Fall 3: φ ist von der Form $\neg\varphi_1$ oder von der Form $(\varphi_1 * \varphi_2)$ mit $* \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$, wobei φ_1 und φ_2 $FO[\sigma_\Sigma]$-Sätze der Quantorentiefe $m+1$ sind:

Analog zu Fall 3, Fall 4 und Fall 5 des Induktionsanfangs.

Insgesamt ist damit der Beweis von Satz 5.8 abgeschlossen.

$\square$

Bemerkung 5.9:

Aus dem Satz von McNaughton und Papert und den Ergebnissen aus Kapitel 3 lässt sich leicht folgern, dass es reguläre Sprachen gibt, die nicht sternfrei regulär sind.

Details: Übung.

Kapitel 6: Die Berechnungskomplexität der Logik erster Stufe

Alle in diesem Kapitel betrachteten Signaturen σ sind endlich und Funktionen-frei.

Definition 6.1:

(a) Die Länge (bzw. Größe) $\|\varphi\|$ einer $FO[\sigma]$-Formel φ ist die Länge von φ, betrachtet als Wort über dem Alphabet A_σ (siehe Definition 1.13, 1.14 und 1.16).

(b) Die Größe $\|\mathfrak{A}\|$ einer endlichen σ-Struktur $\mathfrak{A}$ ist definiert als

$$\|\mathfrak{A}\| := |\sigma| + |A| + \sum_{R\in\sigma} |R^{\mathfrak{A}}| \cdot ar(R).$$

Definition 6.2

(a) Ist σ eine endliche Signatur, so bezeichnet FIN_σ die Klasse aller endlichen σ-Strukturen.

(b) FIN bezeichnet die Klasse aller endlichen σ-Strukturen, für alle endlichen Signaturen σ.

6.1 Die Auswertung von Formeln in endlichen Strukturen

Sei σ eine endliche relationale Signatur.

Als Eingabe für Algorithmen repräsentieren wir eine endliche σ-Struktur wie folgt:

- Die Elemente des Universums von $\mathfrak{A}$ repräsentieren wir durch Objekte eines geeigneten Datentyps Θ (etwa <u>integer</u> oder <u>string</u>).

  Das Universum A von $\mathfrak{A}$ wird dann als Liste von Objekten vom Typ Θ repräsentiert.

- Ein r-Tupel $(a_1, \dots, a_r) \in A^r$ repräsentieren wir als Liste von Objekten vom Typ Θ.

  Für jedes $R \in \sigma$ repräsentieren wir die Relation $R^{\mathfrak{A}}$ dann als Liste aller r-Tupel $(a_1, \dots, a_r) \in R^{\mathfrak{A}}$ (für $r := ar(R)$).

Man beachte, dass die Repräsentation von $\mathfrak{A}$ dann die Größe

$$\approx \quad \|\mathfrak{A}\| = |\sigma| + |A| + \sum_{R \in \sigma} |R^{\mathfrak{A}}| \cdot ar(R)$$

hat

Unsere bisherige Sichtweise auf die Semantik der Logik erster Stufe war, dass FO[σ]-Formeln in σ-Interpretationen ausgewertet werden, so dass für jede FO[σ]-Formel φ und jede zu φ passende σ-Interpretation $\mathcal{I}$ gilt:

entweder $\mathcal{I} \models \varphi$ oder $\mathcal{I} \not\models \varphi$.

Eine alternative Sichtweise ist, dass Formeln Relationen in Strukturen definieren (vgl. Definition 4.12)

<u>Definition 6.3</u>

Für alle σ-Strukturen $\mathfrak{A}$, alle FO[σ]-Formeln φ und alle Tupel $(x_1, \dots, x_k) \in \mathrm{Var}^k$ mit $\mathrm{frei}(\varphi) \subseteq \{x_1, \dots, x_k\}$ ist

$$\varphi(\mathfrak{A}) := \{ (a_1, \dots, a_k) \in A^k : \mathfrak{A} \models \varphi[a_1, \dots, a_k] \}$$

die von φ in $\mathfrak{A}$ definierte k-stellige Relation.

<u>Vorsicht</u>: Die Relation $\varphi(\mathfrak{A})$ hängt nicht nur von der Formel φ ab, sondern auch von dem Tupel $(x_1, \dots, x_k) \in \mathrm{Var}^k$. Wenn wir die Notation $\varphi(\mathfrak{A})$ verwenden, müssen wir daher immer vorher angeben, auf welche Variablen(reihenfolge) sie sich bezieht.

Zur Vereinfachung betrachten wir im Folgenden nur $FO[\sigma]$-Formeln, in denen keins der Symbole $\wedge, \rightarrow, \leftrightarrow, \forall$ vorkommt (gemäß Beobachtung 2.2 ist dies keine wirkliche Einschränkung).

<u>Beobachtung 6.4:</u>

Ist σ eine relationale Signatur und $\mathfrak{A}$ eine σ-Struktur, so können wir für $FO[\sigma]$-Formeln φ und Variablentupel $(x_1, \ldots, x_k) \in Var^k$ mit $frei(\varphi) \subseteq \{x_1, \ldots, x_k\}$ die Relation $\varphi(\mathfrak{A}) \subseteq A^k$ rekursiv wie folgt beschreiben:

- Falls φ von der Form $x_{i_1} = x_{i_2}$, für $i_1, i_2 \in \{1, \ldots, k\}$ ist, so ist

$$\varphi(\mathfrak{A}) = \{ (a_1, \ldots, a_k) \in A^k : a_{i_1} = a_{i_2} \}$$

- Falls φ von der Form $R x_{i_1} \cdots x_{i_r}$, für $R \in \sigma$, $r := ar(R)$ und $i_1, \ldots, i_r \in \{1, \ldots, k\}$ ist, so ist

$$\varphi(\mathfrak{A}) = \{ (a_1, \ldots, a_k) \in A^k : (a_{i_1}, \ldots, a_{i_r}) \in R^{\mathfrak{A}} \}$$

- Falls φ von der Form $\neg\varphi_1$ ist, so ist

$$\varphi(\mathfrak{A}) = A^k \setminus \varphi_1(\mathfrak{A})$$

- Falls φ von der Form $(\varphi_1 \vee \varphi_2)$ ist, so ist

$$\varphi(\mathfrak{A}) = \varphi_1(\mathfrak{A}) \cup \varphi_2(\mathfrak{A})$$

- Falls φ von der Form $\exists x_{k+1}\, \varphi_1(x_1,\dots,x_k,x_{k+1})$ ist, so ist

$$\varphi(\mathfrak{A}) = \{ (a_1,\dots,a_k) \in A^k : \text{ es gibt ein } a_{k+1} \in A \text{ s.d. } (a_1,\dots,a_k,a_{k+1}) \in \varphi_1(\mathfrak{A}) \}.$$

Definition 6.5

Das Auswertungsproblem der Logik erster Stufe auf der Klasse aller endlichen relationalen Strukturen ist wie folgt definiert:

Auswertungsproblem für FO auf FIN

Eingabe: Eine endliche σ-Struktur $\mathfrak{A}$, wobei σ eine endliche relationale Signatur ist, und eine FO[σ]-Formel φ

Ziel: Berechne $\varphi(\mathfrak{A})$ (bzgl. dem Variablentupel $(x_1,\dots,x_k)$ mit $\text{frei}(\varphi) = \{x_1,\dots,x_k\}$ und $k = |\text{frei}(\varphi)|$)

Beobachtung 6.4 führt unmittelbar zu einem rekursiven Algorithmus, der das Auswertungsproblem für FO auf FIN löst.

Die genaue Laufzeit des Algorithmus hängt von dem folgenden Parameter ab:

<u>Definition 6.6</u>

Die <u>Breite</u> (engl. <u>width</u>) $br(\varphi)$ einer $FO[\sigma]$-Formel φ ist die maximale Anzahl freier Variablen in Subformeln von φ, d.h.

$$br(\varphi) = \max\{\,|frei(\psi)| : \psi \in sub(\varphi)\,\}$$

(vgl. Definition 1.19: $sub(\varphi)$ ist die Menge aller Subformeln von φ — beachte: $\varphi \in sub(\varphi)$).

<u>Satz 6.7</u>

Es gibt einen Algorithmus, der das Auswertungsproblem für FO auf FIN bei Eingabe einer Struktur $\mathfrak{A}$ und einer Formel φ in Zeit

$$O\left(\|\varphi\| \cdot \|\mathfrak{A}\|^{br(\varphi)} \cdot br(\varphi) \quad + \|\varphi\| + \|\mathfrak{A}\|\right)$$

löst.

Beweis:

Durch geschickte Implementierung des rekursiven Algorithmus, der sich aus Beobachtung 6.4 ergibt (Beachte: $|sub(\varphi)| \leq \|\varphi\|$).

Details: Übung $\square$

Beobachtung 6.8

(a) $br(\varphi) \leq \|\varphi\|$

(b) $br(\varphi) \leq |\{x \in var : x \text{ kommt in } \varphi \text{ vor}\}|$

(c) Für jede $FO[\sigma]$-Formel φ gibt es eine zu φ äquivalente $FO[\sigma]$-Formel $\tilde{\varphi}$ mit $frei(\tilde{\varphi}) = frei(\varphi)$, in der höchstens $br(\varphi)$ viele verschiedene Variablen vorkommen.

Beweis: Übung

Satz 6.9

Das Auswertungsproblem für FO auf FIN ist vollständig für die Komplexitätsklasse PSPACE aller auf polynomiellem Platz lösbaren Probleme

(hier ohne Beweis) (Ein Beweis dieses Satzes wird in der Vorlesung "Logik und Komplexität" gegeben)

6.2 Der Satz von Trakhtenbrot

Definition 6.10: Sei σ eine endliche Signatur.

Das Erfüllbarkeitsproblem für $FO[\sigma]$ auf FIN ist das wie folgt definierte Berechnungsproblem:

Eingabe: Ein $FO[\sigma]$-Satz φ

Frage: Gibt es eine endliche σ-Struktur $\mathfrak{A}$ mit $\mathfrak{A} \models \varphi$?

Zur Erinnerung:

Ein Berechnungsproblem, bei dem die Ausgabe aus einer der beiden Antworten "ja" bzw "nein" besteht, heißt

- entscheidbar, falls es einen Algorithmus gibt, der bei jeder für das Problem zulässigen Eingabe nach endlich vielen Schritten anhält und die korrekte Antwort "ja" bzw "nein" ausgibt
- semi-entscheidbar, falls es einen Algorithmus gibt, der bei jeder für das Problem zulässigen Eingabe E folgendes Verhalten aufweist:

- ist "ja" die korrekte Antwort bei Eingabe $\mathcal{E}$, so hält der Algorithmus bei Eingabe $\mathcal{E}$ nach endlich vielen Schritten an und gibt "ja" aus
- ist "nein" die korrekte Antwort bei Eingabe $\mathcal{E}$, so hält der Algorithmus bei Eingabe $\mathcal{E}$ nie an.

Unter Verwendung des Satzes von Gaifman sieht man leicht, dass folgendes gilt:

<u>Satz 6.11</u>

Für jede endliche relationale Signatur σ, die ausschließlich aus Relationssymbolen der Stelligkeit 1 besteht, ist das Erfüllbarkeitsproblem für $FO[\sigma]$ auf FIN entscheidbar.

<u>Beweis:</u> Übung.

Der Satz von Trakhtenbrot besagt, dass Satz 6.11 nicht für Signaturen gilt, die Relationssymbole der Stelligkeit ≥ 2 enthalten.