphi,\mathfrak{A}}$ als die Abbildung, die jeder Datenbank-Relation P das Ergebnis der Anfrage zuordnet.

Die im Folgenden definierte *monotone Fixpunktlogik* ist eine Erweiterung der Logik erster

Stufe durch einen Operator, mit dem man aus einer Formel $\varphi(R, \vec{x})$ eine neue Formel erhalten kann, die $[\mathrm{lfp}_{R,\vec{x}}\,\varphi](\vec{t})$ genannt wird, und die in jeder Struktur $\mathfrak{A}$ genau von den Tupeln $\vec{t} \in A^k$ erfüllt wird, die zu dem kleinsten Fixpunkt der Operation $F_{\varphi,\mathfrak{A}}$ gehören. Dafür sollte man natürlich wissen, ob der kleinste Fixpunkt auch wirklich existiert. Gemäß dem Satz von Knaster und Tarski (Satz 2.15) existiert der kleinste Fixpunkt von $F_{\varphi,\mathfrak{A}}$ auf jeden Fall dann, wenn $F_{\varphi,\mathfrak{A}}$ *monoton* ist.

4.11 ~~2.23~~ **Definition** (Monotone Fixpunktlogik MFP)**.** Sei σ eine Signatur.
Die Formelmenge MFP$[\sigma]$ ist induktiv durch die Regeln (A1), (A2), (A3), (BC) und (Q1) der Logik erster Stufe, sowie die folgende Regel (MFP) definiert:

(MFP) Ist $\varphi(R, \vec{x})$ eine MFP$[\sigma]$-Formel, wobei

- R eine k-stellige Relationsvariable, für ein $k \in \mathbb{N}_{\geqslant 1}$,
- $\vec{x} = x_1, \ldots, x_k$ ein Tupel aus k verschiedenen Variablen erster Stufe, und
- φ außer R und $\vec{x}$ evtl. noch andere freie Variablen hat (etwa $\vec{u}, \vec{S}$),

ist $\vec{t} = t_1, \ldots, t_k$ ein k-Tupel aus Variablen erster Stufe und/oder Konstantensymbolen aus σ, und ist ***$F_{\varphi,\mathfrak{A}}$* monoton auf jeder endlichen** $(\sigma\dot\cup\{\vec{u}, \vec{S}\})$***-Struktur*** $\mathfrak{A}$, so ist

$$[\mathrm{lfp}_{R,\vec{x}}\,\varphi](\vec{t})$$

eine MFP$[\sigma]$-Formel.

4.12 ~~2.24~~ **Bemerkungen.**

(a) Die Menge der *freien Variablen* einer MFP$[\sigma]$-Formel ist induktiv definiert wie für FO bzw. SO, mit der zusätzlichen Regel

$$\mathit{frei}\big(\, [\mathrm{lfp}_{R,\vec{x}}\,\varphi](\vec{t}\,)\,\big) \quad := \quad \big(\mathit{frei}(\varphi) \setminus \{R, \vec{x}\}\big) \cup \{t_i \;:\; t_i \in \mathrm{Var}_1\}.$$

(b) Gilt $\mathit{frei}(\varphi) = \{R, \vec{x}, \vec{u}, \vec{S}\}$, für ein Tupel $\vec{u}$ von Variablen erster Stufe und ein Tupel $\vec{S}$ von Relationsvariablen, so nennen wir die Variablen in $\vec{u}$ die *Parameter* der Formel $[\mathrm{lfp}_{R,\vec{x}}(\varphi)](\vec{t})$.

4.13 ~~2.25~~ **Definition** (Semantik von MFP$[\sigma]$-Formeln)**.** Die Semantik von MFP$[\sigma]$-Formeln der Form $\psi := [\mathrm{lfp}_{R,\vec{x}}\,\varphi](\vec{t})$ ist folgendermaßen definiert:
Ist $\mathit{frei}(\psi) = \{\vec{u}, \vec{S}\}$ und ist $\mathfrak{A}$ eine endliche $(\sigma\dot\cup\{\vec{u}, \vec{S}\})$-Struktur, so gilt:

$$\mathfrak{A} \models [\mathrm{lfp}_{R,\vec{x}}\,\varphi](\vec{t}) \quad :\Longleftrightarrow \quad \vec{t}^{\,\mathfrak{A}} \in \mathrm{lfp}(F_{\varphi,\mathfrak{A}}).$$

Aus dem Satz von Trakhtenbrot folgt leicht, ~~Aus Satz 1.68 ist bekannt~~, dass Monotonie von FO-Formeln auf endlichen Strukturen unentscheidbar ist. Es gibt also keinen Algorithmus, der bei Eingabe einer Formel $\varphi(R, \vec{x})$ entscheidet, ob $F_{\varphi,\mathfrak{A}}$ für alle endlichen Strukturen $\mathfrak{A}$ monoton ist.

141

Daher ist die Syntax von MFP unentscheidbar, d.h. es gibt keinen Algorithmus, der bei Eingabe einer Formel ψ entscheidet, ob diese zu MFP gehört.

Wünschenswert wäre, eine Fixpunktlogik zu definieren, bei der man schon an der Syntax einer Formel erkennt, ob diese zur Logik gehört oder nicht. Ein rein *syntaktisches* Kriterium, das hinreichend für Monotonie ist, wird durch die folgende Definition gegeben:

4.14 ~~2.26~~ **Definition.** Eine Formel $\varphi(R, \vec{x})$ heißt *positiv in* R (bzw. *negativ in* R), falls Atome der Form $R(\vec{y})$ in φ stets im Bereich einer *geraden* (bzw. *ungeraden*) Anzahl von Negationssymbolen vorkommen und in φ kein Implikationspfeil "$\rightarrow$" und kein Biimplikationspfeil "$\leftrightarrow$" vorkommt.

Man sieht leicht:

4.15 ~~2.27~~ **Proposition.** *Für jede Formel* $\varphi(R, \vec{x})$, *die positiv in* R *ist, gilt:* $F_{\varphi,\mathfrak{A}}$ *ist* monoton *für alle Strukturen* $\mathfrak{A}$.

Beweisidee: Per Induktion nach dem Formelaufbau zeigt man, dass für jede Formel

$$\varphi(\vec{x}, R_1, .., R_k, S_1, .., S_\ell),$$

die *positiv* in $R_1, .., R_k$ und *negativ* in $S_1, .., S_\ell$ ist, folgendes für alle Strukturen $\mathfrak{A}$, alle Tupel $\vec{a}$ und alle Relationen $R_1^{\mathfrak{A}} \subseteq R_1'^{\mathfrak{A}}, \ldots, R_k^{\mathfrak{A}} \subseteq R_k'^{\mathfrak{A}}$ und $S_1^{\mathfrak{A}} \supseteq S_1'^{\mathfrak{A}}, \ldots, S_\ell^{\mathfrak{A}} \supseteq S_\ell'^{\mathfrak{A}}$ über dem Universum von A gilt: Falls $\mathfrak{A} \models \varphi[\vec{a}, \vec{R}^{\mathfrak{A}}, \vec{S}^{\mathfrak{A}}]$, so auch $\mathfrak{A} \models \varphi[\vec{a}, \vec{R}'^{\mathfrak{A}}, \vec{S}'^{\mathfrak{A}}]$.
Details: *Übung.* □

Die *kleinste Fixpunktlogik* ist analog zur monotonen Fixpunktlogik definiert, mit dem Unterschied, dass man jetzt an Stelle der Monotonie fordert, dass Formeln $\varphi(R, \vec{x})$, auf die der Fixpunktoperator $\mathbf{lfp}(\cdot)$ angewandt werden soll, *positiv* in R sind.

4.16 ~~2.28~~ **Definition** (Kleinste Fixpunktlogik LFP). Sei σ eine Signatur.
Die Formelmenge LFP$[\sigma]$ ist induktiv durch die Regeln (A1),(A2),(A3),(BC) und (Q1) der Logik erster Stufe, sowie die folgende Regel (LFP) definiert:

(LFP) Ist $\varphi(R, \vec{x})$ eine LFP$[\sigma]$-Formel, wobei

- R eine k-stellige Relationsvariable, für ein $k \in \mathbb{N}_{\geqslant 1}$,
- $\vec{x} = x_1, .., x_k$ ein Tupel aus k verschiedenen Variablen erster Stufe, und
- φ außer R und $\vec{x}$ evtl. noch andere freie Variablen hat,

ist $\vec{t} = t_1, .., t_k$ ein k-Tupel aus Variablen erster Stufe und/oder Konstantensymbolen aus σ, und ist $\boldsymbol{\varphi}$ ***positiv in*** $\boldsymbol{R}$, so ist

$$[\mathbf{lfp}_{R,\vec{x}}\,\varphi](\vec{t})$$

eine LFP$[\sigma]$-Formel.

142

Die *Semantik* der LFP-Formeln ist genauso definiert wie die Semantik von MFP.

4.17 ~~2.29~~ **Beispiele.** (a) *Erreichbarkeit:*

Sei $G = (V, E, s, t)$ ein Graph mit zwei ausgezeichneten Knoten s, t. Dann gilt

$$G \models \big[\mathbf{lfp}_{R,x}\,(x = s \;\vee\; \exists z\,(R(z) \wedge E(z,x)))\big](t)$$

genau dann, wenn es in G einen Pfad von s nach t gibt.

Dies sieht man, indem man die einzelnen Stufen $R^0, R^1, R^2, \ldots$ des Operators $F_{\varphi,G}$ ausrechnet. Per Induktion nach i erhält man so für obige Beispielformel:

$$R^i \;=\; \{v \in V \;:\; \text{es gibt in } G \text{ einen Pfad der Länge} < i \text{ von } s \text{ nach } v\}.$$

Daher gilt für $\mathbf{lfp}(F_{\varphi,G}) = R^\infty$:

$$R^\infty \;=\; \{v \in V \;:\; \text{es gibt in } G \text{ einen Pfad von } s \text{ nach } v\}.$$

(b) *Graphzusammenhang:*

Sei $G = (V, E)$ ein Graph. Dann gilt

$$G \models \forall x\, \forall y\, \big[\mathbf{lfp}_{R,x,y}\, x = y \;\vee\; \exists z\, \big(R(x,z) \wedge E(z,y)\big)\big](x,y)$$

genau dann, wenn G zusammenhängend ist.
Für die einzelnen Stufen gilt hier in jedem Graphen G:

$$R^i \;=\; \{(u,v) \in V^2 \;:\; \text{es gibt in } G \text{ einen Pfad der Länge} < i \text{ von } u \text{ nach } v\}.$$

(c) *Eine definierbare Wortsprache:*

Sei $\sigma = \{<, P_a, P_b\}$ die Signatur, mit der Wörter $w \in \{a,b\}^*$ als σ-Strukturen $\mathfrak{A}_w$ kodiert werden (siehe Übungsblatt 1).
Wir definieren nun eine Formel $\varphi(R,x)$, die positiv in R ist, so dass für alle nicht-leeren Worte $w = w_0 \cdots w_{n-1} \in \{a,b\}^*$ gilt:

$\mathfrak{A}_w \models [\mathbf{lfp}_{R,x}\,\varphi](i) \iff$ an Position i steht in w der Buchstabe a und auf den Positionen $0, \ldots, i$ steht eine *ungerade* Anzahl von as.

Wenn wir φ so gewählt haben, gilt für jedes Wort w:

$$\mathfrak{A}_w \;\models\; \forall x\Big(\underbrace{\big(P_a(x) \wedge \neg\exists y\,(P_a(y) \wedge x < y)\big)}_{x \text{ ist die Position des letzten } a\text{s in } w} \rightarrow \neg[\mathbf{lfp}_{R,x}\,\varphi](x)\Big)$$

genau dann, wenn die Anzahl der as in w gerade ist.

Wahl der Formel $\varphi(R, x)$:

$$\begin{aligned}\varphi(R,x) \ :=\ & \Big(P_a(x) \wedge \neg\exists y\,(P_a(y) \wedge y < x)\Big) \vee \\ & \Big(\exists z\, \exists y\ \big(\ R(z) \wedge P_a(y) \wedge P_a(x) \wedge (z < y < x) \wedge \\ & \qquad \forall u\ \big((\neg(z < u < y \vee y < u < x)) \vee P_b(u)\big)\ \big)\Big).\end{aligned}$$

4.18 ~~2.30~~ **Lemma.**
Die Datenkomplexität des Auswertungsproblems für LFP *auf* Fin *liegt in* PTIME.

Beweis: Per Induktion über den Formelaufbau. Der einzige interessante Fall ist der Fixpunktoperator

$$\psi \ :=\ [\mathbf{lfp}_{R,\vec{x}}\,\varphi](\vec{t}).$$

Nach Induktionsannahme kann die Formel φ für jede Induktionsstufe R^i in polynomieller Zeit ausgewertet werden. Die Behauptung folgt damit aus Proposition ~~2.21~~. 4.9 . □

4.3 ~~2.2.3~~ Inflationäre Fixpunktlogik

Bei der Definition der *kleinsten Fixpunktlogik* wurde durch ein syntaktisches Kriterium (nämlich dadurch, dass die Formel $\varphi(R, \vec{x})$ *positiv* in R sein muss), gewährleistet, dass die Sequenz $R^0, R^1, R^2, ..$ der Induktionsstufen *induktiv* ist, d.h.

$$R^0 \subseteq R^1 \subseteq R^2 \subseteq \cdots \subseteq R^i \subseteq R^{i+1} \subseteq \cdots,$$

und daher der induktive Fixpunkt $R^\infty = R^{\mathrm{cl}(F)}$ existiert.
In diesem Abschnitt werden wir die Bedingung "$\varphi(R, \vec{x})$ positiv in R" fallenlassen und stattdessen durch explizites Vereinigen erzwingen, dass jede Induktionsstufe ihre gesamte Vorgängerstufe enthält.

4.19 ~~2.31~~ **Definition** (Inflationärer Fixpunkt $\mathbf{ifp}(F)$). Zu jeder endlichen Menge A und jeder Abbildung $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$ ist die Abbildung I_F wie folgt definiert:

$$\begin{aligned} I_F\ :\ \mathrm{Pot}(A) &\ \to\ \mathrm{Pot}(A) \\ X &\ \mapsto\ X \cup F(X).\end{aligned}$$

Offensichtlich ist I_F *inflationär* und hat daher einen induktiven Fixpunkt $R^\infty = R^{\mathrm{cl}(I_F)}$. R^∞ heißt der *inflationäre Fixpunkt von* F, geschrieben $\mathbf{ifp}(F)$.

4.20 ~~2.32~~ **Bemerkung.** Falls F *monoton* ist, so haben F und I_F die gleichen Induktionsstufen, und es gilt $\mathbf{ifp}(F) = \mathbf{lfp}(F)$.

4.21 ~~2.33~~ **Definition** (Inflationäre Fixpunktlogik IFP). Sei σ eine Signatur.
Die Formelmenge IFP$[\sigma]$ ist induktiv durch die Regeln (A1),(A2),(A3),(BC) und (Q1) der Logik erster Stufe, sowie die folgende Regel (IFP) definiert:

Wahl der Formel $\varphi(R,x)$:

$$\begin{aligned}\varphi(R,x) \;:=\; & \Big(P_a(x) \wedge \neg\exists y\,(P_a(y) \wedge y < x)\Big) \vee \\ & \Big(\exists z\,\exists y\,\big(\,R(z) \wedge P_a(y) \wedge P_a(x) \wedge (z < y < x) \wedge \\ & \qquad \forall u\,\big((\neg(z < u < y \vee y < u < x)) \vee P_b(u)\big)\,\big)\Big).\end{aligned}$$

4.18 ~~2.30~~ **Lemma.**
Die Datenkomplexität des Auswertungsproblems für LFP *auf* Fin *liegt in* PTIME.

Beweis: Per Induktion über den Formelaufbau. Der einzige interessante Fall ist der Fixpunktoperator

$$\psi \;:=\; [\mathbf{lfp}_{R,\vec{x}}\,\varphi](\vec{t}).$$

Nach Induktionsannahme kann die Formel φ für jede Induktionsstufe R^i in polynomieller Zeit ausgewertet werden. Die Behauptung folgt damit aus Proposition ~~2.24~~. 4.9 . □

4.3 ~~2.2.3~~ Inflationäre Fixpunktlogik

Bei der Definition der *kleinsten Fixpunktlogik* wurde durch ein syntaktisches Kriterium (nämlich dadurch, dass die Formel $\varphi(R,\vec{x})$ *positiv* in R sein muss), gewährleistet, dass die Sequenz $R^0, R^1, R^2, ..$ der Induktionsstufen *induktiv* ist, d.h.

$$R^0 \subseteq R^1 \subseteq R^2 \subseteq \cdots \subseteq R^i \subseteq R^{i+1} \subseteq \cdots,$$

und daher der induktive Fixpunkt $R^\infty = R^{\mathrm{cl}(F)}$ existiert.
In diesem Abschnitt werden wir die Bedingung "$\varphi(R,\vec{x})$ positiv in R" fallenlassen und stattdessen durch explizites Vereinigen erzwingen, dass jede Induktionsstufe ihre gesamte Vorgängerstufe enthält.

4.19 ~~2.31~~ **Definition** (Inflationärer Fixpunkt $\mathbf{ifp}(F)$). Zu jeder endlichen Menge A und jeder Abbildung $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$ ist die Abbildung I_F wie folgt definiert:

$$\begin{aligned} I_F \;:\; \mathrm{Pot}(A) &\;\to\; \mathrm{Pot}(A) \\ X &\;\mapsto\; X \cup F(X). \end{aligned}$$

Offensichtlich ist I_F *inflationär* und hat daher einen induktiven Fixpunkt $R^\infty = R^{\mathrm{cl}(I_F)}$.
R^∞ heißt der *inflationäre Fixpunkt von* F, geschrieben $\mathbf{ifp}(F)$.

4.20 ~~2.32~~ **Bemerkung.** Falls F *monoton* ist, so haben F und I_F die gleichen Induktionsstufen, und es gilt $\mathbf{ifp}(F) = \mathbf{lfp}(F)$.

4.21 ~~2.33~~ **Definition** (Inflationäre Fixpunktlogik IFP). Sei σ eine Signatur.
Die Formelmenge IFP$[\sigma]$ ist induktiv durch die Regeln (A1),(A2),(A3),(BC) und (Q1) der Logik erster Stufe, sowie die folgende Regel (IFP) definiert:

(IFP) Ist $\varphi(R, \vec{x})$ eine IFP[σ]-Formel, wobei

- R eine k-stellige Relationsvariable, für ein $k \in \mathbb{N}_{\geqslant 1}$,
- $\vec{x} = x_1, \ldots, x_k$ ein Tupel aus k verschiedenen Variablen erster Stufe, und
- φ außer R und $\vec{x}$ evtl. noch andere freie Variablen hat,

und ist $\vec{t} = t_1, \ldots, t_k$ ein k-Tupel aus Variablen erster Stufe und/oder Konstantensymbolen aus σ, so ist

$$[\mathbf{ifp}_{R,\vec{x}}\, \varphi](\vec{t})$$

eine IFP[σ]-Formel.

4.22 ~~2.34~~ **Definition** (Semantik von IFP[σ]-Formeln). Die Semantik von IFP[σ]-Formeln der Form $\psi := [\mathbf{ifp}_{R,\vec{x}}\, \varphi](\vec{t})$ ist folgendermaßen definiert:
Ist *frei*$(\psi) = \{\vec{u}, \vec{S}\}$ und ist $\mathfrak{A}$ eine endliche $(\sigma \dot\cup \{\vec{u}, \vec{S}\})$-Struktur, so gilt:

$$\mathfrak{A} \models [\mathbf{ifp}_{R,\vec{x}}\, \varphi](\vec{t}) \quad :\Longleftrightarrow \quad \vec{t}^{\,\mathfrak{A}} \in \mathbf{ifp}(F_{\varphi,\mathfrak{A}}).$$

4.23 ~~2.35~~ **Beispiel.** Sei $\sigma = \{<, P_a, P_b\}$ die Signatur, mit der Wörter $w \in \{a,b\}^*$ als σ-Strukturen $\mathfrak{A}_w$ kodiert werden (~~siehe Übungsblatt 1~~).
Wir definieren eine Formel $\varphi(R, x)$, so dass für alle nicht-leeren Worte $w = w_0 \cdots w_{n-1} \in \{a,b\}^*$ gilt:

$$\mathfrak{A}_w \models \forall u\; [\mathbf{ifp}_{R,x}\, \varphi](u) \quad \Longleftrightarrow \quad w \in \{a^n b^n \,:\, n \geqslant 1\}.$$

Wir wählen $\varphi(R, x) :=$

$$\Big(\text{“}P_a(0)\text{”} \;\wedge\; \text{“}P_b(max)\text{”} \;\wedge\; \big(\text{“}x = 0\text{”} \vee \text{“}x = max\text{”}\big)\Big) \quad \vee$$
$$\exists y\, \exists z\, \Big(\; y < z \;\wedge\; R(y) \;\wedge\; R(z) \;\wedge\; \forall v\, \big(\text{“}y < v < z\text{”} \rightarrow \neg R(v)\big) \;\wedge\; \text{“}P_a(y{+}1)\text{”} \;\wedge\; \text{“}P_b(z{-}1)\text{”} \;\wedge\; \big(\text{“}x = y{+}1\text{”} \vee \text{“}x = z{-}1\text{”}\big)\Big).$$

Für die i-te Induktionsstufe R^i des *inflationären* Fixpunkts von F gilt dann:

$$R^i \;=\; \{0, \ldots, j, n{-}1, \ldots, n{-}1{-}j \,:\, j \underset{\text{und}}{<} i \text{ maximal, so dass } w \in a^j\{a,b\}^* b^j\}.$$
$n = |w| - 1$

Man beachte, dass $\varphi(R, x)$ *nicht positiv* in R ist, dass “$\forall u\, [\mathbf{lfp}_{R,x}\, \varphi](u)$” also *keine* LFP-Formel ist.

4.24 ~~2.36~~ **Lemma.**
Die Datenkomplexität des Auswertungsproblems für IFP *auf* Fin *liegt in* PTIME.

Beweis: Analog zum Beweis von Lemma 4.18 ~~2.30~~ für LFP. □

4.25 ~~2.37~~ **Proposition.** *Jede* LFP-*Formel ist äquivalent zu einer* IFP-*Formel (kurz:* LFP $\leqslant$ IFP*).*

145

Beweis: Per Induktion über den Formelaufbau. Der einzige interessante Fall ist der Fixpunktoperator

$$[\mathbf{lfp}_{R,\vec{x}}\,\varphi](\vec{t}),$$

wobei $\varphi(R,\vec{x})$ eine LFP-Formel ist, die *positiv* in R ist. Insbesondere ist die Abbildung $F_{\varphi,\mathfrak{A}}$ *monoton* für alle Strukturen $\mathfrak{A}$. Gemäß Induktionsannahme gibt es eine IFP-Formel $\varphi'(R,\vec{x})$, die äquivalent zu $\varphi(R,\vec{x})$ ist. Insbesondere gilt $F_{\varphi',\mathfrak{A}} = F_{\varphi,\mathfrak{A}}$ für alle Strukturen $\mathfrak{A}$. Aus Bemerkung ~~2.32~~ 4.20 folgt direkt, dass die Formel

$$[\mathbf{ifp}_{R,\vec{x}}\,\varphi'](\vec{t})$$

äquivalent zur Formel $[\mathbf{lfp}_{R,\vec{x}}\,\varphi](\vec{t})$ ist. □

~~2.2.4~~ 4.4 Partielle Fixpunktlogik

Von den Lemmas ~~2.30~~ 4.18 und ~~2.36~~ 4.24 wissen wir, dass jedes Problem, das durch eine LFP-Formel oder eine IFP-Formel beschrieben werden kann, zur Komplexitätsklasse PTIME gehört. Zur Beschreibung von Problemen, deren Komplexität jenseits von PTIME liegt, eignen sich die Logiken LFP und IFP also nicht.

Um eine Logik größerer Ausdrucksstärke zu erhalten, beschränken wir im Folgenden die Aufmerksamkeit nicht mehr nur auf *induktive* Abbildungen F bzw. I_F, sondern betrachten *beliebige* Abbildungen $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$. Für diese bildet die Sequenz $R^0, R^1, R^2, ..$ der Induktionsstufen nicht mehr unbedingt eine aufsteigende Kette, und sie wird auch nicht immer stationär, erreicht also nicht notwendigerweise einen Fixpunkt.

~~2.38~~ 4.26 **Definition** (Partieller Fixpunkt $\mathbf{pfp}(F)$). Sei A eine Menge und $F : \mathrm{Pot}(A) \to \mathrm{Pot}(A)$ eine beliebige Abbildung. Der *partielle Fixpunkt* $\mathbf{pfp}(F)$ ist definiert als

$$\mathbf{pfp}(F) := \begin{cases} R^i, & \text{falls es ein } i \in \mathbb{N} \text{ gibt, so dass } R^i = R^{i+1}, \\ \varnothing, & \text{sonst} \end{cases}$$

~~2.39~~ 4.27 **Bemerkung.** Jede partielle Fixpunktinduktion über einer n-elementigen Menge A kann höchstens $2^n = |\mathrm{Pot}(A)|$ viele Induktionschritte durchlaufen, bevor entweder ein Fixpunkt erreicht oder die Induktion zyklisch wird. Es gilt:

(1) Falls $R^{2^n-1} = R^{2^n}$, so $\mathbf{pfp}(F) = R^{2^n}$.

(2) Falls $R^{2^n-1} \neq R^{2^n}$, so $\mathbf{pfp}(F) = \varnothing$.

Beweis: Es gibt nur 2^n verschiedene Teilmengen von A. Somit gibt es $i < j$ mit $i, j \in \{0, .., 2^n\}$, so dass $R^i = R^j$.
Falls $R^i = R^{i+1}$, so ist $\mathbf{pfp}(F) = R^i = R^{i+1} = R^{2^n-1} = R^{2^n}$.
Falls $R^i \neq R^{i+1}$, so ist die Folge $R^0, R^1, R^2, \ldots$ der Iterationsstufen von der Form

$$R^0, R^1, \ldots, R^{i-1}, \Big(\underbrace{\overbrace{R^i}^{=R^j}, R^{i+1}}_{\neq}, \ldots, R^{j-1}\Big)^*.$$

Daher existiert kein $k \in \mathbb{N}$ mit $R^k = R^{k+1}$. Insbesondere gilt: $R^{2^n-1} \neq R^{2^n}$ und $\mathbf{pfp}(F) = \varnothing$. $\square$

4.28 ~~2.40~~ **Bemerkung.** Für *induktive* Abbildungen $F : \text{Pot}(A) \to \text{Pot}(A)$ existiert der partielle Fixpunkt und es gilt: $\mathbf{pfp}(F) = \mathbf{ifp}(F)$.

4.29 ~~2.41~~ **Beispiel.** Im Folgenden konstruieren wir eine Formel $\varphi(R, x)$, so dass für jede linear geordnete endliche Menge $\mathfrak{A} = (A, <^{\mathfrak{A}})$ gilt:

(1) Für jede Teilmenge $X \subseteq A$ gibt es ein $i < 2^{|A|}$, so dass die i-te Induktionsstufe R^i von $F_{\varphi,\mathfrak{A}}$ genau die Menge X ist, und

(2) $\mathbf{pfp}(F_{\varphi,\mathfrak{A}}) = R^{(2^{|A|})} = A$.

Die Induktionsstufen $R^0, R^1, R^2, ..$ durchlaufen also sämtliche Teilmengen von A und enden schließlich mit der Menge A als partiellem Fixpunkt.

Idee zur Konstruktion von $\varphi(R, x)$:
Sei $A = \{a_0 <^{\mathfrak{A}} \cdots <^{\mathfrak{A}} a_{n-1}\}$. Eine Menge $X \subseteq A$ kodiert ein 0-1-Wort $w_X = w_{n-1} \cdots w_0$ — bzw. die Zahl $z_X := \sum_{i<n} w_i \cdot 2^i \in \{0, .., 2^n - 1\}$ — via

$$w_i = 1 \iff a_i \in X.$$

Die Formel $\varphi(R, x)$ zählt mit ihren Induktionsstufen alle (Binärdarstellungen von) Zahlen aus $\{0, .., 2^n - 1\}$ der Reihe nach auf. Wir wählen dazu $\varphi(R, x) :=$

$$(\forall z\, R(z)) \lor \exists y \Big(\neg R(y) \land \forall z \big(z < y \to R(z)\big) \land \big(x = y \lor (x > y \land R(x))\big)\Big).$$

Durch Betrachten der Binärdarstellungen sieht man leicht, dass

- $R^0 = \varnothing$,
- für alle $i < 2^n - 1$ gilt: $z_{R^{i+1}} = z_{R^i} + 1$, und
- für $i = 2^n - 1$ gilt: $R^i = A = R^{i+1}$.

4.30 ~~2.42~~ **Definition** (Partielle Fixpunktlogik PFP). Sei σ eine Signatur.
Die Formelmenge PFP$[\sigma]$ ist induktiv durch die Regeln (A1),(A2),(A3),(BC) und (Q1) der Logik erster Stufe, sowie die folgende Regel (PFP) definiert:

(PFP) Ist $\varphi(R, \vec{x})$ eine PFP$[\sigma]$-Formel, wobei

- R eine k-stellige Relationsvariable, für ein $k \in \mathbb{N}_{\geqslant 1}$,
- $\vec{x} = x_1, .., x_k$ ein Tupel aus k verschiedenen Variablen erster Stufe, und
- φ außer R und $\vec{x}$ evtl. noch andere freie Variablen hat,

und ist $\vec{t} = t_1, .., t_k$ ein k-Tupel aus Variablen erster Stufe und/oder Konstantensymbolen aus σ, so ist

$$[\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{t})$$

eine PFP[σ]-Formel.

4.31
~~2.43~~ **Definition** (Semantik von PFP[σ]-Formeln). Die Semantik von PFP[σ]-Formeln der Form $\psi := [\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{t})$ ist folgendermaßen definiert:
Ist *frei*$(\psi) = \{\vec{u}, \vec{S}\}$ und ist $\mathfrak{A}$ eine endliche $(\sigma\dot{\cup}\{\vec{u}, \vec{S}\})$-Struktur, so gilt:

$$\mathfrak{A} \models [\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{t}) \quad :\iff \quad \vec{t}^{\,\mathfrak{A}} \in \mathbf{pfp}(F_{\varphi,\mathfrak{A}}).$$

4.32
~~2.44~~ **Lemma.**
Die Datenkomplexität des Auswertungsproblems für PFP *auf* Fin *liegt in* PSPACE.

Beweis: Per Induktion über den Formelaufbau. Der einzige interessante Fall ist der Fixpunktoperator

$$[\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{t}).$$

Gemäß Induktionsannahme kann für jede Induktionsstufe R^i die Formel $\varphi(R, \vec{x})$ in einer endlichen Struktur $\mathfrak{A}$ auf Platz polynomiell in $|A|$ ausgewertet werden.
Jede Induktionsstufe R^i ist eine Teilmenge von A^k (mit $k := ar(R)$) und kann somit auf Platz polynomiell in $|A|$ gespeichert werden.
Um R^{i+1} aus R^i zu berechnen, braucht man *nur 2 Stufen* zu speichern (nämlich R^i und R^{i+1}). Sind R^i und R^{i+1} berechnet und gespeichert, so kann man ohne zusätzlichen Platzaufwand testen, ob $R^i = R^{i+1}$, der partielle Fixpunkt also erreicht ist. Wenn bei $i = 2^{(|A|^k)}$ immer noch kein partieller Fixpunkt erreicht wurde, so muss ein Zyklus eingetreten sein, und man weiß, dass $\mathbf{pfp}(F_{\varphi,\mathfrak{A}}) = \varnothing$ ist.
Insgesamt kann man die Formel $[\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{t})$ also auf polynomiellem Platz auswerten. □

4.33
~~2.45~~ **Proposition.** *Jede* IFP*-Formel ist äquivalent zu einer* PFP*-Formel (kurz:* IFP $\leqslant$ PFP*).*

Beweis: Per Induktion über den Formelaufbau. Der einzige interessante Fall ist der Fixpunktoperator

$$[\mathbf{ifp}_{R,\vec{x}}\,\varphi](\vec{t}).$$

Für inflationäre Abbildungen existiert der partielle Fixpunkt immer und ist identisch mit dem inflationären Fixpunkt. Es gilt daher:

$$[\mathbf{ifp}_{R,\vec{x}}\,\varphi](\vec{t}) \quad \text{ist äquivalent zu} \quad [\mathbf{pfp}_{R,\vec{x}}\,\big(R(\vec{x}) \vee \varphi\big)](\vec{t}).$$

□

4.5 Fixpunktlogiken und $L^{\omega}_{\infty\omega}$

5 Fixpunktlogiken

In Kapitel 2.2 haben wir verschiedene Erweiterungen der Logik erster Stufe um Fixpunkt-Konstrukte kennengelernt, insbesondere die Logiken LFP, IFP und PFP. Standen dort vor allem die Anwendung dieser Logiken zur Beschreibung von Komplexitätsklassen im Vordergrund, so werden wir in diesem Kapitel genauer auf die Eigenschaften von Fixpunktlogiken an sich eingehen.

Anschließend an Abschnitt 3.3 werden wir zunächst zeigen, dass sich alle bisher behandelten Fixpunktlogiken in die Logik $L^{\omega}_{\infty\omega}$ einbetten lassen. Anschließend werden wir auf eine syntaktische Variante solcher Logiken eingehen, die das Aufschreiben von Formeln stark vereinfacht und modularisiert.

5.1 Fixpunktlogiken und $L^{\omega}_{\infty\omega}$

Zur Erinnerung: In Definition 1.11 hatten wir bereits festgelegt, dass

- Fin die Klasse aller endlichen Strukturen bezeichnet (über beliebigen Signaturen),
- FinOrd die Klasse aller endlichen linear geordneten Strukturen bezeichnet (über beliebigen Signaturen, die ein 2-stelliges Relationssymbol $<$ enthalten, das stets durch eine lineare Ordnung auf dem Universum der jeweiligen Struktur interpretiert wird).

4.34 ~~5.1~~ **Definition.** (a) Für Logiken $\mathscr{L}$ und $\mathscr{L}'$ und eine Klasse C von Strukturen schreiben wir "$\mathscr{L} \leqslant \mathscr{L}'$ *auf* C", falls es für jede Formel $\varphi \in \mathscr{L}$ eine Formel $\varphi' \in \mathscr{L}'$ gibt, die zu φ auf C äquivalent ist, d.h. $\mathit{frei}(\varphi') = \mathit{frei}(\varphi)$ und für alle $\mathfrak{A} \in \mathrm{C}$ und alle Interpretationen $\vec{a} \in A$ der freien Variablen von φ gilt: $\mathfrak{A} \models \varphi[\vec{a}] \iff \mathfrak{A} \models \varphi'[\vec{a}]$.

(b) Entsprechend schreiben wir "$\mathscr{L} = \mathscr{L}'$ *auf* C", falls $\mathscr{L} \leqslant \mathscr{L}'$ und $\mathscr{L}' \leqslant \mathscr{L}$ auf C. Analog schreiben wir "$\mathscr{L} < \mathscr{L}'$ *auf* C" , falls $\mathscr{L} \leqslant \mathscr{L}'$ auf C und $\mathscr{L}' \not\leqslant \mathscr{L}$ auf C.

(c) Ist $\mathscr{K}$ eine Komplexitätsklasse, so schreiben wir "$\mathscr{L} = \mathscr{K}$ *auf* C", falls $\mathscr{L}$ die Klasse $\mathscr{K}$ auf C *beschreibt* ~~(im Sinne von Definition 2.1)~~.

(d) Entsprechend schreiben wir "$\mathscr{L} \leqslant \mathscr{K}$ *auf* C", falls die Datenkomplexität des Auswertungsproblems für $\mathscr{L}$ auf C in $\mathscr{K}$ liegt (im Sinne von Definition 2.1 (a)). Analog schreiben wir "$\mathscr{K} \leqslant \mathscr{L}$ *auf* C", falls jedes Problem in $\mathscr{K}$ durch einen Satz in $\mathscr{L}$ beschrieben werden kann (im Sinne von Definition 2.1 (b)).

4.35 ~~5.2~~ **Satz.** *Es gilt:* LFP $\leqslant$ IFP $\leqslant$ PFP $<$ $L^{\omega}_{\infty\omega}$ *auf* Fin.

149

Proposition 4.25 und 4.33

Beweis: LFP $\leqslant$ IFP $\leqslant$ PFP auf Fin wurde schon in ~~Kapitel 2.2 (Proposition 2.37 und 2.45)~~ gezeigt.

Wir zeigen als nächstes, dass $L^{\omega}_{\infty\omega} \not\leqslant$ PFP. Sei dazu $J \subseteq \mathbb{N}$ eine unentscheidbare Menge, z.B. die Menge der Gödel-Nummern aller Turing-Maschinen, die bei leerer Eingabe halten. Sei φ_J der in Beispiel 3.4 konstruierte $L^{3}_{\infty\omega}$-Satz, der genau die Strukturen $\mathfrak{A} := (A, <^{\mathfrak{A}})$ als Modelle hat, für die $|A| \in J$ und $<^{\mathfrak{A}}$ eine lineare Ordnung auf A ist. Da J unentscheidbar ist, ist auch $\text{Mod}_{\text{Fin}}(\varphi)$ unentscheidbar. Andererseits ist jede in PFP definierbare Klasse von Strukturen entscheidbar (und kann laut Lemma ~~2.44~~ 4.32 sogar in PSPACE entschieden werden). Folglich kann es keinen zum $L^{3}_{\infty\omega}$-Satz φ_J äquivalenten PFP-Satz geben.

Es bleibt zu zeigen, dass PFP $\leqslant L^{\omega}_{\infty\omega}$ auf Fin, d.h. dass jede PFP-Formel φ auf Fin äquivalent ist zu einer $L^{\omega}_{\infty\omega}$-Formel $\hat{\varphi}$. Der Beweis folgt per Induktion über den Formelaufbau. Der einzige interessante Fall ist, wenn φ die Form

$$[\text{pfp}_{R,\vec{x}}\,\psi](\vec{t})$$

o.B.d.A gilt: $frei(\hat{\psi}) = frei(\psi) = \{x_1, \dots, x_r\}$

hat. Dabei sei $r := ar(R)$, $\vec{x} = x_1, \dots, x_r$ und $\vec{t} = t_1, \dots, t_r$.

Gemäß Induktionsannahme ist ψ äquivalent zu einer $L^{\omega}_{\infty\omega}$-Formel $\hat{\psi}$. Wegen $L^{\omega}_{\infty\omega} = \bigcup_{k \in \mathbb{N}} L^{k}_{\infty\omega}$ gibt es ein $k \in \mathbb{N}$ so dass $\hat{\psi} \in L^{k}_{\infty\omega}$. Insbesondere kommen in $\hat{\psi}$ nur k verschiedene Variablen erster Stufe vor, und ~~$frei(\hat{\psi}) = frei(\psi) \supseteq \{x_1, \dots, x_r\}$~~.

Seien $\vec{y} := y_1, \dots, y_r$ neue Variablen erster Stufe, die *nicht* in $\hat{\psi}$ vorkommen. Wir definieren induktiv für jedes $\alpha \in \mathbb{N}$ eine $L^{k+r}_{\infty\omega}$-Formel $\hat{\psi}^{\alpha}(\vec{x})$, die die α-te Stufe der partiellen Fixpunktinduktion über $\hat{\psi}$ definiert:

- $\hat{\psi}^{0}(\vec{x}) :=$ ~~$\neg x_1 = x_1$~~ $\left(\bigwedge_{i=1}^{r} \neg x_i = x_i\right)$ (d.h. $\hat{\psi}^{0}$ ist eine Formel, die *nie* erfüllt ist)

- $\hat{\psi}^{\alpha+1}(\vec{x})$ entsteht aus $\hat{\psi}(\vec{x})$, indem jedes Vorkommen eines Atoms der Form $R(\vec{u})$, für ein Tupel $\vec{u} = u_1, \dots, u_r$ von Variablen erster Stufe und/oder Konstantensymbolen aus σ durch die (zur Formel $\hat{\psi}^{\alpha}(\vec{u})$ äquivalenten) Formel

$$\exists y_1 \cdots \exists y_r \Big(\bigwedge_{i=1}^{r} y_i{=}u_i \;\wedge\; \exists x_1 \cdots \exists x_r \,(\, (\bigwedge_{i=1}^{r} x_i{=}y_i) \wedge \hat{\psi}^{\alpha}(\vec{x}) \,)\Big)$$

  ersetzt wird.[1]

  Hierbei ist die Verwendung der neuen Variablen $y_1, \dots, y_r$ nötig, da die Variablen $x_1, \dots, x_r$ als Terme in $\vec{u}$ vorkommen könnten.

Per Induktion nach α zeigt man leicht, dass für jede Struktur $\mathfrak{A}$ und alle $\vec{a} \in A^r$ gilt:

$$\vec{a} \in R^{\alpha} \quad\Longleftrightarrow\quad \mathfrak{A} \models \hat{\psi}^{\alpha}[\vec{a}],$$

[1] Zur Erinnerung: In Kapitel 3 hatten wir vereinbart, *Substitutionen* in Bezug auf k-Variablen Logiken zu vermeiden und lieber explizite Variablenumbenennungen zu verwenden.

Aus dem 0-1-Gesetz für $L^{\omega}_{\infty\omega}$ folgt direkt:

4.36 Korollar

Jede der Logiken LFP, IFP und PFP besitzt das 0-1-Gesetz bzgl ALL(σ), für jede endliche relationale Signatur σ und bzgl der Klasse UG aller ungerichteten Graphen

wobei R^α die α-te Stufe der partiellen Fixpunktinduktion über ψ in $\mathfrak{A}$ bezeichnet. Daher ist die Formel $\varphi := [\mathbf{pfp}_{R,\vec{x}}\,\psi](\vec{t})$ äquivalent zur $L^{k+r}_{\infty\omega}$-Formel

$$\hat{\varphi} \quad := \quad \bigvee_{\alpha\in\mathbb{N}} \Big(\hat{\psi}^\alpha(\vec{t}) \;\wedge\; \forall\vec{x}\,\big(\hat{\psi}^\alpha(\vec{x}) \leftrightarrow \hat{\psi}^{\alpha+1}(\vec{x})\big) \Big).$$

Hierbei steht $\hat{\psi}^\alpha(\vec{t})$ wiederum für $\exists\vec{y}\,\Big(\bigwedge_{i=1}^{r} y_i{=}t_i \;\wedge\; \exists\vec{x}\,\big(\bigwedge_{i=1}^{r} x_i{=}y_i \;\wedge\; \hat{\psi}^\alpha(\vec{x})\,\big)\Big)$. □

Aus Satz ~~5.2~~ 4.35, ~~Beispiel 3.61 und Theorem 3.62 folgt sofort:~~ und dem am Ende von Kapitel 3 Gezeigten folgt sofort:

~~5.3~~ 4.37 **Korollar.** *(a)* ~~*Die Klasse*~~ Even *ist nicht* PFP-*definierbar in* Fin.

(b) ~~*Die Klasse*~~ Hamilton kreis *ist nicht* PFP-*definierbar in* Fin.

Da man selbstverständlich die Klasse Even in Polynomialzeit entscheiden kann, folgt daraus wiederum:

~~5.4~~ 4.38 **Korollar.** *Es gilt*

wir werden später zeigen

(a) PFP < PSPACE *auf* Fin *(zum Vergleich beachte:* PFP = PSPACE *auf* ~~FinOrd~~ FinC*)*

(b) LFP ⩽ IFP < PTIME *auf* Fin *(zum Vergleich beachte:* LFP = IFP = PTIME *auf* ~~FinOrd~~ FinC*)*

(c) PTIME ⩽̸ PFP *auf* Fin

~~*(d) Falls* PTIME ≠ PSPACE, *so gilt* PFP ⩽̸ PTIME *auf* Fin.~~
~~PTIME *und* PFP *liegen vermutlich also schief zueinander.*~~

Beweis: *Übung.* □

In den folgenden Abschnitten wird die Ausdrucksstärke der verschiedenen Fixpunktlogiken genauer untersucht.

5.2 Simultane Fixpunkte

5.5 Definition. Sei $m \in \mathbb{N}_{\geqslant 1}$ und seien $A_1, .., A_m$ Mengen. Ferner sei für jedes $i \in \{1, .., m\}$ eine Abbildung

$$F_i \;:\; \mathrm{Pot}(A_1) \times \cdots \times \mathrm{Pot}(A_m) \;\to\; \mathrm{Pot}(A_i)$$

gegeben.

(a) Die *simultane Abbildung über* $F_1, .., F_m$, kurz: $\vec{F} := (F_1, .., F_m)$, ist definiert als

$$\begin{array}{rcl} \vec{F}: \;\mathrm{Pot}(A_1) \times \cdots \times \mathrm{Pot}(A_m) & \to & \mathrm{Pot}(A_1) \times \cdots \times \mathrm{Pot}(A_m) \\ \vec{R} := (R_1, \ldots, R_m) & \mapsto & \big(F_1(\vec{R}), \ldots, F_m(\vec{R})\big)\,. \end{array}$$

Zusammenfassung:

Für die drei Fixpunktlogiken LFP (kleinste Fixpunktlogik), IFP (inflationäre Fixpunktlogik) und PFP (partielle Fixpunktlogik) gilt:

$$\text{LFP} \leqslant \text{IFP} \leqslant \text{PFP}$$

(d.h.: PFP ist mindestens so ausdrucksstark wie IFP, und IFP ist mindestens so ausdrucksstark wie LFP).

~~2.2.5~~ 4.6 Fixpunktlogiken und Komplexitätsklassen

Ähnlich zum Beweis des Satzes von Fagin kann man zeigen, dass die Logiken LFP und IFP die Komplexitätsklasse PTIME auf der Klasse aller endlichen geordneten Strukturen beschreiben, und dass die Logik PFP die Komplexitätsklasse PSPACE auf der Klasse aller endlichen geordneten Strukturen beschreibt:

4.35 ~~2.46~~ **Theorem** (Satz von Immerman und Vardi, 1982). FinOrd := $\text{Fin}_<$.

(a) LFP *beschreibt* PTIME *auf* FinOrd.

(b) IFP *beschreibt* PTIME *auf* FinOrd.

Beweis: Wegen Lemma ~~2.36~~ 4.24 und Proposition ~~2.37~~ 4.25 folgt *(b)* direkt aus *(a)*. Von Lemma ~~2.30~~ 4.18 wissen wir, dass die Datenkomplexität des Auswertungsproblems für LFP auf FinOrd in PTIME liegt. Im Folgenden brauchen wir also nur noch zu beweisen, dass jedes Problem aus PTIME durch eine LFP-Formel beschrieben werden kann. Sei dazu σ eine Signatur, die u.a. ein 2-stelliges Relationssymbol $<$ enthält, und sei $\mathsf{C} \subseteq \mathsf{FinOrd}$ eine Klasse endlicher geordneter σ-Strukturen, so dass

$$L_{\mathsf{C}} := \{enc_<(\mathfrak{A}) : \mathfrak{A} \in \mathsf{C}\} \in \text{PTIME}.$$

D.h. es gibt eine deterministische Turing-Maschine $M = (Q, \Sigma, \Gamma, \Delta, q_0, F)$ und eine Konstante $k \in \mathbb{N}$, so dass M bei Eingabe (der Kodierung) einer endlichen geordneten σ-Struktur $\mathfrak{A}$ entscheidet, ob $\mathfrak{A} \in \mathsf{C}$ und dabei weniger als n^k Schritte macht. Wie beim Beweis des Satzes von Fagin bezeichnen wir mit n immer die Größe des Universums der Eingabe-Struktur $\mathfrak{A}$ und nehmen o.B.d.A. an, dass

- F_{akz} aus genau einem akzeptierenden Zustand q_{akz} besteht,
- jeder Lauf von M bei jeder Eingabe-Struktur $\mathfrak{A}$ mit $|A| \geqslant 2$ nach *genau* $n^k - 1$ Schritten in einem (akzeptierenden oder verwerfenden) Endzustand endet,
- $n^k \geqslant |enc(\mathfrak{A})|$,

152

- $Q = \{0, 1, .., s_Q\}$ für ein $s_Q \in \mathbb{N}$, Endzustand $q_{\text{akz}} = 0$, Anfangszustand $q_0 = 1$,
- $\Gamma = \{0, 1, 2, .., s_\Gamma\}$ für ein $s_\Gamma \in \mathbb{N}$, Blank-Symbol $\Box = 2$,
- $s := \max\{s_Q, s_\Gamma\}$.

Da M *deterministisch* ist, können wir die Überführungsrelation Δ als Abbildung

$$\delta \ : \ (Q \setminus F) \times \Gamma \rightarrow Q \times \Gamma \times \{-1, 0, 1\}$$

darstellen.

Unser Ziel ist, einen LFP[σ]-Satz ψ zu finden, so dass $\mathsf{C} = \text{Mod}_{\text{FinOrd}}(\psi)$, d.h. für jede endliche geordnete σ-Struktur $\mathfrak{A}$ gilt:

$$\begin{aligned}
\mathfrak{A} \models \psi \quad &\iff \quad \mathfrak{A} \in \mathsf{C} \\
&\iff \quad M \text{ akzeptiert } \mathfrak{A} \\
&\iff \quad \text{der Lauf von } M \text{ bei Eingabe } enc_<(\mathfrak{A}) \text{, endet nach } n^k{-}1 \\
&\qquad\quad\ \text{Schritten in Zustand } q_{\text{akz}}.
\end{aligned}$$

Idee zur Konstruktion von ψ: Ähnlich wie im Beweis des Satzes von Fagin werden Zeitpunkte t und Bandpositionen p in $\{0, 1, .., n^k{-}1\}$ durch k-Tupel $\vec{t} = (t_1, .., t_k)$ bzw. $\vec{p} = (p_1, .., p_k)$ über $\{0, .., n{-}1\}$ kodiert. Unter Verwendung der linearen Ordnung $<^{\mathfrak{A}}$ kann man das Universum A mit der Menge $\{0, .., n{-}1\}$ identifizieren und k-Tupel $\vec{a} \in A^k$ mit Zahlen aus $\{0, 1, .., n^k{-}1\}$.
Jeden Zustand $q \in Q = \{0, 1, .., s_Q\}$ und jedes Symbol $\gamma \in \Gamma \in \{0, 1, .., s_\Gamma\}$ werden wir in einer Struktur $\mathfrak{A}$ durch das q-größte bzw. das γ-größte Element im Universum A repräsentieren.[4]
Um die Notation etwas übersichtlicher zu machen, nehmen wir o.B.d.A. an, dass σ zwei Konstantensymbole 0 und *max* enthält, die in geordneten σ-Strukturen stets mit dem kleinsten bzw. dem größten Element der linearen Ordnung interpretiert werden.

Der Lauf der DTM M bei Eingabe $enc_<(\mathfrak{A})$ wird durch folgende $(2k + 2)$-stellige Relation $R^{\mathfrak{A}} \subseteq A^{2k+2}$ repräsentiert: $enc(\mathfrak{A}) := enc_{<^{\mathfrak{A}}}(\mathfrak{A})$

- $(0, \vec{t}, \vec{p}, 0) \in R^{\mathfrak{A}} \iff$ der Schreib-/Lesekopf steht zum Zeitpunkt $\vec{t}$ auf Bandposition $\vec{p}$.
- $(1, \vec{t}, \vec{p}, \gamma) \in R^{\mathfrak{A}} \iff$ auf Bandposition $\vec{p}$ steht zum Zeitpunkt $\vec{t}$ das Symbol γ.
- $(max, \vec{t}, \vec{max}, q) \in R^{\mathfrak{A}} \iff M$ ist zum Zeitpunkt $\vec{t}$ in Zustand q.

[4]Der LFP-Satz ψ, den wir im Folgenden konstruieren, wird daher nur für solche Strukturen $\mathfrak{A}$ korrekt sein, die mehr als $s = \max\{s_Q, s_\Gamma\}$ Elemente besitzen. Das ist aber nicht weiter schlimm, denn es gibt nur *endlich* viele verschiedene σ-Strukturen mit $\leqslant s$ Elementen, und diese kann man explizit durch eine weitere FO-Formel abhandeln.

bis auf Isomorphie

Wir konstruieren im Folgenden eine FO$[\sigma \dot\cup \{R\}]$-Formel $\varphi(R, u, \vec{t}, \vec{p}, z)$, die schrittweise die Relation $R^{\mathfrak{A}}$ so aufbaut, dass $R^{\mathfrak{A}}$ durch "$[\mathbf{lfp}_{R,u,\vec{t},\vec{p},z}\, \varphi]$" beschrieben wird. Die einzelnen Induktionsstufen entsprechen dabei den Zeitpunkten während der Berechnung von M, d.h.:

$$
\begin{aligned}
R^0 &= \varnothing, \\
R^1 &= \text{"Startkonfiguration bei Eingabe } enc(\mathfrak{A})\text{"} \\
&= \{(u, \vec{t}, \vec{p}, z) \in R^{\mathfrak{A}} \,:\, rg_{<^{\mathfrak{A}}_{lex}}(\vec{t}) = 0\}, \\
&\;\;\vdots \\
R^{i+1} &= \{(u, \vec{t}, \vec{p}, z) \in R^{\mathfrak{A}} \,:\, rg_{<^{\mathfrak{A}}_{lex}}(\vec{t}) \leqslant i\}.
\end{aligned}
$$

Wenn φ so konstruiert ist, dann gilt (beachte dazu, dass $q_{\text{akz}} = 0$):

$$M \text{ akzeptiert } \mathfrak{A} \iff \mathfrak{A} \models [\mathbf{lfp}_{R,u,\vec{t},\vec{p},z}\, \varphi](max, \vec{max}, \vec{max}, 0).$$

Die Formel φ ist von der Form

$$
\begin{aligned}
&\varphi(R, u, \vec{t}, \vec{p}, z) \;:= \\
&\exists z_0 \cdots \exists z_s \, \big(\varphi_{Zahlen}(z_0, .., z_s) \;\wedge\; \big(\big(\text{"}\vec{t} = \vec{0}\text{"} \;\wedge\; \varphi_{Start}(u, \vec{p}, z)\big) \;\vee\; \varphi_{Schritt}(R, u, \vec{t}, \vec{p}, z)\big)\big),
\end{aligned}
$$

wobei $\varphi_{Zahlen}(z_0, \ldots, z_s)$ eine FO$[<]$-Formel ist, die besagt, dass die Variablen $z_0, .., z_s$ mit den $s{+}1$ kleinsten Elementen aus dem Universum der jeweiligen Struktur belegt werden.

Um die Startkonfiguration von M bei Eingabe $enc(\mathfrak{A})$ durch die Formel φ_{Start} zu beschreiben, benutzen wir die Formeln $\zeta_0(\vec{p})$ und $\zeta_1(\vec{p})$ aus dem Beweis des Satzes von Fagin ~~(Theorem 2.8)~~, die besagen, dass in dem 0-1-Wort $enc_{<}(\mathfrak{A})$ an Position $rg^{\mathfrak{A}}_{<_{lex}}(\vec{p})$ das Symbol 0 bzw. das Symbol 1 steht. Wir wählen

$$
\begin{aligned}
&\varphi_{Start}(u, \vec{p}, z) \;:= \\
&\quad \big(\; u = 0 \;\;\wedge\; \text{"}\vec{p} = \vec{0}\text{"} \;\wedge\; z = 0\big) && \text{(Kopf auf Position 0)} \\
&\quad \vee \big(u = max \;\wedge\; \text{"}\vec{p} = \vec{max}\text{"} \;\wedge\; z = z_1\big) && (M \text{ ist im Startzustand } q_0 = 1) \\
&\quad \vee \big(\; u = z_1 \;\;\wedge\; \big(\big(z = 0 \;\wedge \zeta_0(\vec{p})\big) \vee && (\text{Bandbeschriftung: } enc_{<}(\mathfrak{A})) \\
&\qquad\qquad\qquad \big(z = z_1 \wedge \zeta_1(\vec{p})\big) \vee \\
&\qquad\qquad\qquad \big(z = z_2 \;\wedge \neg(\zeta_0(\vec{p}) \vee \zeta_1(\vec{p}))\big) \;\big) \,\big) && (\text{Beachte: Blank-Symbol } \square = 2)
\end{aligned}
$$

Für die Formel $\varphi_{Schritt}$ sei $\chi_m(\vec{p}\,', \vec{p})$ für $m \in \{-1, 0, 1\}$ eine Formel, die besagt "$\vec{p} = \vec{p}\,'{+}m$", d.h.:

$$\chi_1(\vec{p}\,', \vec{p}) := succ_{<_{lex}}(\vec{p}\,', \vec{p}), \quad \chi_{-1}(\vec{p}\,', \vec{p}) := succ_{<_{lex}}(\vec{p}, \vec{p}\,'), \quad \chi_0(\vec{p}\,', \vec{p}) := \text{"}\vec{p}\,' = \vec{p}\text{"}.$$

154

Wir wählen

$\varphi_{Schritt}(R, u, \vec{t}, \vec{p}, z)$:=

$$\exists \vec{t}' \, \exists \vec{p}' \, \Big(succ_{<_{lex}}(\vec{t}', \vec{t}) \,\wedge\, R(0, \vec{t}', \vec{p}', 0) \,\wedge \qquad \text{(Kopfpos. } \vec{p}' \text{ zum Zeitpkt. } \vec{t}' := \vec{t}-1)$$

$$\bigvee_{\substack{q' \in Q \setminus F, \gamma' \in \Gamma, \\ (q,\gamma,m) := \delta(q',\gamma')}} \Big(R(max, \vec{t}', m\vec{a}x, z_{q'}) \,\wedge\, R(z_1, \vec{t}', \vec{p}', z_{\gamma'}) \,\wedge \qquad \text{(zum Zeitpkt. } \vec{t}'\text{: in Zustand } q'\text{, liest Symbol } \gamma')$$

$$\Big(\big(u = max \,\wedge\, \vec{p} = m\vec{a}x \,\wedge\, z = z_q\big) \qquad \text{(zum Zeitpkt. } \vec{t} \text{ in Zustand } q)$$

$$\vee \; \big(u = 0 \,\wedge\, \chi_m(\vec{p}', \vec{p}) \,\wedge\, z = 0\big) \qquad \text{(zum Zeitpkt. } \vec{t} \text{ Kopfpos. } \vec{p}'+m)$$

$$\vee \; \big(u = z_1 \,\wedge\, \text{“}\vec{p}' = \vec{p}\text{”} \,\wedge\, z = z_\gamma\big) \qquad \text{(zum Zeitpkt. } \vec{t} \text{ Symbol } \gamma \text{ auf Pos. } \vec{p}')$$

$$\vee \; \big(u = z_1 \,\wedge\, \text{“}\vec{p}' \neq \vec{p}\text{”} \,\wedge\, R(z_1, \vec{t}', \vec{p}, z)\big) \Big) \Big) \Big) \qquad \text{(auf Pos. } \vec{p} \neq \vec{p}' \text{ zum Zeitpkt. } \vec{t} \text{ gleiches Symbol wie zum Zeitpkt. } \vec{t}')$$

Man beachte, dass diese Formel *positiv* in R ist. Daher ist die Formel

$$\psi \;:=\; [\mathrm{lfp}_{R,u,\vec{t},\vec{p},z}\, \varphi](max, m\vec{a}x, m\vec{a}x, 0)$$

eine LFP[σ]-Formel.
Per Induktion nach i kann man leicht für alle endlichen geordneten σ-Strukturen $\mathfrak{A}$ und alle $i \in \mathbb{N}$ zeigen, dass die $(i{+}1)$-te Induktionsstufe R^{i+1} des Operators $F_{\varphi,\mathfrak{A}}$ aus genau den Tupeln $(u, \vec{t}, \vec{p}, z) \in A^{2k+2}$ besteht, für die gilt: $j := rg_{<^{\mathfrak{A}}_{lex}}(\vec{t}) \leqslant i$ und

$$\{(u', \vec{t}', \vec{p}', z') \in R^{i+1} \;:\; \vec{t}' = \vec{t}\}$$

kodiert die Konfiguration von M bei Eingabe $enc(\mathfrak{A})$ zum Zeitpunkt j.
Daraus folgt dann direkt:

$$M \text{ akzeptiert } enc(\mathfrak{A}) \iff (max^{\mathfrak{A}}, m\vec{a}x^{\mathfrak{A}}, m\vec{a}x^{\mathfrak{A}}, 0^{\mathfrak{A}}) \in R^{n^k} \iff \mathfrak{A} \models \psi.$$

Somit sind wir fertig mit dem Beweis von Theorem ~~2.46~~ 4.35 □

Auf ähnliche Art kann man auch folgendes beweisen:

4.40 ~~2.47~~ **Theorem.** PFP *beschreibt* PSPACE *auf* FinOrd.

Beweis: Ähnlich zum Beweis von Theorem 4.35 ~~2.46~~. Beachte: Eine PSPACE-Berechnung kann evtl. aus exponentiell vielen Schritten bestehen. Daher kann nicht mehr jeder Berechnungszeitpunkt t durch ein k-Tupel $\vec{t} = (t_1, \ldots, t_k) \in A^k$ kodiert werden. Insgesamt konstruiert man für jedes Problem $\mathsf{C} \subseteq$ FinOrd in PSPACE eine FO[σ]-Formel $\varphi(R, u, \vec{p}, z)$, so dass für alle endlichen geordneten σ-Strukturen $\mathfrak{A}$ gilt:

$$\mathfrak{A} \in \mathsf{C} \iff \mathfrak{A} \models [\mathrm{pfp}_{R,u,\vec{p},z}\, \varphi](max, m\vec{a}x, 0).$$

Rest: *Übung*. □

4.41 ~~2.48~~ **Bemerkung.** In den Theoremen 4.39 ~~2.46~~ und 4.40 ~~2.47~~ ist es wichtig, dass die Strukturen in FinOrd liegen, d.h. dass die Strukturen *geordnet* sind, also eine 2-stellige Relation enthalten, von der man weiß, dass sie eine lineare Ordnung des Universums der Struktur darstellt. Beim Satz von Fagin, der besagt

$$\text{ESO beschreibt NP auf Fin,}$$

war dies nicht nötig, da man sich in einer ESO-Formel eine lineare Ordnung "raten" bzw. definieren kann. ~~In Kapitel 3 (Ehrenfeucht-Fraïssé-Spiele) werden wir beweisen, dass dies *nicht* durch eine Fixpunktformel geleistet werden kann und dass daher gilt:~~
Aus Korollar 4.38 wissen wir bereits, dass gilt:

$$\text{LFP bzw. IFP beschreibt } \textit{nicht} \text{ PTIME auf Fin,}$$

$$\text{PFP beschreibt } \textit{nicht} \text{ PSPACE auf Fin.}$$

Aus den Theoremen 4.39 ~~2.46~~ und 4.40 ~~2.47~~ ergibt sich direkt:

4.42 ~~2.49~~ **Korollar.** PSPACE = PTIME $\iff$ PFP = LFP *auf* FinOrd.

Dabei heißt "PFP = LFP auf FinOrd", dass es zu jedem PFP-Satz ψ einen LFP-Satz ψ' gibt, so dass für jede endliche geordnete Struktur $\mathfrak{A}$ gilt:

$$\mathfrak{A} \models \psi \iff \mathfrak{A} \models \psi'.$$

(Die Umkehrung gilt sowieso, da LFP $\leqslant$ PFP; vgl. Proposition 4.33 ~~2.45~~.)

4.43 ~~2.50~~ **Bemerkung.** Es gilt sogar die folgende Verschärfung von Korollar 4.42 ~~2.49~~, die als *Satz von Abiteboul und Vianu* bekannt ist ~~und die wir in Kapitel 5 (Fixpunktlogiken) auch beweisen werden:~~

$$\text{PSPACE} = \text{PTIME} \iff \text{PFP} = \text{LFP auf FinOrd} \iff \text{PFP} = \text{LFP auf Fin.}$$

Ähnlich wie beim Beweis des Satzes von Cook, bei dem wir die logische Charakterisierung von NP genutzt haben, um die NP-Vollständigkeit des aussagenlogischen Erfüllbarkeitsproblems nachzuweisen, können wir die logische Charakterisierung von PSPACE nutzen, um die PSPACE-Vollständigkeit des Auswertungsproblems für FO zu beweisen ~~(dies wurde in Kapitel 1, Satz 1.61 schon erwähnt, aber noch nicht bewiesen):~~

4.44 ~~2.51~~ **Satz.** *Die kombinierte Komplexität des Auswertungsproblems für* FO *auf* FinOrd *ist* PSPACE-*vollständig.*

Beweis: Man sieht leicht ~~Mit Satz 1.60 wurde bereits gezeigt~~, dass die kombinierte Komplexität des Auswertungsproblems für FO in PSPACE liegt. Im Folgenden zeigen wir, dass das Problem auch PSPACE-hart ist, d.h. dass sich jedes Problem $L \in$ PSPACE auf das Auswertungsproblem für FO reduzieren lässt. Jedes Problem $L \in$ PSPACE kann man, für eine geeignete Signatur σ, mit einer Klasse $\mathsf{C} \subseteq$ FinOrd (bzw. der zugehörigen Menge L_{C}) identifizieren. Da nach Voraussetzung $L_{\mathsf{C}} \in$ PSPACE ist, liefert Theorem ~~2.47~~ 4.40, dass es einen PFP[σ]-Satz Φ gibt, so

156

dass für jede endliche geordnete σ-Struktur $\mathfrak{A}$ gilt: $\mathfrak{A} \in \mathsf{C} \iff \mathfrak{A} \models \Phi$.
Aus dem Beweis von Theorem ~~2.47~~ 4.48 folgt, dass Φ o.B.d.A. von der Form

$$[\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{max}, 0)$$

ist, wobei $\varphi(R, \vec{x}) \in \mathrm{FO}[\sigma\dot{\cup}\{R\}]$. Sei $r := ar(R)$ die Stelligkeit von R und sei $\vec{x} = x_1, .., x_r$.
Aus Bemerkung ~~2.39~~ 4.27 folgt, dass für jede endliche σ-Struktur $\mathfrak{A}$, für $n := |A|$, für $N := n^r = |A^r|$ und für die Induktionsstufen R^i des Operators $F_{\varphi,\mathfrak{A}}$ gilt:

(1) Falls $R^{2^N} = R^{2^N+1}$, so $\mathbf{pfp}(F_{\varphi,\mathfrak{A}}) = R^{2^N}$.

(2) Falls $R^{2^N} \neq R^{2^N+1}$, so $\mathbf{pfp}(F_{\varphi,\mathfrak{A}}) = \varnothing$.

Unser Ziel ist, eine Polynomialzeit-Reduktion f von $\mathsf{C} \subseteq \mathsf{FinOrd}$ auf das Auswertungsproblem für FO zu finden.

Ansatz: f bildet jedes $\mathfrak{A} \in \mathsf{FinOrd}$ (von dem man wissen will, ob es zur Klasse C gehört) auf ein Tupel $(\mathfrak{A}, \psi_{\Phi,\mathfrak{A}})$ ab, wobei $\psi_{\Phi,\mathfrak{A}}$ ein $\mathrm{FO}[\sigma]$-Satz ist, für den gilt:

(3) $\mathfrak{A} \models \psi_{\Phi,\mathfrak{A}} \iff \mathfrak{A} \in \mathsf{C} \overset{\text{def}}{\iff} \mathfrak{A} \models \Phi \overset{\text{def}}{\iff} \mathfrak{A} \models [\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{max}, 0)$.

(4) $\psi_{\Phi,\mathfrak{A}}$ ist in Zeit, die polynomiell in $|A|$ ist, berechenbar.
Insbesondere ist die Länge $|\psi_{\Phi,\mathfrak{A}}|$ der Formel $\psi_{\Phi,\mathfrak{A}}$ polynomiell in der Größe von $\mathfrak{A}$.

Idee: Sei $\mathfrak{A}$ gegeben und sei $n := |A|$ und $N := n^r$. O.B.d.A. ist $n \geqslant 2$, insbesondere werden also die Konstantensymbole 0 und *max* durch zwei verschiedene Elemente in A interpretiert.
Seien $v_1, .., v_N$ Variablen erster Stufe und seien $\vec{x}, \vec{y}_1, \dots, \vec{y}_N$ jeweils r-Tupel von Variablen erster Stufe. Für jedes $i \in \{0, .., N\}$ definieren wir induktiv eine $\mathrm{FO}[\sigma]$-Formel

$$\varphi_i(\vec{x}, \vec{y}_1, v_1, \vec{y}_2, v_2, \dots, \vec{y}_N, v_N),$$

so dass für alle $\vec{b}_1, .., \vec{b}_N \in A^r$ und alle $d_1, .., d_N \in \{0^{\mathfrak{A}}, max^{\mathfrak{A}}\}$ gilt:

$$(*)_i: \quad \{\vec{a} \in A^r : \mathfrak{A} \models \varphi_i[\vec{a}, \vec{b}_1, d_1, \dots, \vec{b}_N, d_N]\} \;=\; \underbrace{F_{\varphi,\mathfrak{A}}{}^{2^i}}_{2^i\text{-fache Anwendung des Operators } F_{\varphi,\mathfrak{A}}}\Big(\{\vec{b}_j : 1 \leqslant j \leqslant N,\ d_j \neq 0^{\mathfrak{A}}\}\Big)$$

Bevor wir die genaue Konstruktion der Formeln φ_i (für $i \in \{0, .., N\}$) angeben, zeigen wir zunächst, wie wir die gewünschte Formel $\psi_{\Phi,\mathfrak{A}}$ aus φ_N erhalten. Es gilt:

$$\begin{aligned}
\mathfrak{A} \models \Phi &\iff \mathfrak{A} \models [\mathbf{pfp}_{R,\vec{x}}\,\varphi](\vec{max}, 0) \\
&\iff (\vec{max}, 0) \in R^{2^N} \quad \text{und} \quad R^{2^N} = R^{2^N+1} \\
&\iff \mathfrak{A} \models \psi_{\Phi,\mathfrak{A}},
\end{aligned}$$

wobei

$\psi_{\Phi,\mathfrak{A}}$:=

$$\exists \vec{y}_1 \exists v_1 \cdots \exists \vec{y}_N \exists v_N \Big(\bigwedge_{j=1}^{N} \big(v_j = 0 \vee v_j = max\big) \qquad \text{(Zeile 1)}$$

$$\wedge \ \forall \vec{x} \Big(\varphi_N(\vec{x}, \vec{y}_1, 0, .., \vec{y}_N, 0) \leftrightarrow \bigvee_{j=1}^{N} (v_j \neq 0 \wedge \vec{x} = \vec{y}_j) \Big) \qquad \text{(Zeile 2)}$$

$$\wedge \ \Big(\bigvee_{j=1}^{N} \big(v_j \neq 0 \wedge \vec{y}_j = (\vec{max}, 0)\big) \Big) \qquad \text{(Zeile 3)}$$

$$\wedge \ \forall \vec{x} \Big(\big(\bigvee_{j=1}^{N} (v_j \neq 0 \wedge \vec{x} = \vec{y}_j) \big) \leftrightarrow \varphi\big(\vec{x}, \frac{R(\vec{u})}{\bigvee_{j=1}^{N} (v_j \neq 0 \wedge \vec{u} = \vec{y}_j)}\big) \Big) \Big) \qquad \text{(Zeile 4)}$$

Dabei besagt

- Zeile 2, dass $R^{2^N} \stackrel{\text{def}}{=} {F_{\varphi,\mathfrak{A}}}^{2^N}(\emptyset) = \{\vec{y}_j : v_j \neq 0\}$,
- Zeile 3, dass $(\vec{max}, 0) \in R^{2^N}$,
- Zeile 4, dass $R^{2^N} = R^{2^N+1} \stackrel{\text{def}}{=} F_{\varphi,\mathfrak{A}}(R^{2^N})$.

  Hierbei bezeichnet $\varphi(\vec{x}, \frac{R(\vec{u})}{\bigvee_{j=1}^{N}(v_j \neq 0 \wedge \vec{u} = \vec{y}_j)})$ die Formel, die aus $\varphi(R, \vec{x})$ entsteht, indem jedes Vorkommen eines Atoms der Form $R(\vec{u})$ durch die Formel $\bigvee_{j=1}^{N}(v_j \neq 0 \wedge \vec{u} = \vec{y}_j)$ ersetzt wird.

Klar ist: Wenn die Formel φ_N die Eigenschaft $(*)_N$ hat und in Zeit poly(n) konstruierbar ist, so ist auch $\psi_{\Phi,\mathfrak{A}}$ in Zeit poly(n) konstruierbar und es gilt:

$$\mathfrak{A} \in \mathsf{C} \iff \mathfrak{A} \models [\mathrm{pfp}_{R,\vec{x}} \varphi](\vec{max}, 0) \iff \mathfrak{A} \models \psi_{\Phi,\mathfrak{A}}.$$

Somit ist die Abbildung, die jeder σ-Struktur $\mathfrak{A} \in$ FinOrd das Tupel $(\mathfrak{A}, \psi_{\Phi,\mathfrak{A}})$ zuordnet, eine Polynomialzeit-Reduktion von C auf das Auswertungsproblem für FO auf FinOrd.
Um den Beweis von Satz ~~2.51~~ 4.44 zu beenden, müssen wir also nur noch die gewünschte Formel φ_N konstruieren. Dazu konstruieren wir induktiv für $i \in \{0, .., N\}$ Formeln φ_i mit Eigenschaft $(*)_i$, so dass

$$|\varphi_{i+1}| \leqslant n^{\text{Konstante}} + |\varphi_i|.$$

Insgesamt ist dann

$$|\varphi_N| \leqslant N \cdot n^{\text{Konstante}} = n^r \cdot n^{\text{Konstante}} = \text{poly}(n).$$

Induktionsanfang $i = 0$: $2^i = 2^0 = 1$. Wir setzen

$$\varphi_0(\vec{x}, \vec{y}_1, v_1, .., \vec{y}_N, v_N) := \varphi\Big(\vec{x}, \frac{R(\vec{u})}{\bigvee_{j=1}^{N}(v_j \neq 0 \wedge \vec{u} = \vec{y}_j)}\Big).$$

Offensichtlich hat φ_0 die Eigenschaft $(*)_0$, und es gilt $|\varphi_0| = \mathcal{O}(|\varphi| \cdot r \cdot N) = \text{poly}(n)$.

Induktionsschritt $i \to i{+}1$: Um eine Formel $\varphi_{i+1}(\vec{x}, \vec{y}_1, v_1, .., \vec{y}_N, v_N)$ mit der Eigenschaft $(*)_{i+1}$ zu konstruieren, nutzen wir, dass für $F := F_{\varphi,\mathfrak{A}}$ gilt:

$$F^{2^{i+1}}(\{\vec{y}_j : v_j \neq 0\}) = F^{2^i}\Big(F^{2^i}\Big(\{\vec{y}_j : v_j \neq 0\}\Big)\Big),$$

setzen

$$\begin{array}{lcl} \{\vec{y}'_j : v'_j \neq 0\} & := & F^{2^i}\Big(\{\vec{y}_j : v_j \neq 0\}\Big) \quad \text{und} \\ \{\vec{y}''_j : v''_j \neq 0\} & := & F^{2^i}\Big(\{\vec{y}'_j : v'_j \neq 0\}\Big) = F^{2^{i+1}}\Big(\{\vec{y}_j : v_j \neq 0\}\Big) \end{array}$$

und nutzen die Formel φ_i, um die Mengen $\{\vec{y}'_j : v'_j \neq 0\}$ und $\{\vec{y}''_j : v''_j \neq 0\}$ zu bestimmen. Ein technisches Problem dabei ist, dass die Formel φ_{i+1} nicht *zweimal* die Formel φ_i "aufrufen" kann (einmal mit $\vec{y}_1, v_1, .., \vec{y}_N, v_N$ und dann noch mal mit $\vec{y}'_1, v'_1, .., \vec{y}'_N, v'_N$, um zuerst $\vec{y}'_1, v'_1, .., \vec{y}'_N, v'_N$ und danach $\vec{y}''_1, v''_1, .., \vec{y}''_N, v''_N$ zu ermitteln). Dann wäre nämlich φ_{i+1} *doppelt* so lang wie φ_i, und am Ende wäre

$$|\varphi_N| \geqslant 2^N \cdot |\varphi|,$$

und das ist viel zu lang als dass man φ_N noch in Zeit poly(n) berechnen könnte.
Um dies zu vermeiden, wählen wir die folgende Formel, die Allquantoren benutzt, um mit einem einzigen "Aufruf" der Formel φ_i sowohl die Werte $\vec{y}'_1, v'_1, .., \vec{y}'_N, v'_N$ als auch die Werte $\vec{y}''_1, v''_1, .., \vec{y}''_N, v''_N$ festzulegen.

$$\varphi_{i+1}(\vec{x}, \vec{y}_1, v_1, .., \vec{y}_N, v_N) :=$$

$$\begin{array}{l}
\exists \vec{y}'_1\, \exists v'_1 \cdots \exists \vec{y}'_N\, \exists v'_N\; \exists \vec{y}''_1\, \exists v''_1 \cdots \exists \vec{y}''_N\, \exists v''_N\; \Big(\\
\quad \bigvee_{j=1}^{N} (v''_j \neq 0 \wedge \vec{x} = \vec{y}''_j) \;\wedge \\
\quad \bigwedge_{j=1}^{N} (v'_j = 0 \vee v'_j = max) \;\wedge\; \bigwedge_{j=1}^{N} (v''_j = 0 \vee v''_j = max) \;\wedge \\
\quad \forall \vec{w}_1 \forall u_1 \cdots \forall \vec{w}_N \forall u_N \Big(\big(\overrightarrow{(\vec{w}, u)} = \overrightarrow{(\vec{y}, v)} \;\vee\; \overrightarrow{(\vec{w}, u)} = \overrightarrow{(\vec{y}', v')} \big) \to \\
\qquad\qquad \big(\forall \vec{x}\; \big(\varphi_i(\vec{x}, \vec{w}_1, u_1, .., \vec{w}_N, u_N) \leftrightarrow \\
\qquad\qquad\quad \Big(\big(\overrightarrow{(\vec{w}, u)} = \overrightarrow{(\vec{y}, v)} \;\wedge\; \bigvee_{j=1}^{N} (v'_j \neq 0 \wedge \vec{x} = \vec{y}'_j) \big) \vee \\
\qquad\qquad\quad\; \big(\overrightarrow{(\vec{w}, u)} = \overrightarrow{(\vec{y}', v')} \;\wedge\; \bigvee_{j=1}^{N} (v''_j \neq 0 \wedge \vec{x} = \vec{y}''_j) \big)\Big)\big)\big)\Big)\Big),
\end{array}$$

wobei "$\overrightarrow{(\vec{w}, u)} = \overrightarrow{(\vec{y}, v)}$" als Abkürzung für die Formel $\bigwedge_{j=1}^{N}$ ("$\vec{w}_j = \vec{y}_j$" $\wedge\, u_j = v_j$) benutzt wird.

Gemäß dieser Konstruktion hat φ_{i+1} die Eigenschaft $(*)_{i+1}$ und es gilt

$$|\varphi_{i+1}| \leqslant n^{\text{Konstante}} + |\varphi_i|.$$

Speziell für $i = N$ gilt: Die Formel φ_N hat die Eigenschaft $(*)_N$ und kann in Zeit poly(n) konstruiert werden. Somit sind wir fertig mit dem Beweis von Satz 2.51. □

2.3 TC-Logiken zur Beschreibung von LOGSPACE und NLOGSPACE

2.3.1 Die Logik TC

2.52 Definition. Sei A eine Menge, sei $k \in \mathbb{N}_{\geqslant 1}$ und sei $R \subseteq A^{2k}$.
Die *transitive Hülle* (engl.: transitive closure) $\mathbf{tc}(R)$ von R ist folgendermaßen definiert:

$$\mathbf{tc}(R) \;:=\; \left\{ (\vec{x}, \vec{y}) \in A^{2k} \;\middle|\; \begin{array}{l} \text{es gibt im Graphen } G_R := (V, E) \text{ mit } V_R := A^k \\ \text{und } E_R := \{(\vec{u}, \vec{v}) \in A^k \times A^k \,:\, (\vec{u}, \vec{v}) \in R\} \text{ einen} \\ \text{Pfad}^5 \text{der Länge} \geqslant 1 \text{ von Knoten } \vec{x} \text{ zu Knoten } \vec{y} \end{array} \right\}.$$

2.53 Definition (Transitive Hüllen-Logik TC). Sei σ eine Signatur.
Die Formelmenge TC$[\sigma]$ ist induktiv durch die Regeln (A1),(A2),(BC) und (Q1) der Logik erster Stufe, sowie die folgende Regel (TC) definiert:

(TC) Ist $\varphi(\vec{x}, \vec{y})$ eine TC$[\sigma]$-Formel, wobei

- $\vec{x}$ und $\vec{y}$ zwei k-Tupel aus paarweise verschiedenen Variablen erster Stufe sind, für ein $k \in \mathbb{N}_{\geqslant 1}$,
- φ außer $\vec{x}$ und $\vec{y}$ evtl. noch andere freie Variablen erster Stufe hat,

und sind $\vec{s}$ und $\vec{t}$ zwei k-Tupel aus Variablen erster Stufe und/oder Konstantensymbolen aus σ, so ist

$$[\mathbf{tc}_{\vec{x},\vec{y}}\,\varphi](\vec{s}, \vec{t})$$

eine TC$[\sigma]$-Formel.

2.54 Bemerkungen.

(a) Man beachte, dass es in der Logik TC *keine* Relationsvariablen gibt.

(b) Jede Formel $\varphi(\vec{x}, \vec{y})$ mit $2k$ freien Variablen $\vec{x}, \vec{y}$ definiert in jeder Struktur $\mathfrak{A}$ (der passenden Signatur) eine $2k$-stellige Relation

$$\varphi(\mathfrak{A}) \;:=\; \{\, (\vec{u}, \vec{v}) \in A^{2k} \;:\; \mathfrak{A} \models \varphi[\vec{u}, \vec{v}] \,\}$$

und einen Graph $G_{\varphi,\mathfrak{A}} := (V_{\varphi,\mathfrak{A}}, E_{\varphi,\mathfrak{A}})$ mit $V_{\varphi,\mathfrak{A}} = A^k$ und $E_{\varphi,\mathfrak{A}} = \varphi(\mathfrak{A})$.

[5] Ein Pfad der Länge $\ell \in \mathbb{N}$ ist dabei eine Folge $\vec{v}_0, \ldots, \vec{v}_\ell \in V_R$ von Knoten, so dass für alle $i < \ell$ gilt: $(\vec{v}_i, \vec{v}_{i+1}) \in E_R$.