Logik und Komplexität

Vorlesung im Wintersemester 2025/26

Prof. Dr. Nicole Schweikardt

Lehrstuhl Logik in der Informatik
Institut für Informatik
Humboldt-Universität zu Berlin

Inhaltsverzeichnis

Kapitel 0

Einleitung

Logiken zur Beschreibung von Berechnungsproblemen

Folie 1

Beispiel 0.1 (3-Färbbarkeit von Graphen)
Ein Graph $\mathcal{G} = (V, E)$ heißt *3-färbbar*, falls seine Knoten so mit 3 Farben gefärbt werden können, dass benachbarte Knoten verschiedene Farben haben.

Somit gilt:

$\mathcal{G} = (V, E)$ ist 3-färbbar

$\Leftrightarrow$ es gibt $R \subseteq V, G \subseteq V, B \subseteq V$, so dass $V = R \dot\cup G \dot\cup B$ und
für alle $u, v \in V$ gilt: falls $(u, v) \in E$, so gilt
$\neg\Big(\big(u \in R \wedge v \in R\big) \vee \big(u \in G \wedge v \in G\big) \vee \big(u \in B \wedge v \in B\big)\Big)$

$\Leftrightarrow \mathcal{G} \models \Phi_{\text{3-COL}}$, wobei

Folie 2

$$\Phi_{\text{3-COL}} := \exists R\,\exists G\,\exists B\Bigg(\forall v\,\Big(\big(R(v)\wedge\neg G(v)\wedge\neg B(v)\big)\vee\big(\neg R(v)\wedge G(v)\wedge\neg B(v)\big)\vee\big(\neg R(v)\wedge\neg G(v)\wedge B(v)\big)\Big)\wedge\forall u\,\forall v\Big(E(u,v)\rightarrow\neg\Big(\big(R(u)\wedge R(v)\big)\vee\big(G(u)\wedge G(v)\big)\vee\big(B(u)\wedge B(v)\big)\Big)\Big)\Bigg)$$

$\Phi_{\text{3-COL}}$ ist eine Formel der *Logik zweiter Stufe* (SO), die eine wichtige Rolle in dieser Vorlesung spielt. Wir werden z.B. Folgendes nachweisen:

Folie 3

Satz von Fagin:
Ein Problem gehört genau dann zur Komplexitätsklasse NP, wenn es durch einen Satz der *existentiellen Logik zweiter Stufe* (ESO) beschrieben werden kann.

Bemerkung
Unter Verwendung von Ehrenfeucht-Fraïssé-Spielen und logischen Reduktionen werden wir zeigen, dass 3-Färbbarkeit von Graphen nicht durch einen Satz der Logik erster Stufe (FO) beschrieben werden kann.

Folie 4

Beispiel 0.2 (Erreichbarkeit)
Wir betrachten das Berechnungsproblem

Erreichbarkeit
Eingabe: ein Graph $\mathcal{G} = (V, E)$ und zwei Knoten $s, t \in V$
Frage: Gibt es in $\mathcal{G}$ einen Weg von s nach t?

Dann gilt:

$$\text{Antwort „ja“} \iff (\mathcal{G}, s, t) \models \Phi_{\text{REACH}}, \text{ wobei}$$

$$\Phi_{\text{REACH}} := [\mathsf{lfp}_{R,x}\varphi](t) \text{ mit}$$

$$\varphi := \Big(x = s \ \vee \ \exists z \ \big(R(z) \wedge E(z,x)\big)\Big)$$

Folie 5

Zur Auswertung von Φ_{REACH} in $(\mathcal{G}, s, t)$ wird induktiv eine 1-stellige Relation R definiert:

- Starte mit $R := \emptyset$
- Iteriere so lange, bis sich nichts mehr ändert:
  - $R_{\text{neu}} := \{x \in V \ : \ (\mathcal{G}, s, t, R, x) \models \varphi\}$
  - $R := R_{\text{neu}}$

Φ_{REACH} ist eine Formel der *kleinsten Fixpunktlogik* (LFP). Fixpunktlogiken dienen der Charakterisierung der Komplexitätsklassen P und PSPACE.

Logiken zur Beschreibung formaler Sprachen

Folie 6

Wir betrachten Worte über dem Alphabet $\Sigma = \{a, b\}$ und repräsentieren nicht-leere Worte $w \in \Sigma^*$ durch Strukturen über der Signatur $\sigma_\Sigma := \{\leqslant, P_a, P_b\}$, wobei $\leqslant$ 2-stellig ist, und P_a sowie P_b 1-stellig sind.

Beispiel
Das Wort $w = aabab$ wird durch die Wortstruktur $\mathcal{A}_w$ mit Universum $\{1, 2, 3, 4, 5\}$ repräsentiert, wobei $\leqslant^{\mathcal{A}_w}$ die natürliche Ordnung auf dem Universum ist und $P_a^{\mathcal{A}_w} = \{1, 2, 4\}$ bzw. $P_b^{\mathcal{A}_w} = \{3, 5\}$ die Menge der Positionen von w ist, an denen der Buchstabe a bzw. der Buchstabe b steht.

Folie 7

Beispiel 0.3
Die durch den regulären Ausdruck a^+b^* beschriebene Sprache $L_{a^+b^*}$ wird durch die FO[σ_Σ]-Formel

$$\varphi_{a^+b^*} := \exists x\, \forall y \left(P_a(y) \leftrightarrow y \leqslant x\right)$$

beschrieben, d.h., es gilt für jedes nicht-leere Wort $w \in \Sigma^*$:

$$w \in L_{a^+b^*} \iff \mathcal{A}_w \models \varphi_{a^+b^*}.$$

Beispiel 0.4
Die Sprache $L := \{w \in \Sigma^* : \#_a(w) \text{ ist ungerade}\}$ wird durch folgende Formel Φ_L der *monadischen Logik zweiter Stufe* (MSO) beschrieben. Dabei steht der Ausdruck $\#_a(w)$ für die Anzahl der a's in dem Wort w.

Folie 8

$$\begin{aligned}
\Phi_L := \exists Z \Bigg(& \\
& \forall x \left(\left(P_a(x) \wedge \neg \exists y \left(P_a(y) \wedge y \leqslant x \wedge \neg\, y{=}x \right) \right) \rightarrow Z(x) \right) \wedge \\
& \forall x\, \forall y \Bigg(\Big(\quad P_a(x) \wedge P_a(y) \wedge x \leqslant y \wedge \neg\, x{=}y \wedge \\
& \qquad\qquad \neg \exists z \left(x \leqslant z \wedge z \leqslant y \wedge \neg\, z{=}x \wedge \neg\, z{=}y \wedge P_a(z) \right) \Big) \\
& \qquad\qquad \rightarrow \Big(Z(x) \leftrightarrow \neg Z(y) \Big) \Bigg) \wedge \\
& \forall x \left(\left(P_a(x) \wedge \neg \exists y \left(P_a(y) \wedge x \leqslant y \wedge \neg\, y{=}x \right) \right) \rightarrow Z(x) \right) \wedge \\
& \exists x\, P_a(x) \Bigg)
\end{aligned}$$

Folie 9

Die Zeile 2 der Formel besagt (in $(\mathcal{A}_w, Z)$), dass die Menge Z die erste Position enthält, die den Buchstaben a trägt.

Die Zeilen 3 bis 5 der Formel besagen, dass die Menge Z „jede zweite“ Position enthält, die den Buchstaben a trägt. Genauer: Sind x und y Positionen, an denen ein a steht und zwischen denen kein weiteres a steht, so gehört genau eine der beiden Positionen x und y zu Z.

Die Zeile 6 der Formel besagt, dass die letzte Position, die den Buchstaben a trägt, zur Menge Z gehört.

Die letzte Zeile 7 der Formel gewährleistet, dass das Wort mindestens ein a enthält.

Folie 10

Insgesamt gilt für jedes nicht-leere Wort $w \in \Sigma^*$:

$$\begin{aligned} \mathcal{A}_w \models \Phi_L &\iff w \text{ enthält eine ungerade Anzahl von a's} \\ &\iff w \in L. \end{aligned}$$

Wir werden Folgendes nachweisen:

> (Teil aus dem) Satz von Büchi:
> Eine Sprache $L \subseteq \Sigma^+$ ist genau dann regulär, wenn sie von einem $\mathsf{MSO}[\sigma_\Sigma]$-Satz beschrieben wird.

Logiken zur Beschreibung von Datenbank-Anfragen

Folie 11

Beispiele dazu wurden bereits in der Veranstaltung „Logik in der Informatik“ betrachet.

Voraussetzungen zur Teilnahme an der Veranstaltung

Folie 12

Die Beherrschung des in der Veranstaltung „Logik in der Informatik“ vermittelten Stoffs ist Voraussetzung für die Teilnahme an der Veranstaltung „Logik und Komplexität“.

Zur Vorbereitung wird dringend empfohlen, insbesondere die Kapitel „Logik erster Stufe“ und „Grundlagen des automatischen Schließens“ des Skripts „Logik in der Informatik“ (N. Schweikardt, HU Berlin, [**S-LI**]) durchzuarbeiten.

Aufbau der Vorlesung „Logik und Komplexität“

Folie 13

Kapitel 0: Einleitung

Kapitel 1: Grundlagen und der Satz von Trakhtenbrot

Kapitel 2: Logik zweiter Stufe und die Sätze von Büchi und Fagin

Kapitel 3: Ehrenfeucht-Fraïssé-Spiele und Lokalitätsresultate

Kapitel 4: 0-1-Gesetze

Kapitel 5: Fixpunktlogiken und der Satz von Immerman und Vardi

Organisatorisches

Folie 14

Website zur Vorlesung:

https://www.informatik.hu-berlin.de/de/forschung/gebiete/loginf/lehre/wise-2025-26/vorlesung-logik-und-komplexitaet

Dort finden sich auch Details zum Ablauf des Übungsbetriebs und zur Zulassung zur Modulabschlussprüfung.

Weitere Lektüre

Folie 15

- Lehrbuch [**L**]: Kapitel 1.
- Viele weitere Beispiele zur Bedeutung der Logik in der Informatik finden sich in dem Artikel

  „*On the unusual effectiveness of logic in computer science*“ von Halpern, Harper, Immerman, Kolaitis, Vardi und Vianu, Bulletin of Symbolic Logic 7(2):213-236 (2001)

  Eine Vorabversion des Artikels finden Sie hier.

Kapitel 1

Grundlagen und der Satz von Trakhtenbrot

Folie 16

Notation

- s.d. : „so dass“
- f.a. : „für alle“
- ex. : „es existiert / es gibt“
- leeres Wort: ε
- $\Sigma^+ := \Sigma^* \setminus \{\varepsilon\}$
- f.a. $w \in \Sigma^+$, für $n := |w|$ und f.a. $i \in \{0, 1, \ldots, n-1\}$ schreibe w_i für das Symbol an Position i in w, d.h. $w = w_0 w_1 \ldots w_{n-1}$
- Potenzmenge einer Menge M:
  $\mathcal{P}(M) := \mathrm{Pot}(M) := 2^M := \{X : X \subseteq M\}$
- A, B Mengen, $f : A \to B$, $\overline{a} = (a_1, \ldots, a_k) \in A^k$
  $\rightsquigarrow\ f(\overline{a}) := \big(f(a_1), \ldots, f(a_k)\big) \in B^k$
- $R \subseteq A^k \ \rightsquigarrow\ f(R) := \{f(\overline{a}) : \overline{a} \in R\}$
- natürliche Zahlen:

$$\begin{aligned} \mathbb{N} &:= \{0, 1, 2, \ldots\} \\ \mathbb{N}_{\geqslant 1} &:= \mathbb{N} \setminus \{0\} \end{aligned}$$

Syntax und Semantik der Logik erster Stufe (FO)

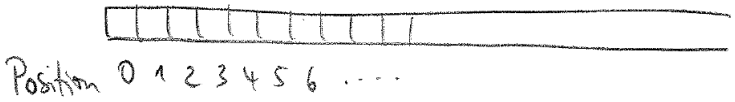
Folie 17

siehe Skript zur Vorlesung „Logik in der Informatik“ (N. Schweikardt, HU Berlin, [**S-LI**])

Turingmaschinen

Folie 18

Intuitiv stellt eine Turingmaschine (TM) ein Band dar, welches linksseitig begrenzt ist:



Folie 19

Definition 1.1 (Turingmaschine)
Eine *nichtdeterministische Turingmaschine* (NTM)

$$M = (Q, \Sigma, \Gamma, \Delta, q_0, F)$$

besteht aus

- einer endlichen Menge Q von *Zuständen*
- einem endlichen *Arbeitsalphabet* Γ
  mit ausgezeichnetem *Blank-Symbol* $\square$
- einem *Eingabealphabet* $\Sigma \subseteq \Gamma \setminus \{\square\}$
- einem *Anfangszustand* $q_0 \in Q$
- einer Menge $F = F_{\text{akz}} \,\dot\cup\, F_{\text{verw}} \subseteq Q$ von *Endzuständen*, die aus einer Menge F_{akz} von *akzeptierenden* und einer Menge F_{verw} von *verwerfenden* Zuständen besteht
- einer *Übergangsrelation* $\Delta \subseteq (Q \setminus F) \times \Gamma \times Q \times \Gamma \times \{-1, 0, 1\}$.

Folie 20

M heißt *deterministisch* (kurz: M ist eine DTM), falls f.a. $q \in Q \setminus F$ und $a \in \Gamma$ genau ein $q' \in Q$, $a' \in \Gamma$ und $m \in \{-1, 0, 1\}$ mit $(q, a, q', a', m) \in \Delta$ existiert. In diesem Fall schreiben wir oft

$$M \;=\; (Q, \Sigma, \Gamma, \delta, q_0, F)$$

mit *Überführungsfunktion* $\delta : (Q \setminus F) \times \Gamma \to Q \times \Gamma \times \{-1, 0, 1\}$.

Folie 21

Definition 1.2 (Konfiguration einer TM)

(a) Eine *Konfiguration* einer TM $M = (Q, \Sigma, \Gamma, \Delta, q_0, F)$ ist ein Tripel

$$\mathrm{C} \;=\; (q, p, u) \;\in\; Q \times \mathbb{N} \times \Gamma^* \quad \text{mit } p < |u|.$$

Idee: $\mathrm{C} = (q, p, u)$ gibt an, dass die TM sich im Zustand q befindet, der Kopf an Position p steht und die Inschrift des Arbeitsbandes u gefolgt von Blank-Symbolen ist.

(b) $\mathcal{C}_M := \{(q, p, u) \in Q \times \mathbb{N} \times \Gamma^* : p < |u|\}$ bezeichnet die Menge aller möglichen Konfigurationen von M.

(c) Die *Startkonfiguration* von M bei Eingabe $w \in \Sigma^*$ ist

$$\mathrm{C}_0(w) := (q_0, 0, w\Box).$$

(d) Eine Konfiguration $\mathrm{C} = (q, p, u)$ heißt *Endkonfiguration*, falls $q \in F$. Sie heißt *akzeptierend*, falls $q \in F_{\text{akz}}$, und *verwerfend*, falls $q \in F_{\text{verw}}$.

Folie 22

Definition 1.3 (Lauf einer TM)
Sei $M = (Q, \Sigma, \Gamma, \Delta, q_0, F)$ eine TM.

(a) Die Übergangsrelation Δ induziert eine Funktion

$$\mathrm{Next}_M : \mathcal{C}_M \to \mathcal{P}(\mathcal{C}_M),$$

wobei für $\mathrm{C} = (q, p, u) \in \mathcal{C}_M$ gilt:

$$\begin{aligned}
\mathrm{Next}_M(\mathrm{C}) \;=\; \{\; (q', p', u') \in \mathcal{C}_M \;:\; & \text{es gibt } (q, a, q', b, m) \in \Delta \text{ mit:} \\
& p' = p + m, \\
& u_p = a, \\
& u'_p = b, \\
& |u'| = |u|, && \text{falls } p' < |u|, \\
& |u'| = |u| + 1 \text{ und } u'_{p'} = \Box, && \text{falls } p' = |u|, \\
& u'_i = u_i \;\text{ f.a. } i < |u| \text{ mit } i \neq p \;\}
\end{aligned}$$

D.h.: $\text{Next}_M(\text{C})$ enthält alle Konfigurationen, in die M von C aus in genau einem Schritt gelangen kann.

Beachte: Ist M deterministisch, so ist $|\text{Next}_M(\text{C})| \leqslant 1$ f.a. $\text{C} \in \mathcal{C}_M$.

Folie 23

(b) Ein *Lauf* von M bei Eingabe w ist ein Tupel $\mathcal{L} = \big(\text{C}_0, \text{C}_1, \ldots\big)$ von Konfigurationen von M, so dass gilt:

- $\text{C}_0 = \text{C}_0(w)$,
- $\text{C}_i \in \text{Next}_M(\text{C}_{i-1})$ f.a. $i \geqslant 1$, und
- entweder ist das Tupel $\mathcal{L}$ unendlich lang, oder es endet mit einer Endkonfiguration. Im letzteren Fall heißt der Lauf dann *endlich* bzw. *terminierend*.

Eine DTM hat folglich für jede Eingabe maximal einen Lauf.

(c) Ein Lauf heißt *akzeptierend* (bzw. *verwerfend*), falls er in einer akzeptierenden (bzw. verwerfenden) Endkonfiguration endet.

Folie 24

Definition 1.4 (Sprache einer TM)

(a) Eine TM M *akzeptiert* eine Eingabe $w \in \Sigma^*$, falls es (mindestens) einen akzeptierenden Lauf von M auf w gibt. M *verwirft* w, falls alle terminierenden Läufe von M auf w verwerfen.

(b) Die von einer TM M *akzeptierte Sprache* ist

$$L(M) := \{w \in \Sigma^* : M \text{ akzeptiert } w\}.$$

(c) Eine Sprache $L \subseteq \Sigma^*$ heißt *semi-entscheidbar*, falls es eine TM M mit $L(M) = L$ gibt.

(d) Eine Sprache $L \subseteq \Sigma^*$ heißt *entscheidbar*, falls es eine TM M mit $L(M) = L$ gibt, so dass jeder Lauf von M auf jeder Eingabe $w \in \Sigma^*$ terminiert.

Folie 25

Aus der Veranstaltung „Einführung in die Theoretische Informatik“ (HU Berlin) sind die beiden folgenden Sätze bekannt (hier ohne Beweise):

Satz 1.5
Jede durch eine NTM entscheidbare (bzw. semi-entscheidbare) Sprache ist auch durch eine DTM entscheidbar (bzw. semi-entscheidbar).

Satz 1.6
Das Halteproblem auf leerem Eingabewort

H_ε
Eingabe: Eine DTM M
Frage: Hält M bei Eingabe des leeren Worts ε?

ist nicht entscheidbar. Es ist jedoch semi-entscheidbar.

Der Satz von Trakhtenbrot

Folie 26

Boris A. Trakhtenbrot: russisch-israelischer Mathematiker (1921–2016).
Alternative Schreibweisen: Trachtenbrot, Trahenbrot.

Definition 1.7
Eine funktionenfreie, endliche *Signatur* (im Folgenden kurz: Signatur) ist eine endliche Menge

$$\sigma = \{R_1, \ldots, R_k, c_1, \ldots, c_\ell\} \text{ mit } k, \ell \in \mathbb{N}.$$

Dabei stellen $R_1, \ldots, R_k$ Relations- und $c_1, \ldots, c_\ell$ Konstantensymbole dar. Jedes R_i hat eine feste Stelligkeit $\text{ar}(R_i) \in \mathbb{N}_{\geqslant 1}$.

Folie 27

Definition 1.8
Sei σ eine (funktionenfreie, endliche) Signatur.
Das *endliche Erfüllbarkeitsproblem für* $\mathsf{FO}[\sigma]$ ist das Berechnungsproblem mit:

Eingabe: Ein $\mathsf{FO}[\sigma]$-Satz φ
Frage: Gibt es eine endliche σ-Struktur $\mathcal{A}$ mit $\mathcal{A} \models \varphi$?

Bemerkung
Man kann sich leicht überlegen, dass das endliche Erfüllbarkeitsproblem für $\mathsf{FO}[\sigma]$ semi-entscheidbar ist.

Theorem 1.9 (Satz von Trakhtenbrot, 1950)
Es gibt eine (endliche, funktionenfreie) Signatur σ, so dass das endliche Erfüllbarkeitsproblem für $\mathsf{FO}[\sigma]$ *unentscheidbar ist.*

Beweis. Durch Widerspruch: Angenommen, das endliche Erfüllbarkeitsproblem für $\mathsf{FO}[\sigma]$ wäre für jede (endliche, funktionenfreie) Signatur σ doch entscheidbar. Wir zeigen, dass dann auch das Halteproblem H_ε entscheidbar wäre. Dies würde einen Widerspruch zu Satz 1.6 darstellen.

Sei dazu M eine DTM. Wir können o.B.d.A. annehmen, dass $M = (Q, \Sigma, \Gamma, \Delta, q_0, F)$ ist mit

- $Q = \{0, 1, ..., s_Q\}$ für ein $s_Q \in \mathbb{N}$
- Anfangszustand $q_0 = 0$
- $\Gamma = \{0, 1, ..., s_\Gamma\}$ für ein $s_\Gamma \in \mathbb{N}$
- Blank-Symbol $\square = 0$
- $s := \max\{s_Q, s_\Gamma\}$
- „Der Lauf“ von M bei Eingabe ε existiert wirklich, d.h. die Maschine M bleibt während der Berechnung nicht dadurch stecken, dass sie in einem Nicht-Endzustand ist, ihr Kopf an Position 0 (ganz links) ist, und die Überführungsfunktion im nächsten Schritt eine Kopfbewegung nach links erfordern würde. (Übung: Man überlege, wie dies o.B.d.A. gewährleistet werden kann!)
- Falls der Lauf von M bei Eingabe ε terminiert, so nach genau n Schritten, wobei n eine natürliche Zahl $\geqslant s$ ist. (Übung: Man überlege, wie “$n \geqslant s$” o.B.d.A. gewährleistet werden kann!)

Wir repräsentieren die Berechnung von M bei Eingabe von ε durch einen $\mathsf{FO}[\sigma]$-Satz φ_M, so dass gilt:

φ_M hat ein endliches Modell $\iff$ der Lauf von M bei Eingabe ε terminiert.

Die Signatur σ wird (unabhängig von der konkreten DTM M) wie folgt gewählt:

$$\sigma \;:=\; \{<, succ, 0, B, K, Z\},$$

wobei

- $<$ und *succ* zwei 2-stellige Relationssymbole,
- 0 ein Konstantensymbol,
- B (Bandbeschriftung) ein 3-stelliges Relationssymbol, und
- K (Kopfposition) und Z (Zustand) zwei 2-stellige Relationssymbole sind.

Wir definieren nun eine σ-Struktur $\mathcal{A}_M$, die den Lauf von M bei Eingabe ε repräsentiert:

- Das Universum von $\mathcal{A}_M$ ist die Menge

$$A_M := \begin{cases} \{0, \ldots, n\}, & \text{falls der Lauf von } M \text{ bei Eingabe } \varepsilon \text{ in Schritt } n \in \mathbb{N} \text{ terminiert} \\ \mathbb{N}, & \text{falls der Lauf von } M \text{ bei Eingabe } \varepsilon \text{ nicht terminiert} \end{cases}$$

- Die Relationen $<^{\mathcal{A}_M}$ und $succ^{\mathcal{A}_M}$ sowie die Konstante $0^{\mathcal{A}_M}$ sind durch die natürliche strikte lineare Ordnung auf A_M, deren Nachfolger-Relation sowie die Zahl 0 belegt.

Für alle $t, p, \gamma, q \in A_M$ gilt:

- $(t, p, \gamma) \in B^{\mathcal{A}_M} \iff$
  zum Zeitpunkt t des Laufs von M bei Eingabe ε steht auf Bandposition p der Buchstabe γ
- $(t, p) \in K^{\mathcal{A}_M} \iff$
  zum Zeitpunkt t des Laufs von M bei Eingabe ε steht der Schreib-/Lesekopf von M auf Bandposition p
- $(t, q) \in Z^{\mathcal{A}_M} \iff$
  zum Zeitpunkt t des Laufs von M bei Eingabe ε ist M im Zustand q.

Wir definieren nun einen $\mathsf{FO}[\sigma]$-Satz φ_M, der erzwingen soll, dass die Modelle $\mathcal{A}$ von φ_M eine zur Struktur $\mathcal{A}_M$ isomorphe Substruktur enthalten. Dies gewährleistet dann, dass gilt:

$$\begin{aligned} & \varphi_M \text{ hat ein endliches Modell} \\ \Rightarrow\ & \mathcal{A}_M \text{ ist endlich} \\ \Rightarrow\ & M \text{ hält bei Eingabe } \varepsilon. \end{aligned}$$

Außerdem konstruieren wir φ_M so, dass auch gilt: $\mathcal{A}_M \models \varphi_M$. Dies gewährleistet dann, dass gilt:

$$\begin{aligned} & M \text{ hält bei Eingabe } \varepsilon \\ \Rightarrow\ & \mathcal{A}_M \text{ ist endlich} \\ \Rightarrow\ & \varphi_M \text{ hat ein endliches Modell.} \end{aligned}$$

Insgesamt gilt dann also:

$$\varphi_M \text{ hat ein endliches Modell} \iff M \text{ hält bei Eingabe } \varepsilon.$$

Zur Konstruktion von φ_M
Damit für ein Modell $\mathcal{A}$ von φ_M mit Universum A die Relation $<^{\mathcal{A}}$ eine strikte lineare Ordnung auf A mit zugehöriger Nachfolgerrelation $succ^{\mathcal{A}}$ und kleinstem Element $0^{\mathcal{A}}$ ist, nutzen wir folgende $\mathsf{FO}[\sigma]$-Formel:

$$\begin{aligned} \varphi_{<,succ,0} := \forall x\, \forall y\, \forall z \Bigg(& \Big(\big(x < y \wedge y < z \big) \rightarrow x < z \Big) \\ & \wedge \Big(x < y \rightarrow \neg\, y < x \Big) \\ & \wedge \Big(x < y \vee y < x \vee x{=}y \Big) \\ & \wedge \Big(0{=}x \vee 0 < x \Big) \\ & \wedge \Big(succ(x,y) \leftrightarrow \big(x < y \wedge \neg \exists u \, \big(x < u \wedge u < y \big) \big) \Big) \Bigg) \end{aligned}$$

Die ersten drei Zeilen dieser Formel besagen, dass die $<$-Relation transitiv, antisymmetrisch und konnex (also eine strikte lineare Ordnung) ist, die vierte Zeile besagt, dass die Konstante 0 mit dem kleinsten Element dieser strikten linearen Ordnung belegt ist, und die letzte Zeile besagt, dass *succ* die zu $<$ passende Nachfolger-Relation ist.

Zur Erinnerung:
$s = \max\{s_Q, s_\Gamma\}$, wobei $Q = \{0, 1, ..., s_Q\}$ und $\Gamma = \{0, 1, ..., s_\Gamma\}$.

Ebenso wie $\mathcal{A}_M$ soll jedes Modell $\mathcal{A}$ der Formel φ_M Elemente $a_0, \ldots, a_s$ enthalten, sodass a_0 das kleinste Element der linearen Ordnung ist und a_i der Nachfolger von a_{i-1} für jedes $i \in \{1, \ldots, s\}$ ist, und welche die Rolle von Zuständen bzw. Buchstaben einnehmen können. In A_M sind dies die Zahlen $0, \ldots, s$.

Die folgende Formel beschreibt die gewünschte Eigenschaft der Elemente:

$$\varphi_{\text{Zahlen}}(z_0, z_1, \ldots, z_s) \;:=\; \left(z_0{=}0 \;\wedge\; \bigwedge_{i=1}^{s} succ(z_{i-1}, z_i) \right)$$

Der FO[σ]-Satz φ_M wird nun folgendermaßen gewählt:

$$\begin{aligned} \varphi_M \;:=\; \varphi_{<,succ,0} \wedge \exists z_0 \ldots \exists z_s \big(& \varphi_{\text{Zahlen}}(z_0, \ldots, z_s) \wedge \\ & \varphi_{\text{Band}} \wedge \varphi_{\text{Kopf}} \wedge \varphi_{\text{Zustand}} \wedge \\ & \varphi_{\text{Start}} \wedge \varphi_{\text{Schritt}} \big), \end{aligned}$$

wobei die Formeln in den letzten beiden Zeilen wie folgt gewählt sind:

- φ_{Band} besagt (in Modellen von φ_M), dass zu jedem Zeitpunkt auf jeder Bandposition genau ein Symbol des Arbeitsalphabets $\Gamma = \{0, 1, \ldots, s_\Gamma\}$ steht:

$$\varphi_{\text{Band}} := \forall x \, \forall y \, \exists z \left(B(x, y, z) \;\wedge\; \Big(\bigvee_{i=0}^{s_\Gamma} z{=}z_i \Big) \;\wedge\; \forall z' \left(B(x, y, z') \;\rightarrow\; z'{=}z \right) \right)$$

- φ_{Zustand} besagt, dass M zu jedem Zeitpunkt in genau einem Zustand aus $Q = \{0, 1, \ldots, s_Q\}$ ist:

$$\varphi_{\text{Zustand}} \;:=\; \forall x \, \exists z \left(Z(x, z) \;\wedge\; \Big(\bigvee_{i=0}^{s_Q} z{=}z_i \Big) \;\wedge\; \forall z' \left(Z(x, z') \;\rightarrow\; z'{=}z \right) \right)$$

- φ_{Kopf} besagt, dass der Schreib-/Lesekopf von M zu jedem Zeitpunkt auf genau einer Bandposition steht:

$$\varphi_{\text{Kopf}} \;:=\; \forall x \, \exists y \left(K(x, y) \;\wedge\; \forall y' \left(K(x, y') \;\rightarrow\; y'{=}y \right) \right)$$

- φ_{Start} besagt, dass die Maschine M zum Zeitpunkt 0 in der Startkonfiguration $\mathrm{C}_0(\varepsilon)$ bei Eingabe des leeren Worts ist, das heißt sie ist im Startzustand $q_0 = 0$, ihr Schreib-/Lesekopf steht auf der Bandposition 0, und auf jeder Bandposition steht das Blank-Symbol $\square = 0$:

$$\varphi_{\text{Start}} \;:=\; \Big(Z(0, 0) \;\wedge\; K(0, 0) \;\wedge\; \forall y \, B(0, y, 0) \Big)$$

- φ_{Schritt} besagt für jeden Zeitpunkt t: Falls M zum Zeitpunkt t im Zustand $q \in Q \setminus F$ ist, der Schreib-/Lesekopf an Position p ist und dort das Symbol γ liest, so ist M zum Zeitpunkt $t' := t+1$ in einem laut Übergangsrelation zulässigen Zustand q', hat das entsprechende Symbol γ' auf das Band geschrieben, den Kopf an die richtige Stelle p' bewegt und die Beschriftung aller anderen Bandpositionen p'' nicht verändert:

$$
\begin{aligned}
&\varphi_{\text{Schritt}} := \\
&\forall t\, \forall p \bigwedge_{q \in Q\setminus F} \bigwedge_{\gamma\in\Gamma} \Bigg(\Big(K(t,p) \,\wedge\, Z(t,z_q) \,\wedge\, B(t,p,z_\gamma) \Big) \rightarrow \\
&\qquad\qquad \exists t'\, \exists p' \Big(succ(t,t') \,\wedge\, K(t',p') \,\wedge \\
&\forall p'' \Big(\neg\, p''{=}p \,\rightarrow\, \bigwedge_{\gamma''\in\Gamma} \Big(B(t',p'',z_{\gamma''}) \,\leftrightarrow\, B(t,p'',z_{\gamma''}) \Big) \Big) \,\wedge \\
&\Big(\bigvee_{\substack{q',\gamma',m\,:\\ (q,\gamma,q',\gamma',m)\in\Delta}} \Big(Z(t',z_{q'}) \,\wedge\, B(t',p,z_{\gamma'}) \,\wedge\, \chi_m(p,p') \Big) \Big) \Big) \Big) \Bigg),
\end{aligned}
$$

  wobei die Formel $\chi_m(p,p')$ die Kopfbewegung für $m \in \{-1,0,1\}$ beschreibt, das heißt

$$
\begin{aligned}
\chi_0(p,p') &:= p'{=}p, \\
\chi_1(p,p') &:= succ(p,p'), \\
\chi_{-1}(p,p') &:= succ(p',p).
\end{aligned}
$$

Wir sind nun fertig mit der Konstruktion der Formel φ_M. Man sieht leicht, dass es einen Algorithmus gibt, der bei Eingabe einer DTM M den $\mathsf{FO}[\sigma]$-Satz φ_M konstruiert.
Außerdem kann man sich leicht davon überzeugen (Details: Übung!), dass Folgendes gilt:

- $\mathcal{A}_M \models \varphi_M$, und
- in jeder σ-Struktur $\mathcal{A}$ mit $\mathcal{A} \models \varphi_M$ gibt es Elemente $a_0, a_1, \ldots, a_i, \ldots$, die den natürlichen Zahlen $0, 1, \ldots, i, \ldots$ entsprechen (für alle $i \in A_M$); und schränkt man $\mathcal{A}$ ein auf das Teiluniversum $\{a_i \,:\, i \in A_M\}$, so erhält man eine Struktur, die isomorph zu $\mathcal{A}_M$ ist.

Das Halteproblem H_ε kann dann dadurch getestet werden, dass man φ_M auf endliche Erfüllbarkeit testet. Dies ist ein Widerspruch zu Satz 1.6. □

Folie 28

Bemerkung 1.10
Man kann sogar zeigen, dass der Satz von Trakhtenbrot für die Signatur $\sigma := \{E\}$ gilt, wobei E ein 2-stelliges Relationssymbol ist.
Daraus folgt dann natürlich auch, dass der Satz von Trakhtenbrot für jede Signatur gilt, die mindestens ein Relationssymbol der Stelligkeit $\geqslant 2$ enthält.

Andererseits werden wir in einem späteren Kapitel unter Verwendung von Lokalitätsresultaten zeigen, dass für jede funktionenfreie Signatur σ, deren Relationssymbole alle die Stelligkeit 1 haben, das endliche Erfüllbarkeitsproblem für $\mathsf{FO}[\sigma]$ entscheidbar ist.

Bemerkung 1.11
Alternativ zum Halteproblem H_ε kann man auf Grund des Satzes von Trakhtenbrot auch das endliche Erfüllbarkeitsproblem für $\mathsf{FO}[\sigma]$ als Grundlage für Reduktionen nutzen, mit denen man die Unentscheidbarkeit bestimmter Probleme nachweist. Beispiele dazu werden in den Übungen betrachtet.

Weitere Lektüre

Folie 29

- Lehrbuch [**L**]: Kapitel 2.1, 2.2 und 9.1.

Kapitel 2

Logik zweiter Stufe und die Sätze von Büchi und Fagin

2.1 Syntax und Semantik der Logik zweiter Stufe

Folie 30

Idee: In der Logik zweiter Stufe gibt es für jede Stelligkeit $k \geqslant 1$ abzählbar viele k-stellige Relationsvariablen $\mathrm{Var}_1^k, \mathrm{Var}_2^k, \ldots$, die wir meistens mit Großbuchstaben wie $X, Y, \ldots$ bezeichnen werden.

Definition 2.1 (SO: Syntax)
Sei σ eine (endliche, funktionenfreie) Signatur.

(a) $\mathrm{Var}_2 := \{\mathrm{Var}_i^k \,;\, i, k \in \mathbb{N}_{\geqslant 1}\}$ ist die Menge aller *Variablen zweiter Stufe* (oder: *Relationsvariablen*). Die Relationsvariable Var_i^k hat die Stelligkeit $\mathrm{ar}(\mathrm{Var}_i^k) = k$.

$\mathrm{Var}_1 := \mathrm{Var} = \{\mathrm{var}_i \,;\, i \in \mathbb{N}_{\geqslant 1}\}$ ist die Menge aller *Variablen erster Stufe* (oder: *Individuenvariablen*).

Ein σ-*Term* ist ein Element der Menge
$\mathsf{T}_\sigma := \mathrm{Var}_1 \cup \{c \in \sigma : c \text{ ist ein Konstantensymbol}\}$.

Folie 31

(b) Die Formelmenge $\mathsf{SO}[\sigma]$ ist rekursiv wie folgt definiert:

(A1) $R(t_1, \ldots, t_k)$ gehört zu $\mathsf{SO}[\sigma]$
f.a. Relationssymbole $R \in \sigma$, $k := \mathrm{ar}(R)$, $t_1, \ldots, t_k \in \mathsf{T}_\sigma$

(A2) $t{=}t'$ gehört zu $\mathsf{SO}[\sigma]$
f.a. $t, t' \in \mathsf{T}_\sigma$

(A3) $X(t_1, \ldots, t_k)$ gehört zu $\mathsf{SO}[\sigma]$
f.a. $X \in \mathrm{Var}_2$, $k := \mathrm{ar}(X)$, $t_1, \ldots, t_k \in \mathsf{T}_\sigma$

(BC) Sind ψ, φ_1, φ_2 Formeln in $\mathsf{SO}[\sigma]$, so gehören auch die folgenden Formeln zu $\mathsf{SO}[\sigma]$:

- $\neg\psi$
- $(\varphi_1 \vee \varphi_2)$
- $(\varphi_1 \wedge \varphi_2)$
- $(\varphi_1 \rightarrow \varphi_2)$
- $(\varphi_1 \leftrightarrow \varphi_2)$

(Q1) Ist ψ eine Formel in $\mathsf{SO}[\sigma]$ und $x \in \mathrm{Var}_1$, so gehören auch folgende Formeln zu $\mathsf{SO}[\sigma]$:

- $\exists x\, \psi$
- $\forall x\, \psi$

(Q2) Ist ψ eine Formel in $\mathsf{SO}[\sigma]$ und $X \in \mathrm{Var}_2$, so gehören auch folgende Formeln zu $\mathsf{SO}[\sigma]$:

- $\exists X\, \psi$
- $\forall X\, \psi$

(c) Die mit (A1), (A2) und (A3) gebildeten Formeln heißen *atomare σ-Formeln.*

Folie 32

Definition 2.2

(a) Für $\varphi \in \mathsf{SO}[\sigma]$ bezeichnet $\mathrm{frei}(\varphi)$ die Menge aller Individuen- und Relationsvariablen, die *frei* in φ vorkommen, und $\mathrm{qr}(\varphi)$ den *Quantorenrang* (bzw. die Quantorentiefe) von φ, d.h. die maximale Anzahl ineinander geschachtelter Quantoren in φ.

Das heißt, f.a. $* \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$, $Q \in \{\exists, \forall\}$, $v \in \mathrm{Var}_1 \cup \mathrm{Var}_2$, σ-Terme $t, t', t_1, \ldots, t_k$, Relationssymbole $R \in \sigma$, Variablen $X \in \mathrm{Var}_2$ und Formeln $\psi, \varphi_1, \varphi_2$ gilt

$$\begin{aligned}
\text{frei}(R(t_1,\ldots,t_k)) &= \{t_1,\ldots,t_k\} \cap \text{Var}_1,\\
\text{frei}(t{=}t') &= \{t,t'\} \cap \text{Var}_1,\\
\text{frei}(X(t_1,\ldots,t_k)) &= (\{t_1,\ldots,t_k\} \cap \text{Var}_1) \cup \{X\},\\
\text{frei}(\neg\psi) &= \text{frei}(\psi),\\
\text{frei}((\varphi_1 * \varphi_2)) &= \text{frei}(\varphi_1) \cup \text{frei}(\varphi_2),\\
\text{frei}(Qv\,\psi) &= \text{frei}(\psi) \setminus \{v\}
\end{aligned}$$

Folie 33

und

$$\begin{aligned}
\text{qr}(\varphi) &= 0 \quad \text{für atomare } \sigma\text{-Formeln } \varphi,\\
\text{qr}(\neg\psi) &= \text{qr}(\psi),\\
\text{qr}((\varphi_1 * \varphi_2)) &= \max\{\text{qr}(\varphi_1), \text{qr}(\varphi_2)\},\\
\text{qr}(Qv\,\psi) &= \text{qr}(\psi) + 1.
\end{aligned}$$

Wir schreiben oft $\varphi(X_1,\ldots,X_t,x_1,\ldots,x_s)$ um anzudeuten, dass $\text{frei}(\varphi) \subseteq \{X_1,\ldots,X_t,x_1,\ldots,x_s\}$ ist.

(b) Eine $\mathsf{SO}[\sigma]$-Formel φ heißt *Satz*, falls $\text{frei}(\varphi) = \emptyset$ ist.

Beispiel
Für die $\mathsf{SO}[\sigma]$-Formel

$$\varphi := \exists X\, \forall Y \Big(\exists z\, \exists Z\, Z(z) \vee \forall u \big(Y(u) \vee u{=}z \big) \Big)$$

gilt $\text{frei}(\varphi) = \{z\}$ und $\text{qr}(\varphi) = 4$.

Folie 34

Definition 2.3 (SO: Semantik)
Die Semantik von $\mathsf{SO}[\sigma]$ ist auf die offensichtliche Weise definiert, als Erweiterung der Semantik von $\mathsf{FO}[\sigma]$. Deshalb geben wir im Folgenden nur die zusätzlichen Fälle an.

Sei dazu $\mathcal{A}$ eine σ-Struktur und $\varphi(X_1,\ldots,X_t,x_1,\ldots,x_s)$ eine $\mathsf{SO}[\sigma]$-Formel.

Weiter sei β eine *Belegung* von $M := \{X_1,\ldots,X_t,x_1,\ldots,x_s\}$ in A, d.h. eine Zuordnung von M mit $\beta(x) \in A$ für alle $x \in M \cap \text{Var}_1$ und $\beta(X) \subseteq A^{\text{ar}(X)}$ für alle $X \in M \cap \text{Var}_2$.

Sei $\mathcal{I} := (\mathcal{A}, \beta)$ die zugehörige $\mathsf{SO}[\sigma]$-*Interpretation*.

Wir definieren $v^{\mathcal{I}} := \beta(v)$ für $v \in M$, und $c^{\mathcal{I}} := c^{\mathcal{A}}$ für Konstantensymbole $c \in \sigma$.
Für $X \in \mathrm{Var}_2$ und $S \subseteq A^{\mathrm{ar}(X)}$ bezeichne $\beta\frac{S}{X}$ die Belegung von $M \cup \{X\}$ in A mit $\beta\frac{S}{X}(X) = S$ und $\beta\frac{S}{X}(v) = \beta(v)$ für alle $v \in M \setminus \{X\}$.
Wir definieren außerdem $\mathcal{I}\frac{S}{X} := \big(\mathcal{A}, \beta\frac{S}{X}\big)$.

Folie 35

(A3) Ist φ von der Form $X(t_1, \ldots, t_k)$, so gilt

$$\mathcal{I} \models X(t_1, \ldots, t_k) \iff (t_1^{\mathcal{I}}, \ldots, t_k^{\mathcal{I}}) \in X^{\mathcal{I}}.$$

(Q2) Ist φ von der Form $\exists X\, \psi$ bzw. $\forall X\, \psi$, so gilt

$$\mathcal{I} \models \exists X\, \psi \iff \mathcal{I}\tfrac{S}{X} \models \psi \text{ für (mind.) eine Menge } S \subseteq A^{\mathrm{ar}(X)}$$

bzw.

$$\mathcal{I} \models \forall X\, \psi \iff \mathcal{I}\tfrac{S}{X} \models \psi \text{ für jede Menge } S \subseteq A^{\mathrm{ar}(X)}.$$

Allgemein schreiben wir statt $\mathcal{I} \models \varphi$ auch $(\mathcal{A}, X_1^{\mathcal{I}}, \ldots, X_t^{\mathcal{I}}, x_1^{\mathcal{I}}, \ldots, x_s^{\mathcal{I}}) \models \varphi$.

Folie 36

Beispiele 2.4
Sei $\sigma := \{E\}$ die Signatur für Graphen, das heißt E ist ein 2-stelliges Relationssymbol.

(a) Ein $\mathsf{SO}[\sigma]$-Satz, der genau dann für einen Graphen gilt, wenn er 3-färbbar ist:

$$\begin{aligned}\Phi_{\text{3-col}} := \exists R\, \exists G\, \exists B \Big(& \forall x \Big(R(x) \vee G(x) \vee B(x) \Big) \wedge \\ & \forall x\, \forall y \Big(E(x,y) \rightarrow \\ & \neg \Big(\big(R(x) \wedge R(y)\big) \vee \big(G(x) \wedge G(y)\big) \vee \\ & \big(B(x) \wedge B(y)\big) \Big) \Big) \Big)\end{aligned}$$

Folie 37

(b) Ein $\mathsf{SO}[\sigma]$-Satz, der besagt, dass der Graph symmetrisch und nicht zusammenhängend ist:

$$\Phi_{\text{non-conn}} \;:=\; \exists X \Big(\exists x\, X(x) \;\wedge \exists y\, \neg X(y) \wedge \\ \neg \exists u\, \exists v \, \big(E(u,v) \;\wedge\; X(u) \;\wedge\; \neg X(v) \big) \\ \wedge\; \forall u\, \forall v \, \Big(E(u,v) \;\leftrightarrow\; E(v,u) \Big) \Big)$$

Folie 38

(c) Eine $\mathsf{SO}[\sigma]$-Formel $\Phi_{\text{reach}}(x,y)$, die besagt, dass es im Graph einen Weg von Knoten x zu Knoten y gibt:

$$\Phi_{\text{reach}}(x,y) \;:=\; \forall X \Big(\Big(X(x) \;\wedge\; \forall z\, \forall z' \, \Big(\big(E(z,z') \;\wedge\; X(z) \big) \;\rightarrow\; X(z') \Big) \Big) \\ \rightarrow\; X(y) \Big)$$

Folie 39

(d) Ein $\mathsf{SO}[\sigma]$-Satz, der in einem endlichen Graphen $G = (V,E)$ mit $n := |V|$ besagt, dass er einen Hamiltonkreis enthält, d.h. es gibt paarweise verschiedene Knoten $v_1, \ldots, v_n$ in V, s.d. $(v_i, v_{i+1}) \in E$ f.a. $1 \leqslant i < n$, und $(v_n, v_1) \in E$:

$$\Phi_{\text{Ham}} \;:=\; \exists R_< \, \exists R_{succ} \, \exists z_0 \, \exists z_{max} \Big(E(z_{max}, z_0) \\ \wedge\; \forall x\, \forall y \, \big(R_{succ}(x,y) \rightarrow E(x,y) \big) \\ \underbrace{\wedge\; \varphi_{<,succ,0}(R_<, R_{succ}, z_0)}_{(*)} \\ \underbrace{\wedge\; \forall x \, \big(R_<(x, z_{max}) \vee x = z_{max} \big)}_{z_{max} \text{ ist das größte Element bzgl. } R_<} \Big)$$

($*$) Dies ist die Formel aus dem Beweis des Satzes von Trakhtenbrot, in der jedes Vorkommen von $<$ bzw. *succ* bzw. 0 durch $R_<$ bzw. R_{succ} bzw. z_0 ersetzt wurde. Sie besagt also, dass $R_<$ eine strikte lineare Ordnung auf V mit kleinstem Element z_0 und Nachfolger-Relation R_{succ} ist.

Folie 40

Definition 2.5

(a) Die *monadische Logik zweiter Stufe* $\mathsf{MSO}[\sigma]$ ist die Klasse aller $\mathsf{SO}[\sigma]$-Formeln φ, s.d. alle in φ vorkommenden Relationsvariablen die Stelligkeit 1 besitzen (solche Relationsvariablen werden auch *Mengenvariablen* genannt).

(b) Die *existenzielle Logik zweiter Stufe* $\mathsf{ESO}[\sigma]$ (auch: $\exists\mathsf{SO}[\sigma]$, $\Sigma^1_1[\sigma]$) ist die Klasse aller $\mathsf{SO}[\sigma]$-Formeln der Form $\exists X_1 \ldots \exists X_d\, \varphi$, wobei gilt: $d \in \mathbb{N}$, $X_1, \ldots, X_d$ sind Relationsvariablen, und $\varphi \in \mathsf{FO}[\sigma \cup \mathrm{Var}_2]$. Letzteres bedeutet, dass φ ohne (Q2) gebildet wird.

(c) Die *monadische existenzielle Logik zweiter Stufe* $\mathsf{EMSO}[\sigma]$ (auch: $\exists\mathsf{MSO}[\sigma]$, $\mathrm{mon}\Sigma^1_1[\sigma]$, $\mathrm{monNP}[\sigma]$) ist die Klasse aller $\mathsf{MSO}[\sigma]$-Formeln, die zugleich $\mathsf{ESO}[\sigma]$-Formeln sind, d.h. aller $\mathsf{SO}[\sigma]$-Formeln der Form $\exists X_1 \ldots \exists X_d\, \varphi$, wobei gilt: $d \in \mathbb{N}$, $X_1, \ldots, X_d$ sind Mengenvariablen, und $\varphi \in \mathsf{FO}[\sigma \cup \{X \in \mathrm{Var}_2 \mid \mathsf{ar}(X) = 1\}]$. Letzteres bedeutet, dass φ ohne (Q2) gebildet wird, und bei Anwendungung von (A3) nur Mengenvariablen genutzt werden.

Folie 41

Beispiele 2.6
Betrachte die Formeln aus Beispiel 2.4:

- $\Phi_{\text{3-col}}$ gehört zu $\mathsf{MSO}[\sigma]$, $\mathsf{ESO}[\sigma]$ und $\mathsf{EMSO}[\sigma]$.
- $\Phi_{\text{non-conn}}$ gehört zu $\mathsf{MSO}[\sigma]$, $\mathsf{ESO}[\sigma]$ und $\mathsf{EMSO}[\sigma]$.
- Φ_{reach} gehört zu $\mathsf{MSO}[\sigma]$, aber nicht zu $\mathsf{ESO}[\sigma]$ und nicht zu $\mathsf{EMSO}[\sigma]$.
- Φ_{Ham} gehört zu $\mathsf{ESO}[\sigma]$, aber nicht zu $\mathsf{MSO}[\sigma]$ und nicht zu $\mathsf{EMSO}[\sigma]$.

2.2 MSO und der Satz von Büchi

Folie 42

Der Satz von Büchi besagt, dass die *regulären Sprachen* genau diejenigen Sprachen sind, die in MSO (bzw. EMSO) beschrieben werden können.

Sei $\Sigma := \{a_1, \ldots, a_\ell\}$ ein endliches, nicht-leeres Alphabet.

Nicht-leere Worte $w \in \Sigma^\star$ repräsentieren wir wie folgt.

Definition 2.7 (Wortstrukturen)

(a) Sei σ_Σ die Signatur bestehend aus:

- einem 2-stelligen Relationssymbol $\leqslant$
- einem 1-stelligen Relationssymbol P_a für jeden Buchstaben $a \in \Sigma$.

(b) Einem endlichen Wort $w_1 \cdots w_n \in \Sigma^+$ der Länge $n \geqslant 1$ (mit $w_i \in \Sigma$ f.a. $i \in \{1, \ldots, n\}$) ordnen wir die σ_Σ-Struktur $\mathcal{A}_w = (A_w, \leqslant^{\mathcal{A}_w}, P_{a_1}^{\mathcal{A}_w}, \ldots, P_{a_\ell}^{\mathcal{A}_w})$ zu, s.d.:

- $A_w = \{1, \ldots, n\}$ ist die Menge aller Positionen in w
- $\leqslant^{\mathcal{A}_w}$ ist die natürliche lineare Ordnung auf A_w
- $P_a^{\mathcal{A}_w} := \{i \in A_w : w_i = a\}$ für jedes $a \in \Sigma$, d.h. $P_a^{\mathcal{A}_w}$ ist die Menge aller Positionen in w, an denen der Buchstabe a steht.

Folie 43

Beispiel 2.8
Für $\Sigma = \{a, b\}$ und $w = aaab$ gilt $\mathcal{A}_w = (A_w, \leqslant^{\mathcal{A}_w}, P_a^{\mathcal{A}_w}, P_b^{\mathcal{A}_w})$ mit

- $A_w = \{1, 2, 3, 4\}$
- $\leqslant^{\mathcal{A}_w}$ ist die natürliche lineare Ordnung auf $\{1, 2, 3, 4\}$
- $P_a^{\mathcal{A}_w} = \{1, 2, 3\}$
- $P_b^{\mathcal{A}_w} = \{4\}$.

Folie 44

Definition 2.9

- Sei $L \subseteq \Sigma^*$ und sei φ ein $\mathsf{MSO}[\sigma_\Sigma]$-Satz. Wir sagen φ *beschreibt* L, falls für jedes nicht-leere Wort $w \in \Sigma^\star$ gilt:

$$w \in L \iff \mathcal{A}_w \models \varphi.$$

- $L \subseteq \Sigma^\star$ heißt MSO-*definierbar* (bzw. EMSO-*definierbar*), falls es einen $\mathsf{MSO}[\sigma_\Sigma]$-Satz (bzw. $\mathsf{EMSO}[\sigma_\Sigma]$-Satz) gibt, der L beschreibt.

Folie 45

Beispiel 2.10
Sei $\Sigma = \{a, b\}$. Die Sprache

$$L_{even} := \{w \in \Sigma^* : |w| \text{ ist gerade}\}$$

wird durch folgenden $\mathsf{EMSO}[\sigma_\Sigma]$-Satz φ_{even} beschrieben:

$$\begin{aligned}\varphi_{\text{even}} \;:=\; \exists X \Big(& \forall x \, \big(min(x) \;\rightarrow\; \neg X(x)\big) \wedge \\ & \forall x \, \big(max(x) \;\rightarrow\; X(x)\big) \wedge \\ & \forall x \, \forall y \, \Big(succ(x,y) \;\rightarrow\; \big(X(x) \;\leftrightarrow\; \neg X(y)\big)\Big)\Big)\end{aligned}$$

mit:
$$\begin{aligned} min(x) &:= \forall z \; x \leqslant z \\ max(x) &:= \forall z \; z \leqslant x \\ succ(x,y) &:= x \leqslant y \;\wedge\; \neg\, x{=}y \;\wedge\; \forall z \, \big(z \leqslant x \;\vee\; y \leqslant z\big) \end{aligned}$$

Idee: X soll genau die geraden Positionen und die letzte Position eines Wortes enthalten, in dessen Wortstruktur die Formel ausgewertet wird.

Die Abkürzungen $min(x)$, $max(x)$ und $succ(x)$ für oben stehende Formeln werden wir im Verlauf dieser Vorlesung immer wieder benutzen.

Folie 46

Theorem 2.11 (Satz von Büchi)
Sei Σ ein endliches, nicht-leeres Alphabet und sei $L \subseteq \Sigma^*$.
Die folgenden Aussagen sind äquivalent:

(a) L ist regulär (d.h. L wird von einem deterministischen endlichen Automaten erkannt).

(b) L ist EMSO-definierbar.

(c) L ist MSO-definierbar.

Der „Satz von Büchi“ ist auch unter dem Namen „Satz von Büchi, Elgot, Trakhtenbrot“ bekannt.

Die Richtung „$(b) \Rightarrow (c)$“ gilt trivialerweise, da $\mathsf{EMSO}[\sigma_\Sigma] \subseteq \mathsf{MSO}[\sigma_\Sigma]$.

Wir beweisen die Richtung „$(a) \Rightarrow (b)$“ in dem folgenden Lemma.

Lemma 2.12
Jede reguläre Sprache $L \subseteq \Sigma^*$ ist EMSO*-definierbar.*

Beweis. Sei $L \subseteq \Sigma^*$ regulär, d.h. L wird von einem deterministischen endlichen Automaten $\mathbb{A} = (Q, \Sigma, q_0, \delta, F)$ erkannt. O.B.d.A. sei

- $Q = \{0, 1, \ldots, m\}$ für ein $m \in \mathbb{N}$,
- $q_0 \in Q$ ist der Startzustand,
- $\delta : Q \times \Sigma \to Q$ ist die Überführungsfunktion,
- $F \subseteq Q$ ist die Menge der akzeptierenden Zustände.

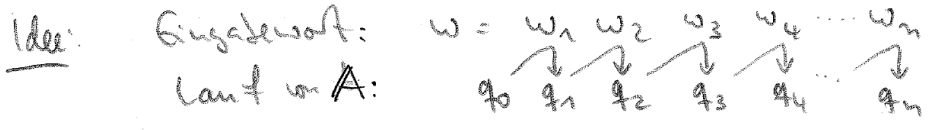


Abbildung 2.1: Lauf von $\mathbb{A}$ auf w

Idee: Für gegebenes $w = w_1 \cdots w_n \in \Sigma^+$ betrachten wir für jeden Zustand $q \in Q$ die Menge Z_q der Positionen i von w, s.d. gilt: Direkt nach dem Lesen des Buchstabens w_i ist $\mathbb{A}$ im Zustand q.
Wir konstruieren einen $\mathsf{EMSO}[\sigma_\Sigma]$-Satz $\Phi_\mathbb{A}$, so dass für alle $w \in \Sigma^+$ gilt:

$$\mathbb{A} \text{ akzeptiert } w \iff \mathcal{A}_w \models \Phi_\mathbb{A}.$$

Dazu wählen wir $\Phi_\mathbb{A}$ wie folgt:

$$\Phi_\mathbb{A} := \exists Z_0 \cdots \exists Z_m \Big(\varphi_{\text{Zustand}} \wedge \varphi_{\text{Start}} \wedge \varphi_{\text{Schritt}} \wedge \varphi_{\text{akzeptiere}} \Big)$$

mit

- φ_{Zustand} besagt (in $\mathcal{A}_w$), dass jede Position von w zu genau einer der Mengen $Z_0, \ldots, Z_m$ gehört:

$$\varphi_{\text{Zustand}} := \forall x \bigvee_{q \in Q} \Big(Z_q(x) \wedge \bigwedge_{q' \in Q \setminus \{q\}} \neg Z_{q'}(x) \Big)$$

- φ_{Start} besagt, dass die Position 1 in $Z_{\delta(q_0,w_1)}$ enthalten ist:

$$\varphi_{\text{Start}} \;:=\; \forall x \Big(min(x) \;\rightarrow\; \bigwedge_{a\in\Sigma} \big(P_a(x) \;\rightarrow\; Z_{\delta(q_0,a)}(x)\big)\Big)$$

- φ_{Schritt} besagt, dass die Mengen $Z_0, \ldots, Z_m$ mit der Überführungsfunktion δ kompatibel sind, d.h. für jede Position $i < n$ und Zustand $q \in Q$ mit $i \in Z_q$ gilt $i{+}1 \in Z_{\delta(q,w_{i+1})}$:

$$\varphi_{\text{Schritt}} \;:=\; \forall x\, \forall y \left(succ(x,y) \;\rightarrow\; \bigwedge_{\substack{a\in\Sigma,\\ q\in Q}} \Big(\big(Z_q(x) \;\wedge\; P_a(y)\big) \;\rightarrow\; Z_{\delta(q,a)}(y)\Big)\right)$$

- $\varphi_{\text{akzeptiere}}$ besagt, dass es einen akzeptierenden Zustand $q \in F$ gibt, s.d. die letzte Position n in Z_q enthalten ist:

$$\varphi_{\text{akzeptiere}} \;:=\; \forall x \Big(max(x) \;\rightarrow\; \bigvee_{q\in F} Z_q(x)\Big).$$

Offenbar ist $\Phi_{\mathbb{A}}$ ein $\mathsf{EMSO}[\sigma_\Sigma]$-Satz.
Man kann leicht nachprüfen, dass für alle $w \in \Sigma^+$ gilt:

$$\mathbb{A} \text{ akzeptiert } w \iff \mathcal{A}_w \models \Phi_{\mathbb{A}}.$$

Somit gilt, dass $\Phi_{\mathbb{A}}$ die von $\mathbb{A}$ akzeptierte reguläre Sprache L beschreibt. Dies beendet den Beweis von Lemma 2.12. □

Die Richtung „$(a) \Rightarrow (b)$“ des Satzes von Büchi gilt gemäß Lemma 2.12.

Folie 47

Die Richtung „$(c) \Rightarrow (a)$“ des Satzes von Büchi (Theorem 2.11) stellt den schwierigsten Teil des Beweises dar. Wir bringen hierfür einen gegebenen $\mathsf{MSO}[\sigma_\Sigma]$-Satz zunächst in eine für die Übersetzung in einen endlichen Automaten besonders geeignete Form.
Seien X und Y Mengenvariablen und $a \in \Sigma$. Wir nutzen die folgenden $\mathsf{MSO}[\sigma_\Sigma]$-Formeln:

- $singl(X)$ besagt $|X| = 1$ („X is a singleton set“):

$$singl(X) \;:=\; \exists y \Big(X(y) \;\wedge\; \forall z \big(X(z) \;\rightarrow\; z{=}y\big)\Big)$$

- $le(X,Y)$ besagt, dass keine Position in Y kleiner als eine Position in X ist:

$$le(X,Y) \;:=\; \forall x\, \forall y \Big(\big(X(x) \;\wedge\; Y(y)\big) \;\rightarrow\; x \leqslant y\Big)$$

- $sub(X,Y)$ besagt $X \subseteq Y$:

$$sub(X,Y) \;:=\; \forall x \big(X(x) \;\to\; Y(x)\big)$$

- $symb_a(X)$ besagt, dass an allen Positionen in X der Buchstabe a steht:

$$symb_a(X) \;:=\; \forall x \big(X(x) \;\to\; P_a(x)\big).$$

Folie 48

Eine gegebene $\mathsf{MSO}[\sigma_\Sigma]$-Formel φ transformieren wir nun in eine „äquivalente“ $\mathsf{MSO}[\sigma_\Sigma]$-Formel φ^*, indem wir jede Individuenvariable x durch eine *neue* Mengenvariable V_x ersetzen und atomare Formeln durch geeignete Kombinationen der Formeln $singl$, le, sub und $symb_a$ ersetzen, s.d. für alle $x, y \in \mathrm{Var}_1$, $a \in \Sigma$, Mengenvariablen $Z \in \mathrm{Var}_2$, Formeln $\psi, \varphi_1, \varphi_2 \in \mathsf{MSO}[\sigma_\Sigma]$, Junktoren $\square \in \{\wedge, \vee, \to, \leftrightarrow\}$ und Quantoren $Q \in \{\exists, \forall\}$ gilt:

- Für $\varphi := x \leqslant y$ ist $\varphi^* := le(V_x, V_y)$.
- Für $\varphi := x{=}y$ ist $\varphi^* := \big(\, le(V_x, V_y) \,\wedge\, le(V_y, V_x) \,\big)$.
- Für $\varphi := P_a(x)$ ist $\varphi^* := symb_a(V_x)$.
- Für $\varphi := Z(x)$ ist $\varphi^* := sub(V_x, Z)$.
- Für $\varphi := \exists x\, \psi$ ist $\varphi^* := \exists V_x \big(singl(V_x) \,\wedge\, \psi^*\big)$.
- Für $\varphi := \forall x\, \psi$ ist $\varphi^* := \forall V_x \big(singl(V_x) \to \psi^*\big)$.
- Für $\varphi := \neg\psi$ ist $\varphi^* := \neg\psi^*$.
- Für $\varphi := (\varphi_1 \,\square\, \varphi_2)$ ist $\varphi^* := (\varphi_1^* \,\square\, \varphi_2^*)$.
- Für $\varphi := Q Z\, \psi$ ist $\varphi^* := Q Z\, \psi^*$.

Lemma 2.13
Für jeden $\mathsf{MSO}[\sigma_\Sigma]$-Satz φ ist auch φ^* ein $\mathsf{MSO}[\sigma_\Sigma]$-Satz, und φ^* beschreibt dieselbe Sprache wie φ.

Beweis. Übung. □

Folie 49

Als nächstes konstruieren wir induktiv für jede Teilformel von φ^* einen „äquivalenten“ endlichen Automaten. Formeln der Form $singl(X)$, $le(X,Y)$, $sub(X,Y)$ und $symb_a(X)$ behandeln wir dabei als „atomar“.

Insbesondere haben die Teilformeln, die wir betrachten müssen, keine freien Individuenvariablen, wohl aber freie Mengenvariablen. Belegungen dieser Mengenvariablen repräsentieren wir durch ein größeres Alphabet:

Seien $X_1, \ldots, X_k$ (mit $k \in \mathbb{N}_{\geqslant 1}$) die Mengenvariablen, die in φ^* vorkommen. Sei

$$\Sigma' \;:=\; \Sigma \times \{0,1\}^k.$$

Ein Wort $w_1 \cdots w_n \in \Sigma^+$ (mit $w_i \in \Sigma$ f.a. $i \in [n]$) und Belegungen $S_1, \ldots, S_k \subseteq \{1, \ldots, n\}$ der Mengenvariablen $X_1, \ldots, X_k$ repräsentieren wir durch das Wort

$$u \;:=\; u_1 \cdots u_n \;\in\; \big(\Sigma'\big)^+$$

mit

$$u_i \;:=\; \big(w_i,\, b_{i,1}, \ldots,\, b_{i,k}\big) \in \Sigma' \text{ für alle } i \in \{1, \ldots, n\},$$

wobei für alle $i \in \{1, \ldots, n\}$ und $j \in \{1, \ldots, k\}$ gilt:

$$b_{i,j} := \begin{cases} 1 & \text{falls } i \in S_j \\ 0 & \text{falls } i \notin S_j. \end{cases}$$

Folie 50

Umgekehrt repräsentiert jedes Wort $u = u_1 \cdots u_n \in \big(\Sigma'\big)^+$ ein Wort

$$w \;=\; w_1 \cdots w_n \;\in\; \Sigma^+$$

und Belegungen

$$S_1, \ldots, S_k \subseteq [n]$$

der Mengenvariablen $X_1, \ldots, X_k$ wie folgt: Ist, für alle $i \in \{1, \ldots, n\}$, $u_i = (\sigma_i,\, b_{i,1}, \ldots, b_{i,k})$, so ist

$$w_i \;=\; \sigma_i \;\text{ für alle } i \in [n]$$

und

$$S_j \;=\; \big\{i \in [n] \,:\, b_{i,j} = 1\big\} \;\text{ für alle } j \in [k].$$

Setze dann $\mathcal{I}_u := \big(\mathcal{A}_w, S_1, \ldots, S_k\big)$.

Eine Teilformel ψ von φ^* *definiert* die Sprache

$$L'(\psi) \;:=\; \big\{u \in \big(\Sigma'\big)^+ \,:\, \mathcal{I}_u \models \psi\big\}.$$

Lemma 2.14
Sei $k \in \mathbb{N}_{\geqslant 1}$, $\Sigma' := \Sigma \times \{0,1\}^k$, seien $i, j \in [k]$, sei $a \in \Sigma$.
Es gibt nichtdeterminstische endliche Automaten $\mathbb{A}_{singl(X_i)}$, $\mathbb{A}_{le(X_i,X_j)}$, $\mathbb{A}_{sub(X_i,X_j)}$ und $\mathbb{A}_{symb_a(X_i)}$ über dem Alphabet Σ', so dass gilt:

(a) $\mathbb{A}_{singl(X_i)}$ erkennt die Sprache $L'\big(singl(X_i)\big)$.

(b) $\mathbb{A}_{le(X_i,X_j)}$ erkennt die Sprache $L'\big(le(X_i,X_j)\big)$.

(c) $\mathbb{A}_{sub(X_i,X_j)}$ erkennt die Sprache $L'\big(sub(X_i,X_j)\big)$.

(d) $\mathbb{A}_{symb_a(X_i)}$ erkennt die Sprache $L'\big(symb_a(X_i)\big)$.

Beweis.
(a) Der Automat $\mathbb{A}_{singl(X_i)}$ soll ein Wort $u = u_1 \cdots u_n \in \big(\Sigma'\big)^+$ genau dann akzeptieren, wenn gilt: Es gibt genau eine Position $p \in \{1, \ldots, n\}$, sodass für den Buchstaben

$$u_p \;=\; (w_p,\, b_{p,1}, \ldots, b_{p,k}) \;\in\; \Sigma \times \{0,1\}^k$$

die zu X_i gehörende Komponente $b_{p,i}$ gleich 1 ist.

Dies wird durch den folgenden nichtdeterministischen Automaten gewährleistet (hierbei bedeutet ein „$*$“, dass in der entsprechenden Komponente jeder mögliche Wert stehen kann):

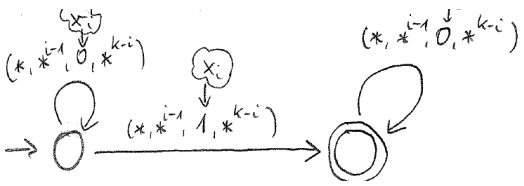


Abbildung 2.2: $\mathbb{A}_{singl(X_i)}$

(b) Übung.

(c) Der Automat $\mathbb{A}_{sub(X_i,X_j)}$ soll ein Wort u genau dann akzeptieren, wenn an jeder Position p für den dortigen Buchstaben $u_p = (w_p, b_{p,1}, \ldots, b_{p,k}) \in \Sigma \times \{0,1\}^k$ gilt:

$$\text{Falls } b_{p,i} = 1, \text{ so auch } b_{p,j} = 1.$$

Dies wird durch den folgenden Automaten gewährleistet. O.B.d.A. betrachten wir hier den Fall, dass $i = 1$ und $j = 2$ ist:

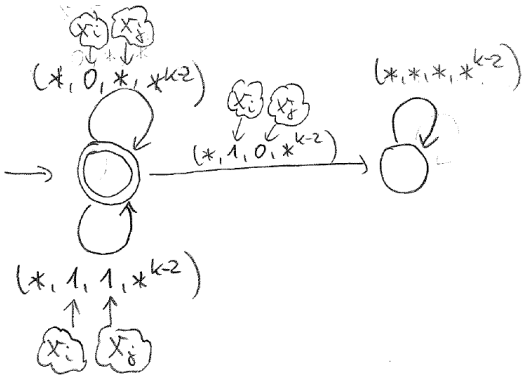


Abbildung 2.3: $\mathbb{A}_{sub(X_i,X_j)}$

(d) Übung.

□

Folie 52

Lemma 2.15
Sei φ ein $\mathsf{MSO}[\sigma_\Sigma]$-Satz, seien $X_1, \ldots, X_k$ (für $k \in \mathbb{N}_{\geqslant 1}$) die in φ^* vorkommenden Mengenvariablen, und sei $\Sigma' := \Sigma \times \{0,1\}^k$.

Für jede Teilformel ψ von φ^* (wobei die Formeln $singl(X_i)$, $sub(X_i, X_j)$, $le(X_i, X_j)$, $symb_a(X_i)$ als „atomar" betrachtet werden) gibt es einen nichtdeterministischen Automaten $\mathbb{A}_\psi$ über dem Alphabet Σ', sodass gilt:

$$\mathbb{A}_\psi \text{ erkennt die Sprache } L'(\psi).$$

Beweis. Wir führen den Beweis induktiv nach dem Aufbau von ψ. Der Induktionsanfang für „atomares" ψ wird durch das Lemma 2.14 gewährleistet.

Induktionsschritt:

- **Fall 1**: $\psi = \neg\chi$ für eine Teilformel χ von φ^*
  Gemäß Induktionsannahme gibt es einen Automaten $\mathbb{A}_\chi$, der die Sprache $L'(\chi)$ erkennt. Es gilt für alle $u \in \big(\Sigma'\big)^+$:

  $$u \in L'(\psi) \iff u \notin L'(\chi)$$

  Also erhalten wir den gesuchten Automaten $\mathbb{A}_\psi$, indem wir $\mathbb{A}_\chi$ zunächst deterministisch machen (Potenzmengenkonstruktion), und dann akzeptierende zu nichtakzeptierenden Zuständen machen und umgekehrt.

- **Fall 2**: $\psi = (\psi_1 \vee \psi_2)$ für Teilformeln ψ_1, ψ_2 von φ^*
  Gemäß Induktionsannahme gibt es Automaten $\mathbb{A}_{\psi_1}$ und $\mathbb{A}_{\psi_2}$, die die Sprachen $L'(\psi_1)$ und $L'(\psi_2)$ erkennen. O.B.d.A. seien deren Zustandsmengen disjunkt. Den nichtdeterministischen Automaten $\mathbb{A}_\psi$ erhalten wir, indem wir die Automaten $\mathbb{A}_{\psi_1}$ und $\mathbb{A}_{\psi_2}$ vereinigen und deren Startzustände mit einem neuen Startzustand identifizieren.
  Das heißt: Ist $\mathbb{A}_{\psi_i} = (Q_i, \Sigma', q_{0,i}, \Delta_i, F_i)$ mit $\Delta_i \subseteq Q_i \times \Sigma_i \times Q_i$ für $i \in \{1,2\}$ und zusätzlich $Q_1 \cap Q_2 = \emptyset$, so ist $\mathbb{A}_\psi = (Q, \Sigma', q_0, \Delta, F)$ mit
  - $Q = Q_1 \,\dot\cup\, Q_2 \,\dot\cup\, \{q_0\}$ für einen neuen Zustand q_0
  - $F = F_1 \cup F_2 \cup M$ mit
    $$M = \begin{cases} \{q_0\} & \text{falls } q_{0,1} \in F_1 \text{ oder } q_{0,2} \in F_2 \\ \emptyset & \text{sonst} \end{cases}$$
  - $\Delta = \Delta_1 \cup \Delta_2 \cup \{(q_0, \sigma', q) : i \in \{1,2\},\ (q_{0,i}, \sigma', q) \in \Delta_i\}$

- **Fall 3**: $\psi = (\psi_1 \,\square\, \psi_2)$ für Teilformeln ψ_1, ψ_2 von φ^*, $\square \in \{\wedge, \to, \leftrightarrow\}$
  Dies folgt aus geeigneten Kombinationen der Fälle 1 und 2, da z.B. $(\psi_1 \to \psi_2)$ äquivalent ist zu $(\neg\psi_1 \vee \psi_2)$. Details: Übung.

- **Fall 4**: $\psi = \exists X_i\, \chi$ für eine Teilformel χ von φ^* und $i \in \{1, \ldots, k\}$.
  Gemäß Induktionsannahme gibt es bereits einen Automaten $\mathbb{A}_\chi$, der die Sprache $L'(\chi)$ erkennt.
  Für jedes $u = u_1 \cdots u_n \in \big(\Sigma'\big)^+$ mit

  $$u_p = (w_p, b_{p,1}, \ldots, b_{p,k}) \in \Sigma \times \{0,1\}^k, \quad \text{für alle } p \in [n]$$

  gilt (Beweis: Übung):

Ⓓ: $u \in L'(\psi) \iff$
Es gibt Werte $\tilde{b}_1, \ldots, \tilde{b}_n \in \{0,1\}$, die für die X_i-Komponente eingesetzt werden können, s.d. für $\tilde{u} := \tilde{u}_1 \cdots \tilde{u}_n$ mit $\tilde{u}_p = (w_p, b_{p,1}, \ldots, b_{p,i-1}, \tilde{b}_p, b_{p,i+1}, \ldots, b_{p,k})$ f.a. $p \in [n]$ gilt, dass

$$\tilde{u} \in L'(\chi) \quad \text{(d.h. } \tilde{u} \text{ wird von } \mathbb{A}_\chi \text{ akzeptiert).}$$

Somit wird die Sprache $L'(\psi)$ durch den nichtdeterminstischen Automaten $\mathbb{A}_\psi$ erkannt, der aus $\mathbb{A}_\chi$ entsteht, indem in der Beschriftung jedes Pfeils der grafischen Darstellung von $\mathbb{A}_\chi$ die X_i-Komponente durch das „Wildcard"-Symbol $*$ ersetzt wird. Das heißt: Ist

$$\begin{aligned}
\mathbb{A}_\chi &= (Q, \Sigma', q_0, \ \Delta, F), \text{ so ist} \\
\mathbb{A}_\psi &= (Q, \Sigma', q_0, \ \tilde{\Delta}, F) \text{ mit} \\
\tilde{\Delta} &= \big\{(q, \tilde{\sigma}, q') \ : (q, \sigma, q') \in \Delta \text{ und } \tilde{\sigma} = (\tilde{a}, \tilde{b}_1, \ldots, \tilde{b}_k) \in \Sigma \times \{0,1\}^k \\
&\qquad \text{entsteht aus } \sigma = (a, b_1, \ldots, b_k) \in \Sigma \times \{0,1\}^k, \\
&\qquad \text{indem der Wert } b_i \text{ durch einen beliebigen Wert} \\
&\qquad \tilde{b}_i \in \{0,1\} \text{ ersetzt wird } \big\}
\end{aligned}$$

Man kann leicht nachprüfen, dass für alle Worte $u \in \big(\Sigma'\big)^+$ gilt:

$$\mathbb{A}_\psi \text{ akzeptiert } u \iff \text{die in Ⓓ nach dem „}\Longleftrightarrow\text{"-Symbol stehende Aussage gilt.}$$

- **Fall 5**: $\psi = \forall X_i \, \chi$ für eine Teilformel χ von φ^* und $i \in \{1, \ldots, k\}$
  Dies folgt durch Kombination der Fälle 1 und 4, da $\forall X_i \, \chi$ äquivalent ist zu $\neg \exists X_i \, \neg \chi$.

Dies beendet den Beweis von Lemma 2.15. □

Folie 53

Das folgende Lemma beweist die Richtung „$(c) \Rightarrow (a)$" des Satzes von Büchi (Theorem 2.11).

Lemma 2.16
Für jeden $\mathsf{MSO}[\sigma_\Sigma]$*-Satz* φ *gibt es einen nichtdeterministischen endlichen Automaten* $\mathbb{A}_\varphi$*, der die von* φ *beschriebene Sprache erkennt.*

Beweis. Wir nutzen Lemma 2.13 und Lemma 2.15, um die Formel φ^* und den Automaten $\mathbb{A}_{\varphi^*}$ zu konstruieren, der die Sprache $L'(\varphi^*) \subseteq \big(\Sigma'\big)^+$ erkennt. Dabei ist $\Sigma' = \Sigma \times \{0,1\}^k$ und k ist so gewählt, dass $X_1, \ldots, X_k$ die in φ^* vorkommenden Mengenvariablen sind.

Das heißt, für jedes Wort $u = u_1 \cdots u_n \in \big(\Sigma'\big)^+$ und die zugehörige Kombination von Wort $w \in \Sigma^+$ und Mengen $S_1, \ldots, S_k \subseteq [n]$ gilt:
$\mathbb{A}_{\varphi^*}$ akzeptiert das Wort u genau dann, wenn $\mathcal{I}_u = (\mathcal{A}_w, S_1, \ldots, S_k)$ die Formel φ^* erfüllt. Da φ^* keine freien Variablen hat (und somit die jeweilige Belegung der Mengenvariablen $X_1, \ldots, X_k$ irrelevant ist), und dieselbe Sprache wie φ beschreibt, gilt

$$\mathcal{I}_u \models \varphi^* \quad \Longleftrightarrow \quad \mathcal{A}_w \models \varphi^* \quad \Longleftrightarrow \quad \mathcal{A}_w \models \varphi.$$

Somit erhalten wir einen nichtdeterministischen Automaten $\mathbb{A}_\varphi$, der genau diejenigen Worte $w \in \Sigma^+$ akzeptiert, für die $\mathcal{A}_w \models \varphi$ gilt, indem wir in der grafischen Darstellung von $\mathbb{A}_{\varphi^*}$ in der Beschriftung jedes Pfeils die Komponenten für $X_1, \ldots, X_k$ weglassen. Das heißt:

$$\begin{aligned}
\text{Ist } \ \mathbb{A}_{\varphi^*} &= (Q, \Sigma', q_0, \Delta, F), \text{ so ist} \\
\mathbb{A}_{\varphi} &= (Q, \Sigma, q_0, \tilde{\Delta}\,, F) \text{ mit} \\
\tilde{\Delta} &= \big\{(q, \sigma, q') : \text{es gibt } b_1, \ldots, b_k \in \{0,1\}, \\
&\qquad\qquad \text{s.d. } \big(q, (\sigma, b_1, \ldots, b_k), q'\big) \in \Delta\big\}.
\end{aligned}$$

Man kann leicht nachprüfen, dass für alle Worte $w \in \Sigma^+$ gilt:

$$\mathbb{A}_\varphi \text{ akzeptiert } w \quad \Longleftrightarrow \quad \mathcal{A}_w \models \varphi.$$

Insbesondere gilt, dass die von φ beschriebene Sprache regulär ist. Dies beendet den Beweis von Lemma 2.16 und damit auch den Beweis des Satzes von Büchi (Theorem 2.11). □

Folie 54

Bemerkung 2.17
Der Satz von Büchi besagt insbesondere, dass existenzielle monadische Logik zweiter Stufe *auf Worten* dieselbe Ausdrucksstärke besitzt wie die „volle“ monadische Logik zweiter Stufe (und genau die regulären Sprachen beschreiben kann).

Man kann sogar beweisen, dass jede reguläre Sprache L durch einen $\mathsf{EMSO}[\sigma_\Sigma]$-Satz beschrieben werden kann, der nur eine Mengenvariable benutzt. Somit ist *auf Worten* die existentielle monadische Logik zweiter Stufe mit nur einer einzigen Mengenvariablen äquivalent zur vollen monadischen Logik zweiter Stufe.

Folie 55

Aus dem Gebiet der formalen Sprachen sind viele Methoden bekannt, mit denen man nachweisen kann, dass bestimmte Sprachen nicht regulär sind (z.B. Pumping Lemma, Satz von Myhill und Nerode, Abschlusseigenschaften, ...). Man siehe dazu einführende Vorlesungen in Theoretische Informatik, formale Sprachen bzw. Automatentheorie.

Insbesondere wissen wir für $\Sigma = \{a, b\}$, dass die Sprache

$$L_{a^n b^n} := \{a^n b^n \,:\, n \in \mathbb{N}_{\geqslant 1}\}$$

nicht regulär ist.

Gemäß dem Satz von Büchi kann $L_{a^n b^n}$ also auch nicht durch einen MSO-Satz beschrieben werden.

Durch Anwenden einer *logischen Reduktion* können wir daraus folgern, dass z.B. bestimmte Graphen-Eigenschaften nicht in MSO beschrieben werden können:

Satz 2.18 („Hamiltonkreis ist nicht MSO-definierbar")
Sei $\sigma_{\text{Graph}} = \{E\}$ für ein 2-stelliges Relationssymbol E.
Es gibt keinen $\mathsf{MSO}[\sigma_{\mathsf{Graph}}]$-Satz φ, s.d. für jeden endlichen Graphen $\mathcal{G}$ gilt:

$$\mathcal{G} \models \varphi \iff \mathcal{G} \text{ besitzt einen Hamiltonkreis.}$$

Beweis. Durch Widerspruch: Angenommen, φ_{Ham} ist ein $\mathsf{MSO}[\sigma_{\mathsf{Graph}}]$-Satz, s.d. für jeden endlichen Graphen $\mathcal{G}$ gilt:

$$\mathcal{G} \models \varphi_{\text{Ham}} \iff \mathcal{G} \text{ besitzt einen Hamiltonkreis.}$$

Ziel: Modifiziere φ_{Ham} zu einem $\mathsf{MSO}[\sigma_\Sigma]$-Satz $\varphi_{a^n b^n}$ (für $\Sigma = \{a, b\}$), der die Sprache $L_{a^n b^n} = \{a^n b^n \,:\, n \in \mathbb{N}_{\geqslant 1}\}$ beschreibt.

Idee: Einem Wort $w = w_1 \cdots w_n \in \{a, b\}^+$ ordnen wir den Graphen $\mathcal{G}_w = (V_w, E_w)$ zu mit

- V_w als Menge der Positionen von w
- $E_w := \{(i, j) \in V_w \times V_w \,:\, w_i = a \;\Leftrightarrow\; w_j = b\}$,
  d.h. zwischen zwei Positionen i und j gibt es genau dann eine Kante, wenn in w an diesen Positionen unterschiedliche Buchstaben stehen.

Beachte: $\mathcal{G}_w$ ist ein vollständig bipartiter Graph auf Partition $P_a^{\mathcal{A}_w} \dot\cup P_b^{\mathcal{A}_w}$. Insbesondere gilt:

$$\begin{aligned} & \mathcal{G}_w \text{ besitzt einen Hamiltonkreis} \\ \iff \quad & |P_a^{\mathcal{A}_w}| = |P_b^{\mathcal{A}_w}|, \text{ d.h. } w \text{ enthält genauso viele } a\text{'s wie } b\text{'s} \end{aligned}$$

Daher wird $L_{a^n b^n}$ durch folgenden $\mathsf{MSO}[\sigma_\Sigma]$-Satz $\varphi_{a^n b^n}$ beschrieben:

$$\varphi_{a^n b^n} \;:=\; \Big(\forall x \, \forall y \, \Big(\big(P_a(x) \,\wedge P_b(y)\big) \;\rightarrow\; x \leqslant y \Big) \;\wedge\; \varphi'_{\text{Ham}} \Big),$$

wobei φ'_{Ham} aus φ_{Ham} entsteht, indem jedes Atom der Form $E(x,y)$ ersetzt wird durch die Formel

$$\big(P_a(x) \;\leftrightarrow\; P_b(y)\big).$$

Dies ist ein Widerspruch, da gemäß dem Satz von Büchi die Sprache $L_{a^n b^n}$ dann regulär sein müsste. Dies beendet den Beweis von Satz 2.18. □

2.3 ESO und der Satz von Fagin

Folie 56

Zur Erinnerung:
ESO-Formeln sind SO-Formeln der Form

$$\exists X_1 \cdots \exists X_\ell \, \varphi,$$

wobei $\ell \geqslant 0$ ist, $X_1, \ldots, X_\ell$ Relationsvariablen beliebiger Stelligkeit sind und φ eine FO-Formel ist, deren atomare Teilformeln auch Relationsvariablen nutzen können.

Definition 2.19 (Logische Beschreibung von Komplexitätsklassen)
Sei K eine Komplexitätsklasse (z.B. NP), sei $\mathcal{L}$ eine Logik (z.B. ESO) und sei $\mathcal{S}$ eine unter Isomorphie abgeschlossene Klasse endlicher Strukturen (z.B. FIN := die Klasse aller endlichen Strukturen über allen endlichen funktionenfreien Signaturen).

Wir sagen „$\mathcal{L}$ *beschreibt* K *auf* $\mathcal{S}$“, falls die folgenden beiden Bedingungen erfüllt sind:

Folie 57

(1) Für jeden Satz $\varphi \in \mathcal{L}$ gehört das folgende Problem zur Komplexitätsklasse K:

$\mathit{EVAL}_\varphi(\mathcal{S})$ *(Das Auswertungsproblem für* φ *auf* $\mathcal{S}$*)*
Eingabe: Eine Struktur $\mathcal{A} \in \mathcal{S}$
Frage: Gilt $\mathcal{A} \models \varphi$?

(2) Für jede endliche, funktionenfreie Signatur σ und jede unter Isomorphie abgeschlossene Klasse $\mathcal{C} \subseteq \mathcal{S}$ von σ-Strukturen gilt:
Falls das Problem

Zugehörigkeit zu $\mathcal{C}$ *in* $\mathcal{S}$
Eingabe: Eine σ-Struktur $\mathcal{A} \in \mathcal{S}$
Frage: Ist $\mathcal{A} \in \mathcal{C}$?

zur Komplexitätsklasse K gehört,
so gibt es einen $\mathcal{L}[\sigma]$-Satz φ, sodass gilt:

$$\mathcal{C} = \underbrace{\{\mathcal{A} \in \mathcal{S} : \mathcal{A} \text{ ist eine } \sigma\text{-Struktur mit } \mathcal{A} \models \varphi\}}_{=: \mathrm{Mod}_{\mathcal{S}}(\varphi)}$$

Folie 58

Theorem 2.20 (Der Satz von Fagin, 1974)
ESO *beschreibt* NP *auf der Klasse* FIN *aller endlicher Strukturen.*

Der Beweis von Theorem 2.20 erfolgt in zwei Teilen:

Im ersten Teil zeigen wir, dass jeder ESO-Satz Ψ bei Eingabe einer Struktur $\mathcal{A}$ nichtdeterministisch in Zeit polynomiell in der Größe von $\mathcal{A}$ ausgewertet werden kann (dies ist der „leichte“ Teil des Beweises).

Im zweiten Teil zeigen wir, dass jedes Problem, das in NP liegt, durch einen ESO-Satz beschrieben werden kann.

Das Berechnungsmodell, mit dem wir arbeiten, sind Turingmaschinen. Um die Details des Beweises von Theorem 2.20 ausarbeiten zu können, müssen wir festlegen, wie eine Struktur $\mathcal{A}$ als Eingabe einer Turingmaschine repräsentiert wird. Wir benutzen dazu die sogenannte *Standardkodierung*, die im Folgenden eingeführt wird.

Folie 59

Definition 2.21
Sei $A = \{a_0, a_1, \ldots, a_{n-1}\}$ eine durch die Relation $<$ linear geordnete endliche Menge, so dass gilt: $a_0 < a_1 < \cdots < a_{n-1}$.

(a) Der *Rang* $\mathrm{rg}_<(a)$ eines Elements $a \in A$ ist

$$\mathrm{rg}_<(a) \;:=\; |\{b \in A \;:\; b < a\}|.$$

Somit gilt $\mathrm{rg}_<(a_i) = i$ (f.a. $i \in \{0, \ldots, n{-}1\}$).

(b) Sei $r \in \mathbb{N}_{\geqslant 1}$. Die *lexikographische Ordnung* $<_{\mathrm{lex}}$ auf A^r ist wie folgt definiert:
Für $\overline{b} = (b_1, \ldots, b_r) \in A^r$ und $\overline{c} = (c_1, \ldots, c_r) \in A^r$ ist

$$\overline{b} <_{\mathrm{lex}} \overline{c} \iff \text{ex. } i \in \{1, \ldots, r\} \text{ s.d. } b_i < c_i \text{ und f.a. } j < i \text{ ist } b_j = c_j.$$

Man kann sich leicht davon überzeugen, dass $<_{\mathrm{lex}}$ eine strikte lineare Ordnung auf A^r ist .

Folie 60

Beispiel 2.22
Sei $A = \{0, 1, 2\}$ mit $0 < 1 < 2$. Bezüglich der lexikographischen Ordnung $<_{\mathrm{lex}}$ auf A^2 gilt f.a. $\overline{b} \in A^2$:

$\overline{b}$	(0,0)	(0,1)	(0,2)	(1,0)	(1,1)	(1,2)	(2,0)	(2,1)	(2,2)
$\mathrm{rg}_{<_{\mathrm{lex}}}(\overline{b})$	0	1	2	3	4	5	6	7	8

Allgemein gilt: Ist $n \geqslant 1$, $A := \{0, \ldots, n{-}1\}$ mit $0 < \cdots < n{-}1$ und $r \geqslant 1$, so gilt für jedes $\overline{b} = (b_{r-1}, \ldots, b_0) \in A^r$

$$\mathrm{rg}_{<_{\mathrm{lex}}}(\overline{b}) \;=\; \sum_{i=0}^{r-1} b_i n^i \;=\; b_{r-1}n^{r-1} + b_{r-2}n^{r-2} + \cdots + b_0 n^0,$$

d.h. für $n \geqslant 2$ ist $\mathrm{rg}_{<_{\mathrm{lex}}}(\overline{b})$ die von der Darstellung $b_{r-1} \cdots b_0$ zur Basis n repräsentierte natürliche Zahl.

Folie 61

Definition 2.23 (Standardkodierung $\mathrm{enc}_<(\mathcal{A})$)

- Sei $\sigma = \{R_1, \ldots, R_\ell, c_1, \ldots, c_{\ell'}\}$ (mit $\ell, \ell' \geqslant 0$) eine endliche funktionenfreie Signatur, wobei für jedes $j \in \{1, \ldots, \ell\}$ gilt: R_j ist ein Relationssymbol der Stelligkeit $r_j := \mathrm{ar}(R_j)$.
  Sei $r := \max\{1, r_1, \ldots, r_\ell\}$ — d.h. wenn σ mindestens ein Relationssymbol enthält, ist r einfach die maximale Stelligkeit der Relationssymbole in σ, und ansonsten ist $r = 1$.

- Sei $\mathcal{A}$ eine endliche σ-Struktur, sei $n := |A|$. Sei $<$ eine beliebige lineare Ordnung auf A und sei $A = \{a_0, \ldots, a_{n-1}\}$ mit $a_0 < a_1 < \cdots < a_{n-1}$.
  Im Folgenden identifizieren wir jedes $a_i \in A$ mit seinem Rang $i = \mathrm{rg}_<(a_i)$, und für $j \in \{1, \ldots, \ell\}$ identifizieren wir jedes r_j-Tupel $\bar{b} \in A^{r_j}$ mit seinem Rang $\mathrm{rg}_{<_{\mathrm{lex}}}(\bar{b}) \in \{0, \ldots, n^{r_j} - 1\}$.

Folie 62

- Für jedes $j \in \{1, \ldots, \ell\}$ kodieren wir die Relation $R_j^{\mathcal{A}}$ durch das Wort

$$\mathrm{enc}_<(R_j^{\mathcal{A}}) \quad := \quad w_0\, w_1 \cdots w_{n^r-1} \quad \in \quad \{0,1\}^{(n^r)},$$

  wobei
  - f.a. $\bar{b} \in A^{r_j}$ gilt: $\bar{b} \in R_j^{\mathcal{A}} \iff w_{\mathrm{rg}_{<_{\mathrm{lex}}}(\bar{b})} = 1$
  - f.a. $m \geqslant n^{r_j}$ gilt: $w_m = 0$.

- Für jedes $j \in \{1, \ldots, \ell'\}$ kodieren wir die Konstante $c_j^{\mathcal{A}}$ durch das Wort

$$\mathrm{enc}_<(c_j^{\mathcal{A}}) \quad := \quad w_0 w_1 \ldots w_{n^r-1} \quad \in \quad \{0,1\}^{(n^r)},$$

  wobei
  - f.a. $m \in \{0, \ldots, n^r - 1\}$ gilt: $w_m = 1 \iff m = \mathrm{rg}_<(c_j^{\mathcal{A}})$.

- Die Mächtigkeit n des Universums von $\mathcal{A}$ kodieren wir durch das Wort

$$\mathrm{enc}_<(|A|) \quad := \quad 1^n 0^{n^r-n} \quad \in \quad \{0,1\}^{(n^r)}.$$

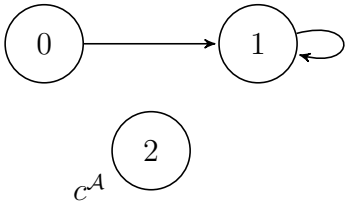
- Die Standardkodierung von $\mathcal{A}$ bzgl. $<$ ist das Wort

$$\mathrm{enc}_<(\mathcal{A}) \;:=\; \mathrm{enc}_<(|A|)\, \mathrm{enc}_<(R_1^{\mathcal{A}}) \cdots \mathrm{enc}_<(R_\ell^{\mathcal{A}})\, \mathrm{enc}_<(c_1^{\mathcal{A}}) \cdots \mathrm{enc}_<(c_{\ell'}^{\mathcal{A}}).$$

  Es gilt: $\mathrm{enc}_<(\mathcal{A}) \in \{0,1\}^{(1+\ell+\ell') \cdot n^r}$.

Folie 63

Beispiel 2.24
Sei $\sigma = \{E, c\}$ die Signatur, die aus einem 2-stelligen Relationssymbol E und einem Konstantensymbol c besteht.
Wir betrachten die σ-Struktur $\mathcal{A}$:



d.h. $\mathcal{A} = (A,\, E^{\mathcal{A}},\, c^{\mathcal{A}})$ mit $A = \{0,1,2\}$, $E^{\mathcal{A}} = \{(0,1),\,(1,1)\}$ und $c^{\mathcal{A}} = 2$.
Es gilt $r = 2$, $n = 3$ und somit $n^r = 9$.
Sei $<$ die natürliche lineare Ordnung auf A. Dann gilt

$$\mathrm{enc}_<(\mathcal{A}) \;=\; \underbrace{111|000000}_{\mathrm{enc}_<(|A|)}\ \underbrace{010|010|000}_{\mathrm{enc}_<(E^{\mathcal{A}})}\ \underbrace{001|000000}_{\mathrm{enc}_<(c^{\mathcal{A}})}.$$

Beachte: $\mathrm{enc}_<(E^{\mathcal{A}})$ ist die zeilenweise gelesene Adjazenzmatrix des Graphen $(A,\, E^{\mathcal{A}})$:

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Folie 64

Bemerkung 2.25

(a) Die Standardkodierung $\mathrm{enc}_<(\mathcal{A})$ hängt von der gewählten linearen Ordnung $<$ ab.

(b) Es gilt:
$$|\mathrm{enc}_<(\mathcal{A})| \;=\; (1+\ell+\ell')\cdot|A|^r.$$
Insbes. ist für jede feste Signatur σ also die Länge von $\mathrm{enc}_<(\mathcal{A})$ polynomiell in der Mächtigkeit $|A|$ der Struktur $\mathcal{A}$.

(c) Sei
$$\mathrm{enc}_<(\mathcal{A}) \;=\; w_0 w_1 \ldots w_{(1+\ell+\ell')\cdot n^r-1}$$
mit $n := |A|$ und $w_i \in \{0,1\}$ f.a. $i < n^r$.
Für $j \in \{1,\ldots,\ell\}$ und ein Tupel $\bar{b} \in A^{r_j}$ lässt sich die Information, ob $\bar{b} \in R_j^{\mathcal{A}}$ ist, in $\mathrm{enc}_<(\mathcal{A})$ am Buchstaben w_m ablesen, für
$$m \;:=\; j\cdot n^r \;+\; \mathrm{rg}_{<_{\mathrm{lex}}}(\bar{b}).$$

Wir können nun den Beweis des Satzes von Fagin (Theorem 2.20) führen.

Beweis von Theorem 2.20 (Satz von Fagin).

Folie 65

Unser Ziel ist es zu zeigen, dass ESO die Klasse NP auf *FIN* beschreibt.

Wir führen den Beweis in zwei Schritten.

1. Schritt: Sei σ eine Signatur und sei $\Psi := \exists X_1 \cdots \exists X_d \, \varphi$ ein $\mathsf{ESO}[\sigma]$-Satz. Wir müssen zeigen, dass das folgende Problem zu NP gehört:

$\mathit{EVAL}_\Psi(\mathit{FIN})$: *Auswertungsproblem für* Ψ *auf* *FIN*
Eingabe: Eine endliche σ-Struktur $\mathcal{A}$
Frage: Gilt $\mathcal{A} \models \Psi$?

Ein nichtdeterministischer Polynomialzeit-Algorithmus kann bei Eingabe einer endlichen σ-Struktur $\mathcal{A}$ wie folgt vorgehen, um zu testen, ob $\mathcal{A} \models \Psi$:

(1) Wenn $d > 0$: Rate Belegungen der Relationsvariablen $X_1, \ldots, X_d$, d.h. rate Relationen $S_1 \subseteq A^{\mathrm{ar}(X_1)}, \ldots, S_d \subseteq A^{\mathrm{ar}(X_d)}$.

Das geht in Zeit $\mathcal{O}(d \cdot n^r)$, wobei $n = |A|$ und r die maximale Stelligkeit der Relationsvariablen ist.

(2) Teste (deterministisch, in Polynomialzeit), ob $(\mathcal{A}, S_1, \ldots, S_d) \models \varphi$.

Dafür können wir den naiven Algorithmus benutzen, der rekursiv entlang der Definition der Semantik von FO vorgeht; siehe Vorlesung „Logik in der Informatik“.

Folie 66

2. Schritt: Sei $\sigma = \{R_1, \ldots, R_\ell, c_1, \ldots, c_{\ell'}\}$ eine Signatur und sei $\mathcal{C}$ eine unter Isomorphie abgeschlossene Klasse endlicher σ-Strukturen, sodass das Problem

Zugehörigkeit zu $\mathcal{C}$ *in* *FIN*
Eingabe: Eine endliche Struktur $\mathcal{A}$.
Frage: Ist $\mathcal{A} \in \mathcal{C}$?

in NP liegt.

Das heißt, es gibt eine NTM

$$M \quad = \quad (Q, \Sigma, \Gamma, \Delta, q_0, F)$$

mit $F = F_{\text{akz}} \,\dot\cup\, F_{\text{verw}}$ und eine Konstante $k \in \mathbb{N}$, s.d. M bei Eingabe (der Kodierung) einer endlichen σ-Struktur $\mathcal{A}$ entscheidet, ob $\mathcal{A} \in \mathcal{C}$ ist und dabei weniger als n^k Schritte macht, für $n := |A|$.

D.h., für jede endliche σ-Struktur $\mathcal{A}$ und jede lineare Ordnung $<$ auf A gilt:

- Jeder Lauf von M bei Eingabe $\text{enc}_<(\mathcal{A})$ endet nach weniger als n^k Schritten, und
- $\mathcal{A} \in \mathcal{C} \iff$
  es gibt einen akzeptierenden Lauf von M bei Eingabe $\text{enc}_<(\mathcal{A})$.