Kapitel 1: Hanf-Normalform und Hanf-Lokalität

Definition 1.1

(a) Ein Typen-1-Zählterm der Signatur σ ist von der Form

$$\#(y).\, sph_{\tau,r}(y)$$

wobei $r \in \mathbb{N}$ und τ ein r-Typ mit 1 Zentrum über σ ist.

(b) Ein einfacher Zählterm der Signatur σ ist von der Form

$$\sum_{\tau \in L} \#(y).\, sph_{\tau,r}(y) \quad - m \, ,$$

wobei $m \in \mathbb{Z}$ und $L \subseteq \mathcal{T}_r^{\sigma,d}(1)$ für $r, d \in \mathbb{N}$ ist.

(c) Ein Hanf-Zählsatz der Signatur σ für $FO(\mathbb{P})$ (mit $\mathbb{P} \subseteq \mathcal{P}(\mathbb{Z})$) ist von der Form

$$P(t),$$

wobei $P \in \mathbb{P} \cup \{\mathbb{N}_{\geq 1}\}$ und t ein einfacher Zählterm der Signatur σ ist.

Beachte: Jeden Hanf-Zählsatz können wir als $FOC(\mathbb{P} \cup \{\mathbb{N}_{\geq 1}\})$-Formel auffassen, aber nicht unbedingt als $FO(\mathbb{P})$-Formel.

Lemma 1.2 (Beweis: Übung)

Für jeden Hanf-Zählsatz $P(t)$ für

$t = \sum_{\tau \in L} \#(y) . sph_{\tau,r}(y) - m$ mit $L \subseteq \mathcal{L}_r^{\sigma,d}(1)$

gilt für $(P+m) := \{ p+m : p \in P \} \subseteq \mathbb{Z}$:

$$P(t) \equiv \underbrace{(P+m)\left(\#(y) . \bigvee_{\tau \in L} sph_{\tau,r}(y) \right)}_{=: \psi},$$

und ψ ist ein $FO(\{(P+m)\})[\sigma]$ - Satz.

Speziell für $P := \mathbb{N}_{\geq 1}$ ist $(P+m)(\#(y).\varphi)$ äquivalent zur Formel

$\exists^{\geq m+1} y\, \varphi := \exists y_1 \dots \exists y_{m+1} \left(\bigwedge_{1 \leq i < j \leq m+1} \neg y_i = y_j \wedge \forall y \left(\left(\bigvee_{i=1}^{m+1} y = y_i \right) \to \varphi \right) \right).$

Definition 1.3 (Hanf-Normalform-Formel für $FO(\mathbb{P})$)

Sei $\mathbb{P} \subseteq \mathcal{P}(\mathbb{Z})$.

Eine HNF-Formel für $FO(\mathbb{P})$ der Signatur σ ist

eine Boolesche Kombination von

Hanf-Zählsätzen der Signatur σ für $FO(\mathbb{P})$ und

Formeln der Form $sph_{\tau,r}(x_1, \dots, x_k)$ mit

$r \in \mathbb{N}$, $k \in \mathbb{N}_{\geq 1}$ und τ ein r-Typ mit k Zentren über σ.

Definition 1.4

(a) Der Grad einer σ-Interpretation $\mathcal{I} = (\mathcal{A}, \beta)$ ist der Grad der σ-Struktur $\mathcal{A}$.

(b) Sei $d \in \mathbb{N}$.
Zwei Formeln φ, ψ der Signatur σ heißen d-äquivalent (kurz: $\varphi \equiv_d \psi$), falls für alle σ-Interpretationen $\mathcal{I}$ vom Grad $\leq d$ gilt: $\mathcal{I} \models \varphi \iff \mathcal{I} \models \psi$.

Theorem 1.5 (Schwache HNF für $FO(\mathbb{P})$ — Kuske & Schweikardt, LICS 2017)

Sei $\mathbb{P} \subseteq \mathcal{P}(\mathbb{Z})$ und sei σ eine Signatur.
Für jede $FO(\mathbb{P})[\sigma]$-Formel φ und jedes $d \in \mathbb{N}$ gibt es eine HNF-Formel ψ für $FO(\mathbb{P})$ der Signatur σ mit $\psi \equiv_d \varphi$ und $frei(\psi) = frei(\varphi)$.

Außerdem gibt es einen Algorithmus, der bei Eingabe von φ und d ein solches ψ berechnet.

Unser nächstes Ziel ist, Theorem 1.5 zu beweisen. Danach werden wir das Theorem anwenden — einerseits um effiziente Algorithmen zum Auswerten von $FO(\mathbb{P})$-Formeln zu erhalten und andererseits um zu zeigen, dass bestimmte Aussagen nicht in $FO(\mathbb{P})$ formuliert werden können.

1.1 Beweis von Theorem 1.5

Um Theorem 1.5 zu beweisen, nutzen wir zwei technische Lemmas, die im Folgenden behandelt werden.

Lemma 1.6

Sei σ eine Signatur, seien $d, r, n \in \mathbb{N}$ mit $n \geq 1$, seien $x_1, \dots, x_n, y$ $n+1$ verschiedene Variablen, sei $\overline{x} = (x_1, \dots, x_n)$, sei $\tau \in \mathcal{L}_r^{\sigma, d}(n+1)$ und sei

$$t(\overline{x}) \quad := \quad \#(y).\, \mathrm{sph}_{\tau, r}(\overline{x}, y).$$

Für jedes $R' \geq R := 3r+1$ und jedes $\rho \in \mathcal{L}_{R'}^{\sigma, d}(n)$ gibt es einen einfachen Zählterm $\hat{t}_\rho$ der Signatur σ, so dass f.a. σ-Strukturen $\mathcal{A}$ und alle Tupel $\overline{a} \in A^n$ vom R'-Typ ρ in $\mathcal{A}$ (d.h. $(\mathcal{N}_{R'}^{\mathcal{A}}(\overline{a}), \overline{a}) \cong \rho$) gilt:

$$t^{\mathcal{A}}[\overline{a}] \quad = \quad \hat{t}_\rho^{\mathcal{A}}$$

Außerdem gibt es einen Algorithmus, der $\hat{t}_\rho$ bei Eingabe von $t(\overline{x}), R', \rho$ berechnet. Des Weiteren gilt: $\hat{t}_\rho$ ist eine natürliche Zahl oder von der Form $\#(y).\, \mathrm{sph}_{\tau_1, r}(y) - m$ für ein $\tau_1 \in \mathcal{L}_r^{\sigma, d}(1)$ und ein $m \in \mathbb{N}$.

Beweis:

Wähle ein beliebiges $R' \geq R := 3r+1$ und ein beliebiges $\varrho \in \mathcal{L}^{\sigma,d}_{R'}(n)$.

Sei $\varrho = (\mathcal{S}, a'_1, \dots, a'_n)$. Setze $\bar{a}' := (a'_1, \dots, a'_n)$.

Wir wissen: $\mathcal{S} = \mathcal{N}^{\mathcal{S}}_{R'}(\bar{a}')$.

Sei $\tau = (\mathcal{J}, e_1, \dots, e_n, f)$. Setze $\bar{e} := (e_1, \dots, e_n)$.

Wir wissen: $\mathcal{T} = \mathcal{N}^{\mathcal{J}}_{r}(\bar{e}, f)$.

Fall 1: Ex. $i \in [n]$ s.d. $\mathcal{N}^{\mathcal{J}}_r(e_i, f)$ zusammenhängend ist, d.h.: $\text{dist}^{\mathcal{J}}(e_i, f) \leq 2r+1$ und $f \in N^{\mathcal{J}}_{2r+1}(e_i)$.

Dann ist $N^{\mathcal{J}}_r(f) \subseteq N^{\mathcal{J}}_{3r+1}(e_i)$, und daher ist

$$T = N^{\mathcal{J}}_r(\bar{e}, f) \subseteq N^{\mathcal{J}}_{3r+1}(\bar{e}).$$

Für jede σ-Struktur $\mathcal{A}$ und jedes Tupel $\bar{a} = (a_1, \dots, a_n) \in A^n$ vom R'-Typ ϱ in $\mathcal{A}$ gilt dann:

$$\begin{aligned}
t^{\mathcal{A}}[\bar{a}] &= \left| \{ b \in A : \mathcal{A} \models sph_{\tau,r}[\bar{a}, b] \} \right| \\
&= \left| \{ b \in A : (\mathcal{N}^{\mathcal{A}}_r(\bar{a}, b), \bar{a}, b) \cong (\mathcal{J}, \bar{e}, f) \} \right| \\
&= \left| \{ b \in N^{\mathcal{A}}_{2r+1}(a_i) : (\mathcal{N}^{\mathcal{A}}_r(\bar{a}, b), \bar{a}, b) \cong (\mathcal{J}, \bar{e}, f) \} \right|
\end{aligned}$$

$$= \underbrace{\left| \left\{ b' \in N^{\mathcal{S}}_{2r+1}(a'_i) \ : \ \left(\mathcal{N}^{\mathcal{S}}_r(\bar{a}', b'), \bar{a}', b' \right) \cong (\mathcal{J}, \bar{e}, f) \right\} \right|}_{=: \ j_{\mathcal{S},\tau} \ \in \ \mathbb{N}}$$

da $R' \geq 3r+1$ und $\left(\mathcal{N}^{\mathcal{A}}_{R'}(\bar{a}), \bar{a}\right) \cong (\mathcal{S}, \bar{a}')$ ist.

Wir sind daher fertig, indem wir

$$\hat{t}_{\mathcal{S}} := j_{\mathcal{S},\tau}$$

wählen.

<u>Fall 2:</u> F.a. $i \in [n]$ ist $\mathcal{N}^{\mathcal{J}}_r(e_i, f)$ nicht zusammenhängend, d.h. $\mathrm{dist}^{\mathcal{J}}(e_i, f) > 2r+1$ f.a. $i \in [n]$.

Setze $W_1 := N^{\mathcal{J}}_r(f)$, $\quad \tau_1 := (\mathcal{J}[W_1], f)$,

$\qquad W_2 := N^{\mathcal{J}}_r(\bar{e})$, $\quad \tau_2 := (\mathcal{J}[W_2], \bar{e})$.

Dann ist $W_1 \cap W_2 = \emptyset$, $\quad W_1 \cup W_2 = T = N^{\mathcal{J}}_r(\bar{e}, f)$, und τ ist die <u>disjunkte Vereinigung</u> von τ_1 und τ_2 (kurz: $\tau = \tau_1 \sqcup \tau_2$), wobei die disjunkte Vereinigung zweier σ-Strukturen $\mathcal{A}$ und $\mathcal{B}$ mit $A \cap B = \emptyset$ definiert ist als die σ-Struktur $\mathcal{C}$ mit Universum $C := A \cup B$ und Relationen $R^{\mathcal{C}} := R^{\mathcal{A}} \cup R^{\mathcal{B}}$ f.a. $R \in \sigma$.

Für jede σ-Struktur $\mathcal{A}$ und jedes Tupel $\bar{a} = (a_1, \dots, a_n) \in A^n$ vom R'-Typ ρ in $\mathcal{A}$

(d.h.: $(\mathcal{N}_{R'}^{\mathcal{A}}(\bar{a}), \bar{a}) \cong (\mathcal{N}_{R'}^{\rho}(\bar{a}'), \bar{a}')$) gilt:

$$t^{\mathcal{A}}[\bar{a}] = |\{ b \in A : (\mathcal{N}_r^{\mathcal{A}}(\bar{a}, b), \bar{a}, b) \cong \tau \}|$$

$$= |\{ b \in A : (\mathcal{N}_r^{\mathcal{A}}(\bar{a}, b), \bar{a}, b) \cong \tau \text{ und } (\mathcal{N}_r^{\mathcal{A}}(b), b) \cong \tau_1 \text{ und } \underbrace{(\mathcal{N}_r^{\mathcal{A}}(\bar{a}), \bar{a}) \cong \tau_2}_{\iff (\mathcal{N}_r^{\rho}(\bar{a}'), \bar{a}') \cong \tau_2 \text{, da } \bar{a} \text{ vom } R'\text{-Typ } \rho \text{ in } \mathcal{A} \text{ und } R' \geq r} \}|$$

$$= \begin{cases} 0 & \text{falls } (\mathcal{N}_r^{\rho}(\bar{a}'), \bar{a}') \not\cong \tau_2 \\ j_1^{\mathcal{A}} - j_2^{\mathcal{A}}[\bar{a}] & \text{sonst} \end{cases} , \qquad (*)$$

wobei

$$j_1^{\mathcal{A}} := |\{ b \in A : (\mathcal{N}_r^{\mathcal{A}}(b), b) \cong \tau_1 \}| = \left(\#(y).\, sph_{\tau_1, r}(y)\right)^{\mathcal{A}}$$

$$j_2^{\mathcal{A}}[\bar{a}] := |\{ b \in A : (\mathcal{N}_r^{\mathcal{A}}(b), b) \cong \tau_1 \text{ und } (\mathcal{N}_r^{\mathcal{A}}(\bar{a}, b), \bar{a}, b) \not\cong \tau \}|$$

<u>Fall 2.1</u>: $(\mathcal{N}_r^{\mathfrak{J}}(\bar{a}'), \bar{a}') \not\cong \tau_2$:

Wegen (*) sind wir fertig, indem wir

$$\hat{t}_{\mathfrak{J}} := 0$$

wählen.

<u>Fall 2.2</u>: $(\mathcal{N}_r^{\mathfrak{J}}(\bar{a}'), \bar{a}') \cong \tau_2$

Sei $\mathcal{J}$ die Menge aller $\tau' = (\mathfrak{J}', \bar{e}', f') \in \mathcal{L}_r^{\sigma,d}(n+1)$, für die gilt:

<u>(1)</u> $\tau' \not\cong \tau$,

<u>(2)</u> $\mathcal{N}_r^{\mathfrak{J}'}(f') \cong \tau_1$ und

<u>(3)</u> $\mathcal{N}_r^{\mathfrak{J}'}(\bar{e}') \cong \tau_2$.

Wegen $\tau = \tau_1 \sqcup \tau_2$ muss für jedes $\tau' = (\mathfrak{J}', \bar{e}', f') \in \mathcal{J}$ gelten: ex. $i \in [n]$ s.d. $dist^{\mathfrak{J}'}(e_i', f') \leq 2r+1$, d.h. $\mathcal{N}_r^{\mathfrak{J}'}(e_i', f')$ ist zusammenhängend.

Sei $j_{\mathfrak{J}, \tau'} \in \mathbb{N}$ wie in Fall 1 gewählt, d.h.

$$j_{\mathfrak{J}, \tau'} := |\{ b' \in N_{2r+1}(a_i') : (\mathcal{N}_r^{\mathfrak{J}}(\bar{a}', b'), \bar{a}', b') \cong \tau' \}|.$$

F.a. σ-Strukturen $\mathcal{A}$ und alle Tupel $\bar{a} \in A^n$ vom R'-Typ ρ in $\mathcal{A}$ gilt:

$$j_2^{\mathcal{A}}[\bar{a}] := \left| \{ b \in A : (\mathcal{N}_r^{\mathcal{A}}(b), b) \cong \tau_1 \text{ und } (\mathcal{N}_r^{\mathcal{A}}(\bar{a}, b), \bar{a}, b) \not\cong \tau \} \right|$$

$$= \left| \dot{\bigcup_{\tau' \in J}} \{ b \in A : (\mathcal{N}_r^{\mathcal{A}}(\bar{a}, b), \bar{a}, b) \cong \tau' \} \right|$$

$$= \sum_{\tau' \in J} \underbrace{\left| \{ b \in A : (\mathcal{N}_r^{\mathcal{A}}(\bar{a}, b), \bar{a}, b) \cong \tau' \} \right|}_{= \text{ gemäß Fall 1}}$$

$$= \sum_{\tau' \in J} j_{\rho, \tau'}$$

$$=: j_{\rho, \tau} \in \mathbb{N}$$

Wir sind daher fertig, indem wir

$$\hat{t}_\rho := \#(y) . \, sph_{\tau_1, r}(y) - j_{\rho, \tau}$$

wählen.

$\square$ Lemma 1.6

Lemma 1.7

Sei σ eine Signatur, seien $d, r, n \in \mathbb{N}$ mit $n \geq 1$,

sei $\mathcal{L}_r^{\sigma,d}(n) = \tau_1, \dots, \tau_\ell$ und seien

$\bar{x} = (x_1, \dots, x_n)$ n verschiedene Variablen.

Sei $s \in \mathbb{N}$ und seien $\chi_1, \dots, \chi_s$ Sätze der Signatur σ einer beliebigen Logik (z.B. FOC($\mathbb{P}$)[σ]-Sätze).

Sei $\psi(\bar{x})$ eine Boolesche Kombination, die aus den Sätzen $\chi_1, \dots, \chi_s$ und aus Formeln der Form

$sph_{\tau', r'}(y_1, \dots, y_{n'})$ mit $r' \leq r$, $n' \leq n$, wobei $y_1, \dots, y_{n'}$ n' verschiedene Variablen aus $\{x_1, \dots, x_n\}$ sind und τ' ein r'-Typ mit n' Zentren der Signatur σ und vom Grad $\leq d$ ist.

Für jede Menge $J \subseteq [s]$ gibt es eine Menge $I \subseteq [\ell]$,

s.d.

$$\psi_J(\bar{x}) \equiv_d \bigvee_{i \in I} sph_{\tau_i, r}(\bar{x})$$

ist, wobei $\psi_J(\bar{x})$ die Formel ist, die aus $\psi(\bar{x})$ entsteht, indem jedes Vorkommen eines Satzes χ_j ersetzt wird durch

$$\begin{cases} \text{true} := \top := \forall z\, z = z, & \text{falls } j \in J \\ \text{false} := \bot := \exists z\, \neg z = z, & \text{falls } j \notin J. \end{cases}$$

Außerdem gibt es einen Algorithmus, der bei Eingabe von $d, r, n, \psi(\bar{x}), x_1, \dots, x_s, J$ die Menge I berechnet

Beweis:

Zunächst bringen wir $\psi_J(\bar{x})$ in "Negationsnormalform", so dass es eine Boolesche Kombination von true, false und Formeln der Form $sph_{\tau', r'}(y_1, \dots, y_{n'})$ ist, bei der Negationen nur unmittelbar vor "true", "false" und "$sph_{\tau', r'}(y_1, \dots, y_{n'})$" stehen.

Dann entfernen wir alle Vorkommen von "true" und "false", indem wir folgende Regeln wiederholt anwenden:

ersetze	durch
$\neg$ true	false
$\neg$ false	true
$(\varphi \wedge \text{true})$	φ
$(\text{true} \wedge \varphi)$	φ
$(\varphi \wedge \text{false})$	false
$(\text{false} \wedge \varphi)$	false
$(\varphi \vee \text{true})$	true
$(\text{true} \vee \varphi)$	true
$(\varphi \vee \text{false})$	φ
$(\text{false} \vee \varphi)$	φ

Dies liefert eine zu $\psi_J(\overline{x})$ äquivalente Formel $\varphi(\overline{x})$, für die gilt:

(1) $\varphi(\overline{x}) = \text{true}$ oder

(2) $\varphi(\overline{x}) = \text{false}$ oder

(3) $\varphi(\overline{x})$ besteht aus Konjunktionen und Disjunktionen von Formeln der Form $sph_{\tau',r'}(y_1,\dots,y_{m'})$ oder $\neg sph_{\tau',r'}(y_1,\dots,y_{m'})$

In Fall (1) sind wir fertig, indem wir $I := [\ell]$ wählen.
In Fall (2) sind wir fertig, indem wir $I := \emptyset$ wählen (per Definition ist $\bigvee_{i\in\emptyset} sph_{\tau_i,r}(\overline{x})$ die Formel false).

In Fall (3) gehen wir wie folgt vor:

Schritt 1: Ersetze jede in $\varphi(\overline{x})$ vorkommende Formel $sph_{\tau',r'}(y_1,\dots,y_{m'})$ durch eine dazu d-äquivalente Disjunktion von Formeln $sph_{\tau_j,r}(\overline{x})$ mit $\tau_j \in \mathcal{L}_r^{od}(n)$:

Sei $(y_1,\dots,y_{m'}) = (x_{i_1},\dots,x_{i_{m'}})$. Sei $\tau' = (\mathcal{J}', e'_1,\dots,e'_{m'})$.

Sei $\mathcal{J}' := \{ j \in [\ell] :$ für $(\mathcal{J}, e_1,\dots,e_n) := \tau_j$ ist
$$\left(N_{r'}^{\mathcal{J}}(e_{i_1},\dots,e_{i_{m'}}), e_{i_1},\dots,e_{i_{m'}}\right) \cong \tau' \}.$$

Dann ist $$sph_{\tau',r'}(y_1,\dots,y_{m'}) \equiv_d \bigvee_{j\in\mathcal{J}'} sph_{\tau_j,r}(\overline{x}).$$

Sei $\varphi_1(\bar{x})$ die in Schritt 1 aus $\varphi(\bar{x})$ resultierende Formel. <u>Beachte:</u> $\varphi_1(\bar{x})$ ist eine Boolesche Kombination von Formeln der Form $sph_{\tau_j,r}(\bar{x})$ mit $j \in [\ell]$.

<u>Schritt 2:</u> Wende wiederholt die De Morgan'sche Regel an, um Negationszeichen nach innen zu schieben. Danach ersetzen wir jedes Vorkommen einer Formel der Form $\neg\, sph_{\tau_j,r}(\bar{x})$ durch die dazu d-äquivalente Formel $\bigvee_{i \in [\ell] \setminus \{j\}} sph_{\tau_i,r}(\bar{x})$.

Sei $\varphi_2(\bar{x})$ die dadurch aus $\varphi_1(\bar{x})$ resultierende Formel.

<u>Beachte:</u> $\varphi_2(\bar{x})$ besteht aus Konjunktionen und Disjunktionen von Formeln der Form $sph_{\tau_i,r}(\bar{x})$ mit $i \in [\ell]$.

<u>Schritt 3:</u> Eliminiere alle Konjunktionen in $\varphi_2(\bar{x})$ wie folgt: Da $\tau_1, \ldots, \tau_\ell = \mathcal{L}_r^{\sigma,d}(n)$ ist, gilt f.a. $i, i' \in [\ell]$:

$$sph_{\tau_i,r}(\bar{x}) \wedge sph_{\tau_{i'},r}(\bar{x}) \equiv \begin{cases} sph_{\tau_i,r}(\bar{x}) & \text{falls } i = i' \\ \text{false} & \text{sonst} \end{cases}$$

Anwenden der Distributivitätsregel liefert f.a. $m \in \mathbb{N}_{\geq 1}$ und alle $I_1, \dots, I_m \subseteq [\ell]$, dass

$$\bigwedge_{j \in [m]} \Big(\bigvee_{i \in I_j} sph_{\tau_i, r}(\bar{x}) \Big)$$

$$\equiv \bigvee_{i_1 \in I_1} \cdots \bigvee_{i_m \in I_m} \Big(sph_{\tau_{i_1}, r}(\bar{x}) \wedge \dots \wedge sph_{\tau_{i_m}, r}(\bar{x}) \Big)$$

$$\equiv \bigvee_{i \in I} sph_{\tau_i, r}(\bar{x})$$

für $I := I_1 \cap \dots \cap I_m$.

Während eines bottom-up-Durchlaufs durch den Syntaxbaum von $\varphi_2(\bar{x})$ wenden wir diese Äquivalenz an, um alle $\wedge$-Symbole zu eliminieren.

Die resultierende Formel $\varphi_3(\bar{x})$ ist von der Form $\bigvee_{i \in I} sph_{\tau_i, r}(\bar{x})$ für eine Menge $I \subseteq [\ell]$.

$\square$ Lemma 1.7

Beweis von Theorem 1.5

Sei $d \in \mathbb{N}$ beliebig gewählt. Seien $\bar{x} = (x_1, \dots, x_n)$ n versch. Variablen, für $n \in \mathbb{N}$.
Per Induktion nach dem Aufbau von φ konstruieren wir für jede $\mathrm{FO}(\mathcal{P})[\sigma]$-Formel φ mit $\mathrm{frei}(\varphi) \subseteq \{x_1, \dots, x_n\}$ eine HNF-Formel ψ für $\mathrm{FO}(\mathcal{P})[\sigma]$ der Signatur σ mit $\mathrm{frei}(\psi) = \{x_1, \dots, x_n\}$ und $\psi \equiv_d \varphi$.

Induktionsanfang:

φ ist von der Form $x_{i_1} = x_{i_2}$ oder $R(x_{i_1}, \dots, x_{i_k})$ mit $R \in \sigma$, $k = ar(R)$, $i_1, \dots, i_k \in \{1, \dots, n\}$.

$$\text{Sei } \mathcal{J} := \left\{ \tau \in \mathcal{L}_0^{\sigma,d}(n) : \text{für } (\mathcal{J}, e_1, \dots, e_n) = \tau \text{ gilt } \mathcal{J} \models \varphi\left[\frac{e_{i_1}, \dots, e_{i_k}}{x_{i_1}, \dots, x_{i_k}}\right] \right\}$$

$$\text{Es gilt: } \quad \varphi \equiv \underbrace{\bigvee_{\tau \in \mathcal{J}} sph_{\tau, r}(\bar{x})}_{=: \psi} \quad \text{— ist eine HNF-Formel!}$$

Induktionsschritt:

Fall 1: φ ist von der Form $\neg\varphi'$ oder von der Form $(\varphi' \vee \varphi'')$. Die Induktionsannahme liefert HNF-Formeln $\psi'(\bar{x})$ und $\psi''(\bar{x})$ mit $\psi'(\bar{x}) \equiv_d \varphi'$ und $\psi''(\bar{x}) \equiv_d \varphi''$

Wir sind fertig, indem wir wählen:

$$\psi(\bar{x}) := \begin{cases} \neg\psi'(x) & \text{falls } \varphi = \neg\varphi' \\ (\psi'(\bar{x}) \vee \psi''(\bar{x})) & \text{falls } \varphi = (\varphi' \vee \varphi'') \end{cases}$$

Fall 2: φ ist von der Form $\exists y\, \varphi'$.

Dann ist $\varphi \equiv \mathbb{N}_{\geq 1}\left(\#(y).\, \varphi'\right)$.

Wir ersetzen φ durch die $FO(\mathbb{P} \cup \{\mathbb{N}_{\geq 1}\})$-Formel $\mathbb{N}_{\geq 1}(\#(y).\varphi')$ und behandeln diese laut dem folgenden Fall 3.

Fall 3: φ ist von der Form $P\left(\#(y).\, \varphi'\right)$ mit $P \in \mathbb{P} \cup \{\mathbb{N}_{\geq 1}\}$. Wegen $frei(\varphi) \subseteq \{x_1, \dots, x_n\}$ ist $frei(\varphi') \subseteq \{x_1, \dots, x_n, y\}$

Gemäß Induktionsannahme können wir eine zu $\varphi'(\bar{x}, y)$ d-äquivalente HNF-Formel $\psi'(\bar{x}, y)$ mit $frei(\psi') = \{x_1, \dots, x_n, y\}$ konstruieren.

F.a. σ-Strukturen $\mathcal{A}$ vom Grad $\leq d$ und alle $\bar{a} \in A^n$ gilt:

$\mathcal{A} \models \varphi[\bar{a}]$

$\iff \left(\#(y).\ \varphi'(\bar{x},y)\right)^{\mathcal{A}}[\bar{a}] \in \mathbb{P}$

$\iff \left(\#(y).\ \psi'(\bar{x},y)\right)^{\mathcal{A}}[\bar{a}] \in \mathbb{P}.$ $\quad (*)_1$

Da $\psi'(\bar{x},y)$ eine HNF-Formel ist, ist sie insbes. eine Boolesche Kombination Sätzen der Signatur σ und von Formeln der Form $\mathrm{sph}_{\tau,r'}(z_1,\dots,z_{n'})$ mit $r' \geq 0$, $1 \leq n' \leq m+1$, $z_1,\dots,z_{n'} \in \{x_1,\dots,x_m,y\}$, τ ein r'-Typ mit n' Zentren über σ vom Grad $\leq d$.

Sei $\chi_1,\dots,\chi_s$ eine Liste aller in dieser Booleschen Kombination vorkommenden Sätze, und sei r das maximale in $\psi'(\bar{x},y)$ vorkommende r'.

Wir wenden Lemma 1.7 auf die Formel $\psi'(\bar{x},y)$ an. Für jedes $J \subseteq [s]$ ist $\psi'_J(\bar{x},y)$ die Formel, die aus $\psi'(\bar{x},y)$ entsteht, indem jedes Vorkommen eines Satzes χ_j ersetzt wir durch $\begin{cases} \text{true} & \text{falls } j \in J, \\ \text{false} & \text{falls } j \notin J. \end{cases}$

Lemma 1.7 liefert uns für jedes $J \subseteq [s]$ eine Menge $I_J \subseteq [\ell]$, so dass

$$\psi_J'(\bar{x}, y) \equiv_d \bigvee_{i \in I_J} sph_{\tau_i, r}(\bar{x}, y). \qquad \circledast_2$$

Hierbei ist $\tau_1, \dots, \tau_\ell := \mathcal{L}_r^{\sigma, d}(n+1)$.

Betrachte ein beliebiges $J \subseteq [s]$.

Sei $\chi_J := \bigwedge_{j \in J} \chi_j \wedge \bigwedge_{j \in [s] \setminus J} \neg \chi_j$.

F.a. σ-Strukturen $\mathcal{A}$ vom Grad $\leq d$ und alle $\bar{a} \in A^n$ gilt:

$$\mathcal{A} \models \chi_J \text{ und } \mathcal{A} \models \varphi[\bar{a}]$$

$$\underset{\circledast_1}{\iff} \mathcal{A} \models \chi_J \text{ und } \left(\#(y).\, \psi'(\bar{x}, y)\right)^{\mathcal{A}}[\bar{a}] \in P$$

$$\underset{\text{Def. } \psi_J'}{\iff} \mathcal{A} \models \chi_J \text{ und } \left(\#(y).\, \psi_J'(\bar{x}, y)\right)^{\mathcal{A}}[\bar{a}] \in P$$

$$\underset{\circledast_2}{\iff} \mathcal{A} \models \chi_J \text{ und } \left(\#(y).\, \bigvee_{i \in I_J} sph_{\tau_i, r}(\bar{x}, y)\right)^{\mathcal{A}}[\bar{a}] \in P$$

$$\iff \mathcal{A} \models \chi_J \text{ und } \sum_{i \in I_J} \left(\#(y).\, sph_{\tau_i, r}(\bar{x}, y)\right)^{\mathcal{A}}[\bar{a}] \in P$$

$\circledast_3$

Fall 3.1: $n=0$, dh $\bar{x}=()$. Dann ist

$$\hat{t}_J := \sum_{i\in I_J} \#(y).\, sph_{\tau_i,r}(y) \quad \text{ein}$$

einfacher Zählterm der Signatur σ, und es gilt:

$$\varphi \equiv \bigvee_{J\subseteq[s]} (\chi_J \wedge \varphi)$$

$$\underbrace{\equiv_d^{\circledast_3} \bigvee_{J\subseteq[s]} \Big(\chi_J \wedge P\Big(\sum_{i\in I_J} \#(y).\, sph_{\tau_i,r}(y)\Big)\Big)}$$

$=: \psi$ — und dies ist eine HNF-Formel mit $frei(\psi)=\emptyset$, wie gewünscht!

Fall 3.2: $n>0$.

Für jedes $i\in[e]$ wende Lemma 1.6 auf den Zählterm

$$t_i(\bar{x}) := \#(y).\, sph_{\tau_i,r}(\bar{x},y)$$

an. Setze $R' := R := 3r+1$. Für jedes $\rho\in\mathcal{L}^{\sigma,d}_{R'}(n)$ liefert Lemma 1.6 einen einfachen Zählterm $\hat{t}_{i,\rho}$ der Signatur σ, s.d. f.a. σ-Strukturen $\mathcal{A}$ und alle Tupel $\bar{a}\in A^n$ vom R'-Typ ρ in $\mathcal{A}$ gilt:

$$t_i^{\mathcal{A}}[\bar{a}] = \hat{t}_{i,\rho}^{\mathcal{A}}. \qquad \circledast_4$$

Außerdem ist $\hat{t}_{i,\rho}$ entweder eine nat. Zahl oder von der Form $\#(y).\mathrm{sph}_{\tau_i,r}(y)$ mit $\tau \in \mathcal{L}_r^{\sigma,d}(1)$.

Insgesamt gilt für $L := \mathcal{L}_{R'}^{\sigma,d}(n)$:

$$\begin{aligned}
&\varphi(\bar{x}) \\
\equiv_d\ & \bigvee_{\rho \in L} \left(\mathrm{sph}_{\rho,R'}(\bar{x}) \wedge \varphi(\bar{x}) \right) \\
\equiv\ & \bigvee_{\rho \in L} \Big(\mathrm{sph}_{\rho,R'}(\bar{x}) \wedge \bigvee_{J \subseteq [s]} \left(\chi_J \wedge \varphi(\bar{x}) \right) \Big) \\
\underset{\circledast_3}{\equiv_d}\ & \bigvee_{\rho \in L} \Big(\mathrm{sph}_{\rho,R'}(\bar{x}) \wedge \bigvee_{J \subseteq [s]} \Big(\chi_J \wedge P\Big(\sum_{i \in I_J} \underbrace{\#(y).\mathrm{sph}_{\tau_i,r}(\bar{x},y)}_{= t_i(\bar{x})} \Big) \Big) \Big) \\
\underset{\circledast_4}{\equiv_d}\ & \underbrace{\bigvee_{\rho \in L} \Big(\mathrm{sph}_{\rho,R'}(\bar{x}) \wedge \bigvee_{J \subseteq [s]} \Big(\chi_J \wedge P\Big(\sum_{i \in I_J} \hat{t}_{i,\rho} \Big) \Big) \Big)}_{=: \psi(\bar{x})}
\end{aligned}$$

Man kann sich leicht davon überzeugen, dass sich $\sum_{i \in I_J} \hat{t}_{i,\rho}$ als einfachen Zählterm darstellen lässt (Details: Übungsaufgabe!). Somit können wir $P(\sum_{i \in I_J} \hat{t}_{i,\rho})$ als Hanf-Zählsatz auffassen und erhalten insgesamt, dass $\psi(\bar{x})$ eine zu φ d-äquivalente HNF-Formel mit $\mathrm{frei}(\psi) = \{x_1, \ldots, x_n\}$ ist.

$\square$ Theorem 1.5

Bemerkung 1.8

Aus dem Beweis von Theorem 1.5 folgt direkt für die HNF-Formel ψ, die bei Eingabe einer FO(3)-Formel φ und einer Gradschranke $d \geq 2$ berechnet wird:

(a) Jede in ψ vorkommende Sphärenformel

$sph_{\sigma,r}(\bar{x})$ oder $sph_{\tau,r}(y)$ hat Radius

$$r \leq \frac{3^{qr(\varphi)}-1}{2},$$

wobei $qr(\varphi)$ der Quantorenrang von φ ist, d.h. die maximale Schachtelungstiefe von Quantoren der Form $\exists z \cdots$, $\forall z \cdots$, $P(\#(z) \cdots)$ in φ.

(b) ψ kann in Zeit

$$\exp_5\left(poly(\|\varphi\| + \|\sigma\|) + lg\,lg\,d\right) = 2^{2^{2^{d^{2^{poly(\|\varphi\| + \|\sigma\|)}}}}}$$

berechnet werden.

Beweis: (b): Übung

(a) Für $q := qr(\varphi)$ und $r := r(q)$ erhält man aus dem Beweis von Theorem 1.5 folgende Rekursionsgleichung:

$r(0) = 0$ und $r(q+1) = 3 \cdot r(q) + 1$.

Somit gilt:

$$
\begin{aligned}
r(q+1) &= 3\,r(q) + 1 \\
&= 3\,(3\,r(q-1) + 1) + 1 \\
&= 3^2\,r(q-1) + 3 + 1 \\
&= 3^2\,(3\,r(q-2) + 1) + 3 + 1 \\
&= 3^3\,r(q-2) + 3^2 + 3 + 1 \\
&= \dots \\
&= 3^{q+1}\,r(q-q) + 3^q + 3^{q-1} + \dots + 3 + 1 \\
&= 3^{q+1}\,\underbrace{r(0)}_{=0} + \sum_{i=0}^{q} 3^i \\
&= \sum_{i=0}^{q} 3^i \\
&= \frac{3^{q+1} - 1}{2}
\end{aligned}
$$

$\square$

Speziell für die Logiken FO und FO+MOD erhalten wir aus Theorem 1.5 und Bemerkung 1.8:

Folgerung 1.9 (Satz von Bollig und Kuske, 2012)

Für jede Signatur σ, jeden $FO[\sigma]$-Satz φ und jede Gradschranke $d \in \mathbb{N}$ mit $d \geq 2$ kann in Zeit $\exp_5(\mathrm{poly}(\|\varphi\| + \|\sigma\|) + \log\log d)$ ein zu φ d-äquivalenter $FO[\sigma]$-Satz ψ in Hanf-Normalform vom Radius $r := \frac{3^{qr(\varphi)} - 1}{2}$ berechnet werden, d.h.:

ψ ist eine Boolesche Kombination von $FO[\sigma]$-Sätzen der Form $\exists^{\geq m} y\ sph_{\tau,r}(y)$ mit $m \in \mathbb{N}_{\geq 1}$ und $\tau \in \mathcal{L}_r^{\sigma,d}(1)$.

Beweis:

Theorem 1.5 liefert eine zu φ d-äquivalente Boolesche Kombination von Aussagen der Form

$$P\left(\sum_{\tau \in L} \#(y).sph_{\tau,r}(y) - m\right) \text{ mit } P = \mathbb{N}_{\geq 1},\ m \in \mathbb{N},$$

$L \subseteq \mathcal{L}_r^{\sigma,d}(1)$; und gemäß Bemerkung 1.8 ist $r = \frac{3^{qr(\varphi)} - 1}{2}$.

Außerdem gilt:

$$P\Big(\sum_{\tau\in L} \#(y).\,\mathrm{sph}_{\tau,r}(y) \;-m\Big)$$

$$\equiv (P+m)\Big(\sum_{\tau\in L} \#(y).\,\mathrm{sph}_{\tau,r}(y)\Big)$$

$$\underset{P=\mathbb{N}_{\geq 1}}{\equiv} \exists^{\geq m+1} y \bigvee_{\tau\in L} \mathrm{sph}_{\tau,r}(y).$$

Sei $\ell' := |L|$ und $L = \{\tau_1, \dots, \tau_{\ell'}\}$.

Sei $I := \{(i_1, \dots, i_{\ell'}) \in \{0, 1, \dots, m+1\}^{\ell'} : \sum_{j=1}^{\ell'} i_j \geq m+1\}$.

<u>Behauptung:</u>

$$\exists^{\geq m+1} y \bigvee_{\tau\in L} \mathrm{sph}_{\tau,r}(y)$$

$$\equiv \bigvee_{(i_1,\dots,i_{\ell'})\in I} \Big(\bigwedge_{j=1}^{\ell'} \exists^{\geq i_j} y\; \mathrm{sph}_{\tau_j,r}(y)\Big)$$

<u>Beweis:</u> Übung.

$\square$ Folgerung 1.9

<u>Folgerung 1.10</u> (Heimberg, Kuske, Schweikardt, 2016)

Für jede Signatur σ, jede FO+MOD[σ]-Formel φ und jede Gradschranke $d \in \mathbb{N}$ mit $d \geq 2$ kann in Zeit $\exp_5(\mathrm{poly}(\|\varphi\| + \|\sigma\|) + \lg\lg d)$ eine zu φ d-äquivalente <u>FO+MOD[σ]-Formel ψ in Hanf-Normalform</u> vom Radius $r := \frac{3^{qr(\varphi)} - 1}{2}$ berechnet werden, d.h. ψ ist eine Boolesche Kombination von

(1) Formeln $sph_{\rho, r}(\bar{x})$ mit $\bar{x} = (x_1, \dots, x_n)$, $n := |\mathrm{frei}(\varphi)|$, $\{x_1, \dots, x_n\} = \mathrm{frei}(\varphi)$ und $\rho \in \mathcal{L}_r^{\sigma, d}(n)$,

(2) Sätzen $\exists^{\geq m} y \; sph_{\tau, r}(y)$ mit $m \in \mathbb{N}_{\geq 1}$, $\tau \in \mathcal{L}_r^{\sigma, d}(1)$, und

(3) Sätzen $\exists^{i \bmod p} y \; sph_{\tau, r}(y)$ mit $\tau \in \mathcal{L}_r^{\sigma, d}(1)$, $i \in \{0, \dots, p-1\}$, $p \in \mathbb{N}$ so dass φ eine Teilformel der Form $\exists^{j \bmod p} z \, \chi$ enthält

<u>Beweis:</u>
Ersetze in φ jede Teilformel der Form $\exists^{j \bmod p} z \, \chi$ durch $(p \cdot \mathbb{N} + j)(\#(z).\chi)$.
Theorem 1.5 und Bemerkung 1.8 liefern für $r := \frac{3^{qr(\varphi)} - 1}{2}$ eine zu φ d-äquivalente Boolesche Kombination von Formeln der Form (1) und Aussagen der Form

$$P\left(\sum_{\tau \in L} \#(y).\, sph_{\tau,r}(y) \quad - m\right) \quad \text{mit}$$

$m \in \mathbb{N}$, $L \subseteq \mathcal{L}_r^{\sigma,d}(1)$ und

$P = \mathbb{N}_{\geq 1}$ oder $P = p\mathbb{N} + j$ für p, j so, dass φ eine Teilformel der Form $\exists^{j \bmod p} z\, \chi$ enthält.

Für $P = \mathbb{N}_{\geq 1}$ können wir genauso vorgehen wie im Beweis von Folgerung 1.9.

Für $P = p\mathbb{N} + j$ beachte, dass Folgendes gilt:

$$P\left(\sum_{\tau \in L} \#(y).\, sph_{\tau,r}(y) \quad - m\right)$$

$$\equiv \quad p\mathbb{N}+j+m \left(\sum_{\tau \in L} \#(y).\, sph_{\tau,r}(y)\right)$$

$$\equiv \quad \exists^{\geq j+m} y \bigvee_{\tau \in L} sph_{\tau,r}(y) \quad \wedge \quad \exists^{i \bmod p} y \bigvee_{\tau \in L} sph_{\tau,r}(y),$$

wobei $i \in \{0, \dots, p-1\}$ so, dass $i \equiv j+m \bmod p$ ist.

Für die Formel $\exists^{\geq j+m} y \bigvee_{\tau \in L} sph_{\tau,r}(y)$ gehen wir genauso vor wie im Beweis von Folgerung 1.9

Für die Formel $\exists^{i \bmod p} y \bigvee_{\tau \in L} sph_{\tau,r}(y)$ sei

$\ell' := |L|$ und $L = \{\tau_1, \dots, \tau_{\ell'}\}$ und sei

$$I := \left\{ (i_1, \dots, i_{\ell'}) \in \{0, 1, \dots, p-1\}^{\ell'} : \sum_{j=1}^{\ell'} i_j \equiv i \bmod p \right\}.$$

Behauptung:

$$\exists^{i \bmod p} y \bigvee_{\tau \in L} sph_{\tau,r}(y)$$

$$\equiv \bigvee_{(i_1,\dots,i_{e'}) \in I} \left(\bigwedge_{j=1}^{e'} \exists^{i_j \bmod p} y \; sph_{\tau_j,r}(y) \right)$$

Beweis: Übung

□ Folgerung 1.10

1.2 Hanf-Lokalität

Definition 1.11 $(\mathcal{A} \rightleftarrows_r \mathcal{B})$

Sei σ eine Signatur, sei $r \in \mathbb{N}$.
Zwei σ-Strukturen $\mathcal{A}$ und $\mathcal{B}$ heißen r-bijektiv,
kurz: $\mathcal{A} \rightleftarrows_r \mathcal{B}$, wenn es eine Bijektion
$f: A \to B$ gibt, s.d. f.a. $a \in A$ gilt:

$$(\mathcal{N}_r^{\mathcal{A}}(a), a) \cong (\mathcal{N}_r^{\mathcal{B}}(f(a)), f(a)).$$

Definition 1.12 ($HT_r^d(\mathcal{A})$)

Sei σ eine Signatur.
Sei $r, d \in \mathbb{N}$ und $\mathcal{L}_r^{\sigma,d}(1) = \tau_1, \dots, \tau_\ell$.

Sei $\mathcal{A}$ eine σ-Struktur.
Das Hanf-Tupel für Radius r und Grad d von $\mathcal{A}$ ist das Tupel

$$HT_r^d(\mathcal{A}) := \left([\![\#(y).\, sph_{\tau_i, r}(y)]\!]^{\mathcal{A}} \right)_{i \in \{1,\dots,\ell\}} \in \mathbb{N}^\ell .$$

Bemerkung 1.13

Für alle σ-Strukturen $\mathcal{A}, \mathcal{B}$ und alle $r \in \mathbb{N}$ gilt:

$$\mathcal{A} \rightleftarrows_r \mathcal{B} \iff HT_r^d(\mathcal{A}) = HT_r^d(\mathcal{B})$$

für $d := \max\{Grad(\mathcal{A}), Grad(\mathcal{B})\}$

Beweis: Übung.

Definition 1.14

Sei σ eine Signatur, S eine Klasse von σ-Strukturen und $C \subseteq S$.

C heißt **Hanf-lokal** in S, falls es ein $r \in \mathbb{N}$ gibt, so dass f.a. $\mathcal{A}, \mathcal{B} \in S$ gilt:

$$\text{Falls } \mathcal{A} \leftrightarrows_r \mathcal{B}, \text{ so } \big(\mathcal{A} \in C \iff \mathcal{B} \in C\big).$$

Als einfache Folgerung von Theorem 1.5 erhält man:

Satz 1.15 (Hanf-Lokalität von $FO(\mathbb{P})$).

Sei σ eine Signatur, $\mathbb{P} \subseteq \mathcal{P}(\mathbb{Z})$, S eine Klasse von σ-Strukturen.

Dann gilt für jeden $FO(\mathbb{P})[\sigma]$-Satz φ:

$$\underbrace{Mod_S(\varphi)}_{:= \{\mathcal{A} \in S \,:\, \mathcal{A} \models \varphi\}} \text{ ist Hanf-lokal in } S$$

Beweis: Sei $r := \frac{3^{qr(\varphi)}-1}{2}$. Seien $\mathcal{A}, \mathcal{B} \in S$ mit $\mathcal{A} \leftrightarrows_r \mathcal{B}$.

Zu zeigen: $\mathcal{A} \models \varphi \iff \mathcal{B} \models \varphi$. Sei $d := \max\{Grad(\mathcal{A}), Grad(\mathcal{B})\}$.

Gemäß Theorem 1.5 ist φ d-äquivalent zu einer Booleschen Kombination ψ von Aussagen der Form

$P\big(\sum_{\tau \in L} \#(y).sph_{\tau,r}(y) - m\big)$ mit $m \in \mathbb{N}$, $P \in \mathbb{P} \cup \{\mathbb{N}_{\geq 1}\}$, $L \subseteq \mathcal{L}^{\sigma,d}_r(1)$.

Wegen $\mathfrak{A} \rightleftarrows_r \mathfrak{B}$ gilt gemäß Bemerkung 1.13, dass $HT_r^d(\mathfrak{A}) = HT_r^d(\mathfrak{B})$ ist. Somit ist

$$[\![\#(y).\mathrm{sph}_{\tau,r}(y)]\!]^{\mathfrak{A}} = [\![\#(y).\mathrm{sph}_{\tau,r}(y)]\!]^{\mathfrak{B}} \quad \text{f.a. } \tau \in L.$$

Daher gilt (*):

$$\mathfrak{A} \models P\Big(\sum_{\tau \in L} \#(y).\mathrm{sph}_{\tau,r}(y) - m\Big) \iff \mathfrak{B} \models P\Big(\sum_{\tau \in L} \#(y).\mathrm{sph}_{\tau,r}(y) - m\Big).$$

Und insgesamt erhalten wir:

$$\begin{aligned}
& \mathfrak{A} \models \varphi \\
\iff & \mathfrak{A} \models \psi && (\text{da } \psi \equiv_d \varphi \text{ und } \mathrm{Grad}(\mathfrak{A}) \le d) \\
\iff & \mathfrak{B} \models \psi && (\text{wegen } (*)) \\
\iff & \mathfrak{B} \models \varphi && (\text{da } \psi \equiv_d \varphi \text{ und } \mathrm{Grad}(\mathfrak{B}) \le d).
\end{aligned}$$

$\square$ Satz 1.15

<u>Bemerkung 1.16</u>

Indem man zeigt, dass eine Klasse C <u>nicht</u> Hanf-lokal in S ist, kann man (unter Verwendung von Satz 1.15) folgern, dass C nicht $FO(\mathbb{P})$-definierbar in S ist, für $\mathbb{P} := \mathcal{P}(\mathbb{Z})$

Beispiel 1.17

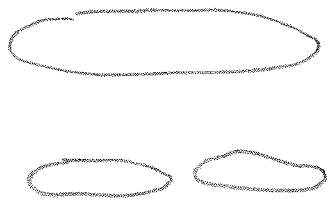
Sei $\mathcal{S} := \mathcal{S}(\mathbb{Z})$.
Graph-Zusammenhang ist nicht FO($\mathcal{S}$)-definierbar.

Beweis:

Angenommen doch, dann ist Graph-Zusammenhang Hanf-lokal in der Klasse aller endlichen Graphen. Dann gibt es also eine Zahl $r \in \mathbb{N}$ so dass für alle endlichen Graphen $\mathfrak{A}, \mathfrak{B}$ mit $\mathfrak{A} \leftrightarrows_r \mathfrak{B}$ gilt: $\mathfrak{A}$ ist genau dann zusammenhängend, wenn $\mathfrak{B}$ zusammenhängend ist.

Aber betrachte den Graphen $\mathfrak{A}$, der einen Kreis auf $2m$ Knoten bildet, für $m := 2r+2$, und den Graphen $\mathfrak{B}$, der aus der disjunkten Vereinigung von 2 Kreisen auf je m Knoten besteht. Für jedes $\mathfrak{C} \in \{\mathfrak{A}, \mathfrak{B}\}$ und jeden Knoten $c \in C$ ist $\left(\mathcal{W}_r^{\mathfrak{C}}(c), c\right)$ ein Pfad auf $2r+1$ Knoten, bei dem c der Knoten in der Mitte ist. Somit ist $\mathfrak{A} \leftrightarrows_r \mathfrak{B}$, $\mathfrak{A}$ ist zusammenhängend und $\mathfrak{B}$ ist nicht zusammenhängend. ↯ □

Skizze: $\mathfrak{A}$: $\mathfrak{B}$:



Beispiel 1.18

Sei $\mathbb{P} := \mathbb{P}(\mathbb{Z})$.
Es gibt keinen $FO(\mathbb{P})$-Satz der Signatur $\sigma := \{E\}$, der von genau denjenigen endlichen Graphen erfüllt wird, die Bäume sind.

Beweisidee:

Analog zum Beweis in Beispiel 1.17, wobei $\mathcal{A}$ und $\mathcal{B}$ so gewählt werden, dass $|A| = |B|$ und $\mathcal{A}$ ein sehr langer Pfad ist und $\mathcal{B}$ die disjunkte Vereinigung eines langen Pfades und eines Kreises auf $2r+2$ Knoten ist.

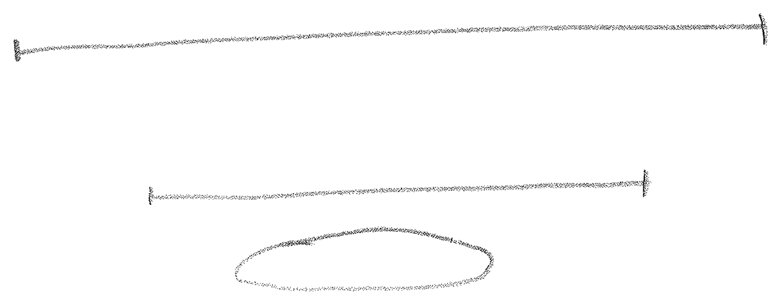
Details: Übung.

$\square$

Skizze:

$\mathcal{A}$:

$\mathcal{B}$:



1.3 Algorithmische Meta-Theoreme auf Grad-beschränkten Klassen von Strukturen

Ein Satz von Seese (1996) besagt, dass für jedes $d \in \mathbb{N}$ und jeden FO-Satz φ das Auswertungsproblem für φ auf Graphen vom Grad $\leq d$ in Linearzeit lösbar ist.
Unter Verwendung der schwachen Hanf-Normalform (Theorem 1.5) können wir dies auf die Logik $FO(\mathbb{P})$ verallgemeinern:

Theorem 1.18

Sei $\mathbb{P} \subseteq \mathcal{P}(\mathbb{Z})$ so dass für jedes $P \in \mathbb{P}$ gilt:
Bei Eingabe einer Zahl $n \in \mathbb{Z}$ kann in Zeit $O(|n|)$ entschieden werden ob $n \in P$ ist.

Sei σ eine Signatur, sei φ ein $FO(\mathbb{P})[\sigma]$-Satz und sei $d \in \mathbb{N}$.
Das Problem

$\boxed{EVAL_{\varphi,d}}$ (Auswertungsproblem für φ auf Strukturen vom Grad $\leq d$)

Eingabe: Eine σ-Struktur $\mathcal{A}$ vom Grad $\leq d$

Frage: Gilt $\mathcal{A} \models \varphi$?

kann in Zeit $O(|\mathcal{A}|)$ gelöst werden.

Beweis:

Gemäß Theorem 1.5 gibt es einen zu φ d-äquivalenten HNF-Satz ψ für FO(P) der Signatur σ. D.h.: ψ ist eine Boolesche Kombination von Hanf-Zählsätzen der Signatur σ für FO(P).

Um Theorem 1.18 zu beweisen genügt es also, einen beliebigen Hanf-Zählsatz χ der Signatur σ für FO(P) zu betrachten und zu zeigen, dass das Problem $EVAL_{\chi,d}$ in Zeit $O(\|\mathcal{A}\|)$ gelöst werden kann.

Sei im Folgenden χ von der Form

$$P\Big(\sum_{\tau \in L} \#(y).\,sph_{\tau,r}(y) \;-\; m\Big)$$

mit $r \in \mathbb{N}$, $m \in \mathbb{Z}$, $L \subseteq \mathcal{L}_r^{\sigma,d}(1)$, $P \in \mathbb{P} \cup \{\mathbb{N}_{\geq 1}\}$.

Sei $\mathcal{L}_r^{\sigma,d}(1) = \tau_1, \dots, \tau_\ell$. Sei $\{1 \leq j_1 < \dots < j_q \leq \ell\}$ s.d. $L = \{\tau_{j_1}, \dots, \tau_{j_q}\}$

Bei Eingabe einer σ-Struktur $\mathcal{A}$ vom Grad $\leq d$ geht der Algorithmus zum Lösen von $EVAL_{\chi,d}$ wie folgt vor:

1) f.a. $i \in \{1, \dots, \ell\}$ setze $anz_i := 0$
2) f.a. $a \in A$ tue Folgendes:
   - 2.1) Nutze Lemma 0.3 (b), um den r-Typ $\tau := (\mathcal{N}_r^{\mathcal{A}}(a), a)$ zu berechnen
   - 2.2) Nutze Lemma 0.4, um bei Eingabe von τ die Zahl $i \in [\ell]$ mit $\tau \cong \tau_i$ zu berechnen
   - 2.3) setze $anz_i := anz_i + 1$

3) Setze $n := \sum_{\nu=1}^{q} anz_{j_\nu} - m$

4) Entscheide, ob $n \in \mathbb{P}$ ist;
falls ja, gib "$\mathcal{A} \models \chi$" aus
sonst gib "$\mathcal{A} \not\models \chi$" aus.

Man sieht leicht, dass der Algorithmus die korrekte Ausgabe liefert.
Zur Laufzeitanalyse beachte, dass l, r, d, m, q Konstanten sind, die nicht von der Eingabe $\mathcal{A}$ abhängen (sondern nur vom Satz χ).

Für jedes feste $a \in A$ ist die für 2.1) und 2.2) verwendete Laufzeit gemäß Lemma 0.3(b) und Lemma 0.4

$$v_d(r)^{O(\|\sigma\|)} \quad \text{und} \quad 2^{v_d(r)^{O(\|\sigma\|)}},$$

wobei $v_d(r) = 1 + d \cdot \sum_{i=0}^{r-1} (d-1)^i$ ist.
Somit wird für jedes feste $a \in A$ in den Zeilen 2.1), 2.2) und 2.3) nur konstant viel Zeit aufgewendet.
Für 2) wird insgesamt also Zeit $O(|A|)$ verwendet.
Die in Zeile 3) berechnete Zahl n ist von der Größe $O(|A|)$, und gemäß unserer Voraussetzung an $\mathbb{P}$ kann Zeile 4) daher in Zeit $O(|A|)$ gelöst werden. Insgesamt benötigt der Algorithmus in den Zeilen 1)–4) also Zeit $O(|A|)$.

$\square$ Theorem 1.18

Unser nächstes Ziel ist, Theorem 1.18 auf geeignete Weise auf Formeln zu erweitern, die freie Variablen besitzen.

Sei $\bar{x} = (x_1, \dots, x_n)$ ein Tupel aus $n \geq 1$ verschiedenen Variablen und sei φ eine $FO(P)[\sigma]$-Formel mit $frei(\varphi) \subseteq \{x_1, \dots, x_n\}$. Wir betrachten folgende Probleme:

$\boxed{COUNT_{\varphi(\bar{x}),d}}$ (Zählproblem für $\varphi(\bar{x})$ auf Strukturen vom Grad $\leq d$)

Eingabe: Eine σ-Struktur $\mathcal{A}$ vom Grad $\leq d$

Ziel: Berechne die Anzahl der Tupel $\bar{a} \in A^n$, für die gilt: $\mathcal{A} \models \varphi[\bar{a}]$.

$\boxed{ENUM_{\varphi(\bar{x}),d}}$ (Aufzählungsproblem für $\varphi(\bar{x})$ auf Strukturen vom Grad $\leq d$)

Eingabe: Eine σ-Struktur $\mathcal{A}$ vom Grad $\leq d$

Ziel: Gib nacheinander (ohne Duplikate) alle Tupel $\bar{a} \in A^n$ mit $\mathcal{A} \models \varphi[\bar{a}]$ aus.

Wir sagen:

„$ENUM_{\varphi(\bar{x}),d}$ kann nach linearer Vorverarbeitung mit konstanter Taktung gelöst werden"

um auszudrücken, dass es einen Algorithmus gibt, der bei Eingabe einer σ-Struktur $\mathcal{A}$ vom Grad $\leq d$ in Zeit $O(|A|)$ eine Datenstruktur aufbaut, die es ermöglicht, nacheinander (ohne Duplikate) alle Tupel in $[\![\varphi(\bar{x})]\!]^{\mathcal{A}} := \{\bar{a} \in A^n : \mathcal{A} \models \varphi[\bar{a}]\}$, gefolgt von der Nachricht "EOE" ("end-of-enumeration") auszugeben, so dass die Wartezeit bis zur ersten Ausgabe und die Wartezeit zwischen zwei aufeinander folgenden Ausgaben $O(1)$ ist.

Unter Verwendung der schwachen Hanf-Normalform können wir Folgendes beweisen:

Theorem 1.19

Sei $\mathbb{P} \subseteq \mathcal{P}(\mathbb{Z})$ so dass für jedes $P \in \mathbb{P}$ gilt: Bei Eingabe einer Zahl $n \in \mathbb{Z}$ kann in Zeit $O(|n|)$ entschieden werden, ob $n \in P$ ist.
Sei σ eine Signatur, sei $n \in \mathbb{N}_{\geq 1}$, sei $\bar{x} = (x_1, \dots, x_n)$ ein Tupel von n verschiedenen Variablen, sei φ eine $FO(\mathbb{P})[\sigma]$-Formel mit $frei(\varphi) \subseteq \{x_1, \dots, x_n\}$. Sei $d \in \mathbb{N}$.

(a) $COUNT_{\varphi(\bar{x}),d}$ kann in Zeit $O(|A|)$ gelöst werden.

(b) $ENUM_{\varphi(\bar{x}),d}$ kann nach linearer Vorverarbeitung mit konstanter Taktung gelöst werden.

Bemerkung: Aussage (b) für FO statt $FO(\mathbb{P})$ wurde von Durand und Grandjean (2007) bewiesen. Die Aussagen (a) und (b) für FO+MOD statt $FO(\mathbb{P})$ wurden von Berkholz, Keppeler, Schweikardt (2017) erzielt. Die Verallgemeinerung von Theorem 1.19 für $FOC(\mathbb{P})$ statt $FO(\mathbb{P})$ wurde von Kuske und Schweikardt (2017) erzielt.

Beweis von Theorem 1.19:

Gemäß Theorem 1.5 gibt es eine zu $\varphi(\bar{x})$ d-äquivalente HNF-Formel $\psi(\bar{x})$ für FO($\mathbb{P}$) der Signatur σ mit $frei(\psi) = \{x_1, \dots, x_n\}$, und für $r := \frac{3^{qr(\varphi)} - 1}{2}$ und $\mathcal{L}_r^{\sigma, d}(n) = \tau_1, \dots, \tau_\ell$ gilt:

ψ ist eine Boolesche Kombination von Hanf-Zählsätzen $\chi_1, \dots, \chi_s$ und von Sphärenformeln $sph_{\tau_i, r}(\bar{x})$ mit $i \in [\ell]$ (für eine geeignete Zahl $s \in \mathbb{N}$).

Bei Eingabe einer σ-Struktur $\mathcal{A}$ vom Grad $\leq d$ nutzen wir für jedes $j \in [s]$ den Algorithmus von Theorem 1.18 um in Zeit $O(|A|)$ zu testen, ob $\mathcal{A} \models \chi_j$.

Insgesamt können wir so in Zeit $O(|A|)$ die Menge

$$J := \{ j \in [s] : \mathcal{A} \models \chi_j \}$$

berechnen.

Sei $\psi_J(\bar{x})$ die Formel, die aus $\psi(\bar{x})$ entsteht, indem jedes Vorkommen eines Satzes χ_j ersetzt wird durch

$$\begin{cases} \text{true}, & \text{falls } j \in J \\ \text{false}, & \text{falls } j \notin J \end{cases}$$

Gemäß Lemma 1.7 können wir dann in Zeit $O(1)$ eine Menge $I \subseteq [\ell]$ berechnen, so dass gilt:

$$\psi_J(\bar{x}) \equiv_d \bigvee_{i \in I} sph_{\tau_i, r}(\bar{x}). \qquad (*)$$

Für die Eingabestruktur $\mathcal{A}$ gilt dann:

$$
\begin{aligned}
[\![\varphi(\bar{x})]\!]^{\mathcal{A}} &= [\![\psi(\bar{x})]\!]^{\mathcal{A}} && (\text{da } \psi \equiv_d \varphi) \\
&= [\![\psi_{\gamma}(\bar{x})]\!]^{\mathcal{A}} && (\text{gemäß Wahl von } \gamma) \\
&= [\![\bigvee_{i \in I} sph_{\tau_i, r}(\bar{x})]\!]^{\mathcal{A}} && (\text{wegen } \circledast) \\
&= \dot{\bigcup_{i \in I}} [\![sph_{\tau_i, r}(\bar{x})]\!]^{\mathcal{A}}.
\end{aligned}
$$

Also ist

$$
|\,[\![\varphi(\bar{x})]\!]^{\mathcal{A}}\,| = \sum_{i \in I} |\,[\![sph_{\tau_i, r}(\bar{x})]\!]^{\mathcal{A}}\,|,
$$

und um den Beweis von Theorem 1.19 (a) und (b) zu beenden genügt es, die Aussagen (a) und (b) für den Spezialfall zu beweisen, dass $\varphi(\bar{x})$ von der Form $sph_{\tau, r}(\bar{x})$ für $r \in \mathbb{N}$ und $\tau \in \mathcal{L}_r^{\text{rid}}(n)$ ist.

Diesen Spezialfall betrachten wir im Folgenden.

Um den Beweis von Theorem 1.19 abzuschließen genügt es also, im Folgenden nur noch den Spezialfall zu betrachten, in dem $\varphi(\bar{x})$ von der Form $\mathrm{sph}_{\tau,r}(\bar{x})$ ist.

Wir betrachten zunächst <u>den Fall, dass τ zusammenhängend ist.</u>

In diesem Fall können wir wie folgt vorgehen:

1) Setze $R := r + (n-1)\cdot(2r+1)$
2) Initialisiere $M := \emptyset$, count := 0
3) Für jedes $a_1 \in A$ tue Folgendes:
   3.1) Berechne $N_R^{\mathcal{A}}(a_1)$
   3.2) Probiere alle Tupel $(a_2,\ldots,a_n) \in \left(N_R^{\mathcal{A}}(a_1)\right)^{n-1}$ durch und entscheide (durch ein brute-force-Verfahren), ob $\left(\mathcal{N}_r^{\mathcal{A}}(a_1,a_2,\ldots,a_n),\, a_1,a_2,\ldots,a_n\right) \cong \tau$ ist, und falls ja, füge das Tupel $(a_1,a_2,\ldots,a_n)$ in M ein und erhöhe count um 1
4) Gib count aus und zähle M auf

Man sieht leicht (Details: Übung!), dass dieser Algorithmus nach Zeit $O(|A|)$ die Zahl $|\llbracket sph_{\tau,r}(\bar{x}) \rrbracket^{\mathcal{A}}|$ ausgibt und mit konstanter Taktung die Menge $M = \llbracket sph_{\tau,r}(\bar{x}) \rrbracket^{\mathcal{A}}$ ausgibt.

Im Folgenden betrachten wir <u>den Fall, dass τ aus $c \geq 2$ Zusammenhangkomponenten besteht</u>.

Sei $\tau = (\mathcal{T}, \bar{t})$ mit $\bar{t} = (t_1, \ldots, t_n) \in T^n$

F.a. $i \in [c]$ sei C_i die Knotenmenge der i-ten Zusammenhangkomponente von $\mathcal{T}$ (d.h. $T = C_1 \dot\cup \cdots \dot\cup C_c$) und sei $\tau_i = \big(\mathcal{T}[C_i], \bar{t}_i \big)$, wobei $\bar{t}_i$ dasjenige Tupel ist, das aus $\bar{t}$ entsteht, indem alle Einträge t_j gelöscht werden, die <u>nicht</u> zu C_i gehören.

OBdA betrachten wir den Fall, dass es Zahlen $\nu_1, \ldots, \nu_c \geq 1$ gibt, s.d. $\bar{t}_1 = (t_1, \ldots, t_{\nu_1})$, $\bar{t}_2 = (t_{\nu_1+1}, \ldots, t_{\nu_1+\nu_2}), \ldots, \bar{t}_i = (t_{\nu_1+\cdots+\nu_{i-1}+1}, \ldots, t_{\nu_1+\cdots+\nu_{i-1}+\nu_i}),$ $\ldots, \bar{t}_c = (t_{\nu_1+\cdots+\nu_{c-1}+1}, \ldots, t_n)$ ist (insbes: $n = \nu_1 + \cdots + \nu_c$ und $\nu_i = |C_i|$ f.a. $i \in [c]$).

Analog sei $\bar{x}_i := (x_{\nu_1+\cdots+\nu_{i-1}+1}, \ldots, x_{\nu_1+\cdots+\nu_{i-1}+\nu_i})$ f.a. $i \in [c]$.

<u>Notation:</u> Für Tupel $\overline{y}=(y_1,\dots,y_k)$ und $\overline{z}=(z_1,\dots,z_\ell)$ schreiben wir $\overline{y}\,\overline{z}$ um das Tupel $(y_1,\dots,y_k,z_1,\dots,z_\ell)$ zu bezeichnen.

Man sieht leicht (Details: Übung), dass für $\varphi(\overline{x}) := \mathrm{sph}_{\tau,r}(\overline{x})$ das Folgende gilt: $(*)_1$

$$[\![\varphi(\overline{x})]\!]^{\mathcal{A}} =$$

$$\Big\{\, \overline{a}_1 \cdots \overline{a}_c \ :\ \begin{array}{l} \overline{a}_i \in [\![\mathrm{sph}_{\tau_i,r}(\overline{x}_i)]\!]^{\mathcal{A}} \text{ f.a. } i\in[c], \\ \text{und f.a. } i,j\in[c] \text{ und alle Einträge} \\ a_{i'} \text{ in } \overline{a}_i \text{ und } a_{j'} \text{ in } \overline{a}_j \text{ gilt:} \\ \quad \mathrm{dist}^{\mathcal{A}}(a_{i'},a_{j'}) > 2r+1 \end{array} \,\Big\}$$

An Stelle der Formel $\varphi(\overline{x})$ betrachten wir im Folgenden die Formel

$$\varphi_c(z_1,\dots,z_c) := \bigwedge_{i=1}^{c} C_i(z_i) \ \wedge \bigwedge_{\substack{i,j\in[c],\\ i\neq j}} \neg E(z_i,z_j)$$

(Stichwort: "rainbow-colored independent set") über der Signatur $\sigma_c := \{E, C_1,\dots,C_c\}$, wobei E ein 2-stelliges und $C_1,\dots,C_c$ 1-stellige Relationssymbole sind.

An Stelle der σ-Struktur $\mathcal{A}$ (vom Grad $\leq d$) betrachten wir die σ_c-Struktur $\mathcal{G} = (V, E^{\mathcal{G}}, C_1^{\mathcal{G}}, \ldots, C_c^{\mathcal{G}})$, die wie folgt definiert ist:

- Für jedes $i \in [c]$ und jedes $\bar{a}_i \in [\![sph_{\tau_i, r}(\bar{x}_i)]\!]^{\mathcal{A}}$ sei $v^i_{\bar{a}_i}$ ein neuer Knoten.

- Für jedes $i \in [c]$ sei
$$C_i^{\mathcal{G}} := \{ v^i_{\bar{a}_i} : \bar{a}_i \in [\![sph_{\tau_i, r}(\bar{x}_i)]\!]^{\mathcal{A}} \}$$

- Sei $V := C_1^{\mathcal{G}} \cup \cdots \cup C_c^{\mathcal{G}}$ und
$$E^{\mathcal{G}} := \{ (v^i_{\bar{a}_i}, v^j_{\bar{a}_j}) : i,j \in [c],\ i \neq j,\ \text{ex } a_i' \text{ in } \bar{a}_i \text{ und } a_j' \text{ in } \bar{a}_j \text{ s.d. } dist^{\mathcal{A}}(a_i', a_j') \leq 2r+1 \}$$

Behauptung 1: Die Abbildung $\pi : C_1^{\mathcal{G}} \times \cdots \times C_c^{\mathcal{G}} \to A^n$ mit $\pi(v^1_{\bar{a}_1}, \ldots, v^c_{\bar{a}_c}) := \bar{a}_1 \ldots \bar{a}_c$ f.a. $v^1_{\bar{a}_1} \in C_1^{\mathcal{G}}, \ldots, v^c_{\bar{a}_c} \in C_c^{\mathcal{G}}$ ist injektiv; und π eingeschränkt auf $[\![\psi_c(z_1, \ldots, z_c)]\!]^{\mathcal{G}}$ ist eine Bijektion zwischen $[\![\psi_c(z_1, \ldots, z_c)]\!]^{\mathcal{G}}$ und $[\![\psi(\bar{x})]\!]^{\mathcal{A}}$.

<u>Beweis:</u> Übung.

<u>Behauptung 2</u>: Bei Eingabe von $\mathcal{A}$ kann $\mathcal{G}$ in Zeit $O(|A|)$ erzeugt werden, und der Grad d' von $\mathcal{G}$ hängt von d, τ, r, n aber <u>nicht</u> von $|A|$ ab.

<u>Beweisidee:</u> Für jedes $i \in [c]$ berechne

$[\![sph_{\tau_i, r}(\bar{x}_i)]\!]^{\mathcal{A}}$ (das geht in Zeit $O(|A|)$, da τ_i zusammenhängend ist).

Für jedes $\bar{a}_i \in [\![sph_{\tau_i, r}(\bar{x}_i)]\!]^{\mathcal{A}}$ erzeuge einen neuen Knoten $v^i_{\bar{a}_i}$ und füge ihn in $C_i^{\mathcal{G}}$ ein.

So können wir $C_1^{\mathcal{G}}, \dots, C_c^{\mathcal{G}}$ und $V := C_1^{\mathcal{G}} \cup \dots \cup C_c^{\mathcal{G}}$ in Zeit $O(|A|)$ erzeugen.

Um $E^{\mathcal{G}}$ in Zeit $O(|A|)$ zu erzeugen, nutze, dass für alle $i, j \in [c]$ mit $i \neq j$ und alle $\bar{a}_i \in [\![sph_{\tau_i, r}(\bar{x}_i)]\!]^{\mathcal{A}}$ und $\bar{a}_j \in [\![sph_{\tau_j, r}(\bar{x}_j)]\!]^{\mathcal{A}}$ gilt:

- jeder Eintrag $a_{i'}$ in $\bar{a}_i$ hat in $\mathcal{A}$ den Abstand $\leq (2r+1) \cdot v_i$ zum ersten Eintrag $a_{i,1}$ in $\bar{a}_i$ (wobei v_i die Länge des Tupels $\bar{a}_i$ ist) (und analog für $a_{j'}, \bar{a}_j, v_j, a_{j,1}$)

- Falls $(v^i_{\bar{a}_i}, v^j_{\bar{a}_j}) \in E^{\mathcal{G}}$, dann gilt insbes.:

  Es gibt ein $a_{i'}$ in $\bar{a}_i$ und ein $a_{j'}$ in $\bar{a}_j$ s.d. $dist^{\mathcal{A}}(a_{i'}, a_{j'}) \leq 2r+1$ ist —

  insbes ist $dist^{\mathcal{A}}(a_{j'}, a_{i,1}) \leq 2r+1 + (2r+1) \cdot v_i$,

  und jeder Eintrag in $\bar{a}_j$ ist in

  $N_R^{\mathcal{A}}(a_{i,1})$ für $R := (2r+1) + (2r+1) \cdot v_i + (2r+1) \cdot v_j$.

Nutze dies, um in Zeit $O(|A|)$ die Kantenmenge $E^{\mathcal{G}}$ zu erzeugen; und zeige, dass der Grad d' von $\mathcal{G}$ nur von d, τ, r, n abhängt, aber nicht von $|A|$. Details: Übung.

$\square_{\text{Beh. 2}}$

Aus Behauptung 1 und Behauptung 2 ergibt sich, dass wir, um den Beweis von Theorem 1.19 abzuschließen, nur noch den folgenden Fall betrachten müssen:

Sei $d' \in \mathbb{N}$, $c \in \mathbb{N}_{\geq 1}$, $\sigma_c := \{E, C_1, \dots, C_c\}$;

$\overline{z} = (z_1, \dots, z_c)$ seien c verschiedene Variablen,

$$\varphi_c(\overline{z}) := \bigwedge_{i=1}^{c} C_i(z_i) \wedge \bigwedge_{\substack{i,j \in [c], \\ i \neq j}} \neg E(z_i, z_j)$$

("rainbow-colored independent set").

<u>Zu zeigen</u>: Bei Eingabe einer σ_c-Struktur $\mathcal{G} = (V, E^{\mathcal{G}}, C_1^{\mathcal{G}}, \dots, C_c^{\mathcal{G}})$ vom Grad $\leq d'$ kann $\text{COUNT}_{\varphi_c(\overline{z}), d'}$ in Zeit $O(|V|)$ gelöst werden, und $\text{ENUM}_{\varphi_c(\overline{z}), d'}$ kann nach $O(|V|)$ Vorverarbeitungszeit mit konstanter Taktung gelöst werden.

<u>Wir stellen nun die Lösung für $\boxed{\text{COUNT}_{\varphi_c(\overline{z}), d'}}$ vor:</u>

Für jede Eingabe $\mathcal{G}$ sieht $[\![\varphi_c(\overline{z})]\!]^{\mathcal{G}}$ wie folgt aus, wobei $\theta_{j,j'}(\overline{z}) := \left(\bigwedge_{i=1}^{c} C_i(z_i)\right) \wedge E(z_j, z_{j'})$ sei:

$$[\![\varphi_c(\bar z)]\!]^{\mathcal{G}} = \left(C_1^{\mathcal{G}} \times \cdots \times C_c^{\mathcal{G}} \right) \setminus \left(\bigcup_{\substack{j,j' \in [c], \\ j \neq j'}} [\![\theta_{j,j'}(\bar z)]\!]^{\mathcal{G}} \right),$$

und es gilt:

$$\left| [\![\varphi_c(\bar z)]\!]^{\mathcal{G}} \right| = \underbrace{\left(\prod_{i=1}^{c} \left| C_i^{\mathcal{G}} \right| \right)}_{=: N_1} - \underbrace{\left| \bigcup_{j \neq j'} [\![\theta_{j,j'}(\bar z)]\!]^{\mathcal{G}} \right|}_{=: N_2}$$

<u>Klar:</u> N_1 kann in Zeit $O(|V|)$ berechnet werden.

Um N_2 zu berechnen, nutzen wir das
Prinzip der Inklusion und Exklusion (kurz: P.I.E.), das
für Mengen $M_1, \ldots, M_n$ besagt:

$$|M_1 \cup \cdots \cup M_n| = \sum_{\emptyset \neq K \subseteq \{1,\ldots,n\}} (-1)^{|K|+1} \cdot \left| \bigcap_{k \in K} M_k \right|$$

(Übung: Zeigen Sie, dass dies für alle $n \geq 1$ und alle
endlichen Mengen $M_1, \ldots, M_n$ korrekt ist).

Für $J := \{ (j,j') : j,j' \in [c], \ j \neq j' \}$

gilt:

$$\left| \bigcup_{(j,j')\in J} [\![\theta_{j,j'}(\bar{z})]\!]^{\mathcal{G}} \right| \overset{\text{P.I.E.}}{=}$$

$$\sum_{\emptyset \neq K \subseteq J} (-1)^{|K|+1} \cdot \left| \bigcap_{(j,j')\in K} [\![\theta_{j,j'}(\bar{z})]\!]^{\mathcal{G}} \right| =$$

$$\sum_{\emptyset \neq K \subseteq J} (-1)^{|K|+1} \cdot \left| [\![\alpha_K(\bar{z})]\!]^{\mathcal{G}} \right| ,$$ wobei

$$\alpha_K(\bar{z}) := \left(\bigwedge_{i=1}^{c} C_i(z_i) \right) \wedge \left(\bigwedge_{(j,j')\in K} E(z_j, z_{j'}) \right)$$ sei.

<u>Beachte:</u> $|J|$ ist konstant ($< c^2$). Somit genügt es zu zeigen, dass für jedes $K \subseteq J$ mit $K \neq \emptyset$ die Zahl $\left| [\![\alpha_K(\bar{z})]\!]^{\mathcal{G}} \right|$ in Zeit $O(|V|)$ berechnet werden kann.

Betrachte im Folgenden also ein beliebiges $K \subseteq J$ mit $K \neq \emptyset$.

Sei $H = (W, K)$ der gerichtete Graph mit Knotenmenge $W := [c]$ und Kantenmenge K.

Sei s die Anzahl der Zusammenhangskomponenten des Gaifman-Graphen von H. Für jedes $i \in [s]$ sei

W_i die Knotenmenge der i-ten Zusammenhangskomponente, sei $H_i := H[W_i]$, und schreibe K_i, um die Kantenmenge von H_i zu bezeichnen.

Für jedes $i \in [s]$ sei $v_i := |W_i|$.

OBdA sei $W_1 = \{1, \dots, v_1\}$, $W_2 = \{v_1+1, \dots, v_2\}$, $\dots$,
$W_\ell = \{v_1 + \dots + v_{\ell-1}+1, \dots, v_1 + \dots + v_{\ell-1} + v_\ell\}$ f.a. $\ell \in [s]$.

Analog sei $\overline{z_1} := (z_1, \dots, z_{v_1})$, $\overline{z_2} := (z_{v_1+1}, \dots, z_{v_2})$ und allgemein $\overline{z_\ell} := (z_{v_1+\dots+v_{\ell-1}+1}, \dots, z_{v_1+\dots+v_{\ell-1}+v_\ell})$ f.a. $\ell \in [s]$.

Dann ist $\overline{z} = (z_1, \dots, z_c) = \overline{z_1} \cdots \overline{z_s}$, und es gilt

$$[\![\alpha_K(\overline{z})]\!]^g \underset{\text{Warum? – Übung!}}{=} [\![\alpha'_{K_1}(\overline{z_1})]\!]^g \times \dots \times [\![\alpha'_{K_s}(\overline{z_s})]\!]^g,$$

wobei f.a. $\ell \in [s]$

$$\alpha'_{K_\ell}(\overline{z_\ell}) := \left(\bigwedge_{i \in W_\ell} C_i(z_i) \right) \wedge \left(\bigwedge_{(j,j') \in K_\ell} E(z_j, z_{j'}) \right).$$

Es bleibt also nur noch zu zeigen, dass für jedes $\ell \in [s]$ die Zahl $\left| [\![\alpha'_{K_\ell}(\overline{z_\ell})]\!]^g \right|$ in Zeit $O(|V|)$ berechnet werden kann.

Betrachte dazu ein beliebiges $\ell \in [s]$.

Beachte, dass $H_\ell = (W_\ell, K_\ell)$ zusammenhängend ist.

Daher gilt für jedes Tupel

$$(a_1, \dots, a_{v_\ell}) = \overline{a}_\ell \in [\![\alpha'_{K_\ell}(\overline{z}_\ell)]\!]^{\mathcal{G}},$$

dass $a_2, \dots, a_{v_\ell} \in N_R^{\mathcal{G}}(a_1)$ für $R := v_\ell$ ist

(Begründung: Übung!).

Daher können wir zur Berechnung von $[\![\alpha'_{K_\ell}(\overline{z}_\ell)]\!]^{\mathcal{G}}$ wie folgt vorgehen:

1.) Für jedes $a_1 \in V$ berechne $N_R^{\mathcal{G}}(a_1)$ (diese Menge enthält höchstens $(d')^{R+1}$ Knoten, da $\mathcal{G}$ Grad $\leq d'$ hat) und berechne die Menge

$$M_{a_1} := \{ (a_2, \dots, a_{v_\ell}) \in \left(N_R^{\mathcal{G}}(a_1)\right)^{v_\ell - 1} :$$
$$\mathcal{G} \models \alpha'_{K_\ell}[a_1, a_2, \dots, a_{v_\ell}] \}$$

(für jedes feste $a_1 \in V$ geht das in konstanter Zeit) und berechne die Zahl $m_{a_1} := |M_{a_1}|$.

2) Setze $m_{K_\ell} := \sum_{a_1 \in V} m_{a_1}$

All dies geht in Zeit $O(|V|)$, und es gilt:
$M := \bigcup_{a_1 \in V} M_{a_1} = [\![\alpha'_{K_\ell}(\overline{z}_\ell)]\!]^{\mathcal{G}}$ und $m_{K_\ell} = |M| = |[\![\alpha'_{K_\ell}(\overline{z}_\ell)]\!]^{\mathcal{G}}|$
(Begründung: Übung!)

Insgesamt erhalten wir dadurch einen Algorithmus, der COUNT$_{\varphi_c(\bar{z}),d'}$ in Zeit $O(|V|)$ löst.

Um den Beweis von Theorem 1.19 abzuschließen, müssen wir nur noch einen Algorithmus finden, der $\boxed{\text{ENUM}_{\varphi_c(\bar{z}),d'}}$ nach $O(|V|)$ Vorverarbeitungszeit mit konstanter Taktung löst.

Zur Erinnerung: $\varphi_c(\bar{z}) = \bigwedge_{i=1}^{c} C_i(z_i) \wedge \bigwedge_{j \neq j'} \neg E(z_j, z_{j'})$.

Als Eingabe erhalten wir eine σ_c-Struktur

$$\mathcal{G} = (V, E^{\mathcal{G}}, C_1^{\mathcal{G}}, \ldots, C_c^{\mathcal{G}})$$ vom Grad $\leq d'$.

1. Idee zur Aufzählung von $[\![\varphi_c(\bar{z})]\!]^{\mathcal{G}}$:

```
for all a_1 ∈ C_1^G do
   for all a_2 ∈ C_2^G do
      if (a_1,a_2) ∉ E^G and (a_2,a_1) ∉ E^G
      then for all a_3 ∈ C_3^G do
         if (a_i,a_3) ∉ E^G and (a_3,a_i) ∉ E^G f.a. i∈{1,2}
         then for all a_4 ∈ C_4^G do
            ⋱
               for all a_c ∈ C_c^G do
                  if (a_i,a_c) ∉ E^G and (a_c,a_i) ∉ E^G f.a. i∈[c-1]
                  then output (a_1,a_2,...,a_c).
```

Gut: Dieser Algorithmus gibt genau diejenigen Tupel aus, die zu $[\![\varphi_c(\bar{z})]\!]^{\mathcal{G}}$ gehören — und zwar jedes davon genau einmal.

Schlecht: Wir haben keine Garantie darüber, dass zwischen der Ausgabe von je 2 Tupeln nur konstant viel Zeit (unabhängig von $|V|$) vergeht (Warum? — Übung!)

Betrachte z.B. den Fall $d' = 3$ und die Eingabe $\mathcal{G} = (V, E^{\mathcal{G}}, C_1^{\mathcal{G}}, C_2^{\mathcal{G}}, C_3^{\mathcal{G}}, C_4^{\mathcal{G}})$ (also $c = 4$) mit $C_1^{\mathcal{G}} = \{0, 1\}$, $C_2^{\mathcal{G}} = \{2, 3, 4, \ldots, n+1\}$, $C_3^{\mathcal{G}} = \{n+2, n+3, \ldots, 2n+1\}$, $C_5^{\mathcal{G}} = \{2n+2, 2n+3\}$ und $E^{\mathcal{G}} = \{(0, 2n+2), (0, 2n+3)\}$.

2. bessere Idee zur Aufzählung von $[\![\varphi_c(\bar{z})]\!]^{\mathcal{G}}$:

Berechne $I := \{ i \in [c] : |C_i^{\mathcal{G}}| \leq c \cdot d' \}$.

Die $C_i^{\mathcal{G}}$ mit $i \in I$ nennen wir "die kleinen Farben", während wir die $C_i^{\mathcal{G}}$ mit $i \in [c] \setminus I$ "die großen Farben" nennen.

Sei $k := |I|$ die Anzahl der kleinen Farben in $\mathcal{G}$.

Wir betrachten im Folgenden o.B.d.A. den Fall, dass $I = \{ i \in [c] : i \leq k \}$ ist

(d.h.: $C_1^g, \dots, C_k^g$ sind die kleinen Farben, und $C_{k+1}^g, \dots, C_c^g$ sind die großen Farben;

beachte: Es könnte $k=0$ (also $I=\emptyset$) oder auch $k=c$ (also $I=[c]$) sein).

Der Begriff "große Farbe" ist so gewählt, dass Folgendes gilt:

<u>Behauptung</u> (*): F.a. $i \in [c-1]$ mit $i \geq k$ und alle Tupel $(a_1, \dots, a_i) \in C_1^g \times \dots \times C_i^g$ gilt:

es gibt (mind.) ein $a_{i+1} \in C_{i+1}^g$ s.d. f.a. $j \in [i]$ gilt: $(a_i, a_j) \notin E^g$ und $(a_j, a_i) \notin E^g$.

<u>Beweis</u>: Wir wissen, dass (der Gaifman-Graph von) $\mathcal{G}$ den Grad $\leq d'$ hat. Daher gibt es für jedes $j \in [i]$ höchstens d' verschiedene Knoten b mit $(a_j, b) \in E^g$ oder $(b, a_j) \in E^g$.

Wegen $i \geq k$ ist C_{i+1}^g eine große Farbe, d.h. $|C_{i+1}^g| \geq c \cdot d' + 1 \geq i \cdot d' + 1$. Somit gibt es in C_{i+1}^g mind. einen Knoten, der zu keinem der Knoten $a_1, \dots, a_i$ benachbart ist

$\square$ Beh (*).

In der Vorverarbeitungsphase berechnen wir brute-force die Menge

$$M := \{ (a_1,\dots,a_k) \in C_1^g \times \dots \times C_k^g :$$
$$\text{f.a. } j,j' \in [k] \text{ mit } j \neq j' \text{ gilt } (a_j, a_{j'}) \notin E^g \}.$$

Da $C_1^g, \dots, C_k^g$ kleine ~~Farben~~ sind, geht dies in Zeit, die nur von c und d', nicht aber von $|V|$ abhängt.

Zum Aufzählen aller Tupel in $[\![\varphi_c(\bar{z})]\!]^g$ nutzen wir nun den folgenden Algorithmus:

for all $(a_1,\dots,a_k) \in M$ do
  for all $a_{k+1} \in C_{k+1}^g$ do
    if $(a_i, a_{k+1}) \notin E^g$ and $(a_{k+1}, a_i) \notin E^g$ f.a. $i \in [k]$
    then for all $a_{k+2} \in C_{k+2}^g$ do
        if $(a_i, a_{k+2}) \notin E^g$ and $(a_{k+2}, a_i) \notin E^g$ f.a. $i \in [k+1]$
        then for all $a_{k+3} \in C_{k+3}^g$ do
          ⋱
            for all $a_c \in C_c^g$ do
              if $(a_i, a_c) \notin E^g$ and $(a_c, a_i) \notin E^g$ f.a. $i \in [c-1]$
              then output $(a_1, a_2, \dots, a_c)$

Deutlich hübscher lässt sich dieser Algorithmus rekursiv formulieren, nämlich wie folgt:

for all $(a_1,\dots,a_k) \in M$ do
    ENUM $(a_1,\dots,a_k)$

mit der rekursiv definierten Funktion

<u>function ENUM $(a_1,\dots,a_i)$</u>

if $i=c$
then output $(a_1,\dots,a_c)$
else for all $a_{i+1} \in C_{i+1}^{\mathcal{G}}$ do
        if $(a_{i+1},a_j) \notin E^{\mathcal{G}}$ and $(a_j,a_{i+1}) \notin E^{\mathcal{G}}$ f.a. $j \in [i]$
        then ENUM$(a_1,\dots,a_i,a_{i+1})$

<u>Behauptung</u> (**): F.a. $i \in \{k,k+1,\dots,c\}$ und alle Tupel $(a_1,\dots,a_i) \in C_1^{\mathcal{G}} \times \dots \times C_i^{\mathcal{G}}$ mit $(a_j,a_{j'}) \notin E^{\mathcal{G}}$ f.a. $j,j' \in [i]$ mit $j \neq j'$ gilt: Beim Aufruf von ENUM$(a_1,\dots,a_i)$ werden genau diejenigen Tupel $(b_1,\dots,b_c)$ ausgegeben, für die gilt: $(b_1,\dots,b_i)=(a_1,\dots,a_i)$ und $(b_1,\dots,b_c) \in [\![\varphi_c(\bar{z})]\!]^{\mathcal{G}}$. Jedes dieser Tupel wird nur einmal ausgegeben. Vor der Ausgabe des ersten Tupels, zwischen der Ausgabe von zwei aufeinander folgenden Tupeln und nach der Ausgabe des letzten Tupels vergehen jeweils höchstens $O\big((c-i)\cdot(cd'+1)\big)$ Berechnungsschritte.

Beweis: Per Induktion nach i.

Induktionsanfang: $i = c$ ✓

Induktionsschritt: $i+1 \to i$

Sei $(a_1, \dots, a_i) \in C_1^{\mathcal{G}} \times \dots \times C_i^{\mathcal{G}}$ mit $(a_j, a_{j'}) \notin E^{\mathcal{G}}$ f.a. $j, j' \in [i]$ mit $j \neq j'$.

Beim Aufruf von $\mathrm{ENUM}(a_1, \dots, a_i)$ wird nach höchstens $i \cdot d' + 1$ Schritten ein $a_{i+1} \in C_{i+1}^{\mathcal{G}}$ gefunden, s.d. $(a_{i+1}, a_j) \notin E^{\mathcal{G}}$ und $(a_j, a_{i+1}) \notin E^{\mathcal{G}}$ f.a. $j \in [i]$ gilt (denn: $\mathcal{G}$ hat Grad $\leq d'$, und $C_{i+1}^{\mathcal{G}}$ ist eine große Farbe, dh $|C_{i+1}^{\mathcal{G}}| \geq cd'+1 \geq id'+1$).

D.h: nach höchstens $id'+1 \leq c \cdot d' + 1$ Schritten wird ein rekursiver Aufruf $\mathrm{ENUM}(a_1, \dots, a_i, a_{i+1})$ gestartet. Gemäß Induktionsannahme zählt dieser mit Taktung $O\big((c-(i+1))(cd'+1)\big)$ alle Tupel $(b_1, \dots, b_c) \in [\![\psi_c(\bar{z})]\!]^{\mathcal{G}}$ auf, für die gilt: $(b_1, \dots, b_i, b_{i+1}) = (a_1, \dots, a_i, a_{i+1})$.

Danach wird bei der weiteren Berechnung von $\mathrm{ENUM}(a_1, \dots, a_i)$ wieder nach max $id'+1 \leq cd'+1$ Schritten das nächste passende $a_{i+1} \in C_{i+1}^{\mathcal{G}}$ gefunden, für das wiederum ein rekursiver Aufruf $\mathrm{ENUM}(a_1, \dots, a_i, a_{i+1})$ gestartet wird usw., bis alle Elemente aus $C_{i+1}^{\mathcal{G}}$ abgearbeitet wurden.

Insgesamt erhält man, dass ENUM$(a_1,\dots,a_i)$ mit Taktung $O\big(cd'+1 + (c-(i+1))(cd'+1)\big)$

$= O\big((c-i)(cd'+1)\big)$ genau diejenigen Tupel $(b_1,\dots,b_c) \in [\![\varphi_c(\bar{z})]\!]^{\mathcal{G}}$ ausgibt, für die gilt: $(b_1,\dots,b_i) = (a_1,\dots,a_i)$.

$\square$ Beh (**).

Die Aussage von Beh (**) in Kombination mit der Wahl der Menge M liefert, dass unser Algorithmus mit Taktung $O(c^2 \cdot d')$ arbeitet und die Menge $[\![\varphi_c(\bar{z})]\!]^{\mathcal{G}}$ ausgibt.

Dies beendet den Beweis von Theorem 1.19

$\square$ Thm 1.19

Zum Abschluss von Kapitel 1 betrachten wir folgende Übungsaufgabe.

<u>Übungsaufgabe:</u>

Für einen gerichteten Graphen $G = (V, E)$ sei $\|G\| := |V| + |E|$.

Finden Sie einen Algorithmus, der bei Eingabe eines beliebigen gerichteten Graphen $G = (V, E)$ (von beliebigem Grad) nach $O(\|G\| \cdot \log \|G\|)$ Vorverarbeitungszeit mit konstanter Taktung die Menge $V^2 \setminus E$ $\left(= [\![\neg E(z_1, z_2)]\!]^{\mathcal{G}} \right)$ ausgibt.