Kapitel 6

Die Unentscheidbarkeit der Logik der 1. Stufe

Kontext und Ziele

In diesem Kapitel werden wir zeigen, dass das Erfüllbarkeitsproblem für die Logik der 1. Stufe unentscheidbar ist.

6.1 Wiederholung: Grundlagen der Berechenbarkeitstheorie

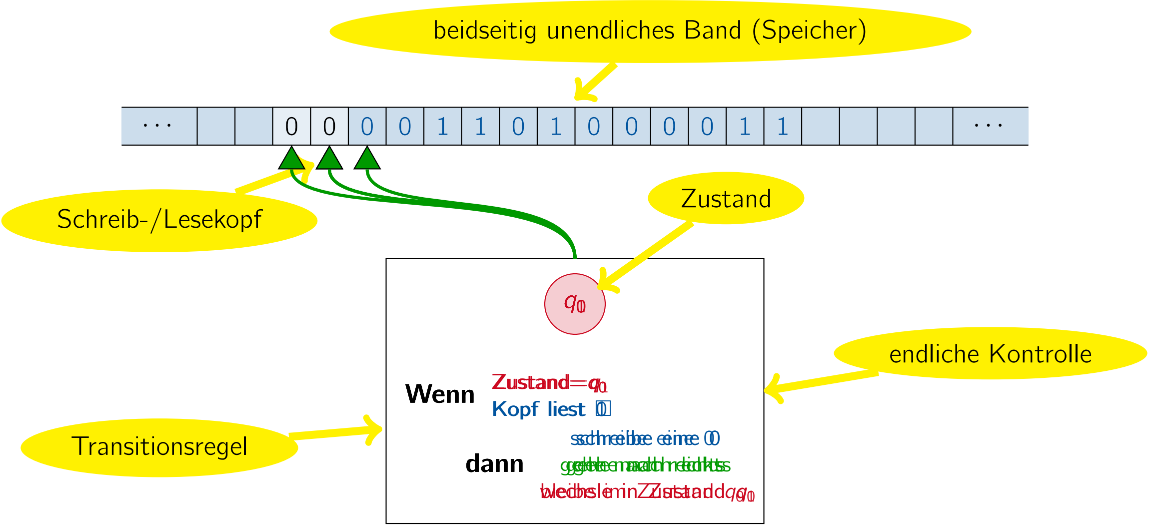
Entscheidungsprobleme

- In der Berechenbarkeitstheorie werden Ein- und Ausgaben von Berechnungsproblemen als Wörter über einem endlichen Alphabet kodiert.
- Wir betrachten in diesem Kapitel nur Entscheidungsprobleme, also Probleme mit Ja/Nein-Antwort. Diese modellieren wir als Sprache (Menge von Wörtern) $L \subseteq \Sigma^*$ über dem Eingabealphabet Σ: Eingabewörter $w \in L$ liefern die Antwort „ja“, Eingabewörter $w \in \Sigma^* \setminus L$ außerhalb der Sprache die Antwort „nein“.

  Ein Sprache $L \subseteq \Sigma^*$ entspricht also dem Einscheidungsproblem:

L	
Eingabe:	$w \in \Sigma^*$.
Frage:	Ist $w \in L$?

Turingmaschinen



Definition 6.1

Eine (deterministische) Turingmaschine (kurz: TM) ist ein Tupel $(Q, \Gamma, \Sigma, \delta)$ bestehend aus:

- einer endlichen Menge Q von Zuständen, die die drei ausgezeichneten Zustände q_0, q_+, q_- enthält: den Anfangszustand q_0, den akzeptierenden Endzustand q_+ und den verwerfenden Endzustand q_-;
- einem endlichen Alphabet Γ, dem Bandalphabet, das das ausgezeichnete Symbol $\Box$, das Leerzeichen, enthält;
- einem endlichen Alphabet $\Sigma \subseteq \Gamma \setminus \{\Box\}$, dem Eingabealphabet;
- einer Transitionsfunktion

$$\delta : (Q \setminus \{q_+, q_-\}) \times \Gamma \to Q \times \Gamma \times \{-1, 0, +1\}.$$

Bemerkung 6.2

Man beachte, dass bei dieser Definition der Anfangszustand *jeder* TM q_0 ist, genauso die Endzustände q_+, q_- und das Leerzeichen $\Box$.

Konfigurationen

Konfigurationen eine Turingmaschinen beschreiben vollständig den Stand einer Berechnung, also den Zustand der endlichen Kontrolle, die Kopfposition und die Bandbeschriftung.

Wir numerieren dazu die Bandzellen mit ganzen Zahlen. Am Anfang steht der Kopf auf Zelle 1.

Menge aller Funktion $\mathbb{Z} \to \Gamma$

Definition 6.3

Sei $M = (Q, \Gamma, \Sigma, \delta)$ eine TM. Eine Konfiguration von M ist ein Tripel

$$(q, p, \beta) \in Q \times \mathbb{Z} \times \Gamma^{\mathbb{Z}}\Gamma^{\mathbb{Z}}.$$

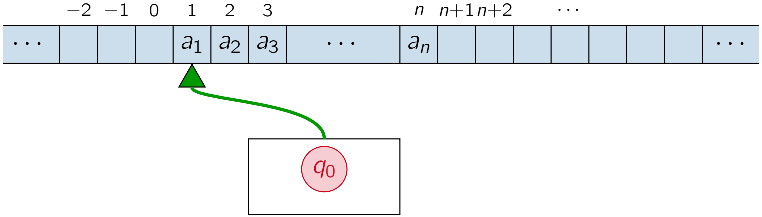
Bemerkung 6.4

Formal haben wir hier Konfigurationen als unendliche Objekte definiert, weil die Bandbeschriftung ja unendlich ist. Allerdings treten während einer Berechnung (mit endlicher Eingabe) immer nur Konfigurationen auf, bei denen nur endlich viele Bandzellen mit einem Symbol $\neq \Box$ beschriftet sind. Solche Beschriftungen können wir leicht endlich beschreiben.

Definition 6.5

Sei $M = (Q, \Gamma, \Sigma, \delta)$ eine TM.

(1) Die Anfangskonfiguration von M bei Eingabe $w = a_1 \dots a_n \in \Sigma^*$ ist $(q_0, 1, \beta_w)$, wobei $\beta_w : \mathbb{Z} \to \Gamma$ definiert ist durch $\beta(i) := a_i$ für alle $i \in [n]$ und $\beta(i) = \Box$ für alle $i \in \mathbb{Z} \setminus [n]$.



(2) Eine Endkonfiguration von M ist eine Konfiguration (q, p, β) mit $q \in \{q_+, q_-\}$. Falls $q = q_+$ handelt es sich um eine akzeptierende Endkonfiguration, und falls $q = q_-$ handelt es sich um eine verwerfende Endkonfiguration.

Notation

Sei $\beta : \mathbb{Z} \to \Gamma$ und $p \in \mathbb{Z}, c \in \Gamma$. Dann ist $\beta\frac{c}{p} : \mathbb{Z} \to \Gamma$ definiert durch $\beta\frac{c}{p}(i) := c$ für $i = p$ und $\beta\frac{c}{p}(i) := \beta(i)$ für $i \in \mathbb{Z} \setminus \{p\}$.

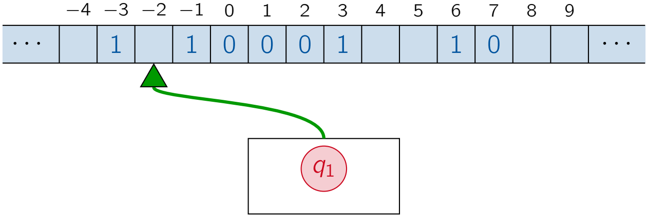
Definition 6.6

Sei $M = (Q, \Gamma, \Sigma, \delta)$ eine TM und $\kappa := (q, p, \beta)$ eine Konfiguration mit $q \notin \{q_+, q_-\}$ (also keine Endkonfiguration). Die Nachfolgekonfiguration von κ ist die Konfiguration

$$\kappa' := \left(q', p + d, \beta\frac{c}{p}\right)$$

falls $\delta(q, \beta(p)) = (q', c, d)$. Wir schreiben $\kappa \to \kappa'$, um auszudrücken, dass κ' die Nachfolgerkonfuguration von κ ist.

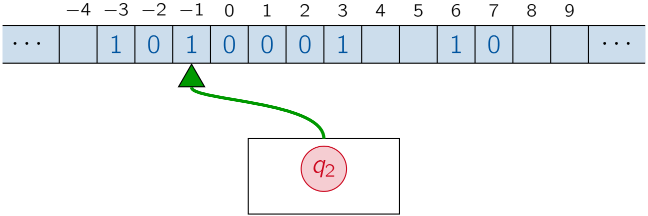
Beispiel 6.7



Die TM befindet sich in der Konfiguration $(q_1, -2, \beta_1)$ mit

$$\beta_1(i) := \begin{cases} 1 & \text{falls } i \in \{-3, -1, 3, 6\}, \\ 0 & \text{falls } i \in \{0, 1, 2, 7\}, \\ \Box & \text{sonst.} \end{cases}$$

Sei $\delta(q_1, \Box) = (q_2, 0, +1)$.



Dann ist die Nachfolgekonfiguration $(q_2, -1, \beta_2)$ mit

$$\beta_2(i) := \begin{cases} 1 & \text{falls } i \in \{-3, -1, 3, 6\}, \\ 0 & \text{falls } i \in \{-2, 0, 1, 2, 7\}, \\ \Box & \text{sonst.} \end{cases}$$

Definition 6.8

Sei $M = (Q, \Gamma, \Sigma, \delta)$ eine TM und $w \in \Sigma^*$. Der Lauf von M bei Eingabe w ist

- entweder die endliche Konfigurationsfolge

  $$\kappa_0 \to \kappa_1 \to \ldots \to \kappa_n,$$

  wobei $\kappa_0 = \kappa_0(M, w)$ die Anfangskonfiguration von M bei Eingabe w ist, κ_i Nachfolgerkonfiguration von κ_{i-1} ist für alle $i \in [n]$ und κ_n eine Endkonfiguration ist,

- oder die unendliche Konfigurationsfolge

  $$\kappa_0 \to \kappa_1 \to \kappa_2 \to \ldots,$$

  wobei $\kappa_0 = \kappa_0(M, w)$ die Anfangskonfiguration von M bei Eingabe w ist und κ_i Nachfolgerkonfiguration von κ_{i-1} ist für alle $i \in \mathbb{N}_{>0}$.

Definition 6.9

Sei $M = (Q, \Gamma, \Sigma, \delta)$ eine TM.

(1) M hält bei Eingabe $w \in \Sigma^*$, wenn der Lauf von M bei Eingabe w endlich ist.

(2) M akzeptiert $w \in \Sigma^*$, wenn der Lauf von M bei Eingabe w endlich ist und in einer akzeptierenden Endkonfiguration endet.

(3) Die von M akzeptierte Sprache ist die Sprache

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

Ergänzungen zu Seite 6.12

Wir könnten noch folgende Definition hinzufügen: M verwirft $w \in \Sigma^*$, wenn der Lauf von M bei Eingabe w endlich ist und in einer verwerfenden Endkonfiguration endet.

Allerdings benötigen wir diese Definition hier nicht.

Semi-Entscheidbare und Entscheidbare Sprachen

Definition 6.10

Eine Sprache $L \subseteq \Sigma^*$ ist semi-entscheidbar, wenn es eine TM M gibt, so dass $L = L(M)$.

Definition 6.11

Eine Sprache $L \subseteq \Sigma^*$ ist entscheidbar, wenn es eine TM M gibt, so dass

(i) $L = L(M)$ und

(ii) M hält bei jeder Eingabe $w \in \Sigma^*$.

In diesem Fall sagen wir, M entscheidet L.

Beobachtung 6.12

Eine TM entscheidet eine Sprache, und zwar die Sprache $L(M)$, falls M bei jeder Eingabe hält.

Ergänzungen zu Seite 6.13

Nicht jede TM hält bei jeder oder auch nur bei irgendeiner Eingabe.

Die Church-Turing These

Church-Turing These

Der durch Turingmaschinen formalisierte Berechenbarkeitsbegriff stimmt mit dem intuitiven Berechenbarkeitsbegriff überein.

Insbesondere ist eine Sprache genau dann entscheidbar mit einer Turingmaschine, wenn es einen Algorithmus (im intuitiven Sinn) gibt, der das zugehörigen Entscheidungsproblem löst.

Wie schon die ganze Vorlesung verwenden wir jetzt wieder einen intuitiven Begriff von Algorithmen und Berechenbarkeit. Wir kodieren auch Entscheidungsprobleme nicht mehr formal als Sprache über einem endlichen Alphabet, sondern beschreiben sie in natürlicher Sprache.

Das Halteproblem

Das allgemeine Halteproblem:

H

Eingabe: TM M, Wort $w \in \Sigma^*$.

Frage: Hält M bei Eingabe w?

Das spezielle Halteproblem:

H_ε

Eingabe: TM M.

Frage: Hält M bei Eingabe ε?

Die Unentscheidbarkeit des Halteproblems

Satz 6.13

H *und* H_ε *sind unentscheidbar.*

Ergänzungen zu Seite 6.16

Der Satz wird in der Vorlesung Berechenbarkeit und Komplexität beweisen.

Das Komplement eines Entscheidungsproblems P ist das Entscheidungsproblem $\overline{P}$, das man aus P erhält, wenn man ja-Instanzen und nein-Instanzen vertauscht.

Beispiel 6.14

Das Komplement $\overline{\textsc{Erf}(\mathsf{L}(\sigma))}$ des Erfüllbarkeitsproblems ist das Unerfüllbarkeitsproblem:

$\textsc{UnErf}(\mathrm{L}(\sigma))$

Eingabe:	Formel $\varphi \in \mathsf{L}(\sigma)$.
Frage:	Ist φ unerfüllbar?

Satz 6.15

Ein Entscheidungsproblem P ist genau dann entscheidbar, wenn P und $\overline{P}$ semi-entscheidbar sind.

Ergänzungen zu Seite 6.17

Zur Erinnerung

Eine Entscheidungsproblem is semi-entscheidbar, wenn es eine Algorithmus gibt, der bei Eingabe einer ja-Instanz mit Ausgabe „ja“ anhält und bei Eingabe einer nein-Instanz entweder mit Ausgabe „nein“ anhält oder nicht anhält.

Der Satz wird in der Vorlesung Berechenbarkeit und Komplexität beweisen.

Semi-Entscheidbarkeit der Halteprobleme

Satz 6.16

H *und* H_ε *sind semi-entscheidbar.*

6.2 Die Unentscheidbarkeit des Erfüllbarkeitsproblems

Unentscheidbarkeit des Erfüllbarkeitsproblems

Satz 6.17

Es gibt eine endliche Symbolmenge σ_{UE}*, so dass* $\textsc{Erf}(\mathsf{L}(\sigma_{\mathsf{UE}}))$ *unentscheidbar ist.*

Tatsächlich ist schon $\textsc{Erf}(\mathsf{L}(\{E\}))$ für die Symbolmenge, die aus einem einzigen zweistelligen Relationssymbol E besteht, unentscheidbar. Das beweisen wir aber in dieser Vorlesung nicht.

Beweisstrategie

- Wir reduzieren $\overline{\mathrm{H}_\varepsilon}$ auf $\textsc{Erf}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$ für eine geeignete Symbolmenge σ_{UE}.
- Dazu müssen wir für jede TM M einen σ_{UE}-Formel φ_M konstruieren, so dass

  $$M \text{ hält bei Eingabe } \varepsilon \text{ nicht} \iff \varphi_M \text{ ist erfüllbar.}$$

  Weiterhin muss die Abbildung $M \mapsto \varphi_M$ berechenbar sein.
- Wäre dann $\textsc{Erf}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$ entscheidbar, so wäre auch $\overline{\mathrm{H}_\varepsilon}$ und damit H_ε entscheidbar. Das *widerspricht* Satz 6.13.

Die Symbolmenge

Sei

$$\sigma_{\mathrm{UE}} := \{\dot{\leq}, g_{+1}, g_{-1}, f_q, f_p, f_\beta, \dot{0}, c_0, c_+, c_-, c_\square\},$$

wobei $\dot{\leq}$ ein 2-stelliges Relationssymbol, g_{+1}, g_{-1}, f_q, f_p einstellige Funktionssymbole, f_β ein zweistelliges Funktionssymbol und $\dot{0}, c_0, c_+, c_-, c_\square$ Konstantensymbole sind.

Eine Turingmaschine

Um die Reduktion zu beschreiben, ist es am einfachsten, eine TM M festzuhalten.

Vereinbarung

Im Folgenden sei $M = (Q, \Gamma, \Sigma, \delta)$ eine Turingmaschine mit Zusandsmenge $Q = \{q_0, \dots, q_k\}$, Bandalphabet $\Gamma = \{a_0, \dots, a_\ell\}$ und Eingabealphabet $\Sigma = \{a_0, \dots, a_m\}$.

Weiterhin sei der Lauf von M bei Eingabe ε

$$\begin{cases} \kappa^{(0)}, \kappa^{(1)}, \dots, \kappa^{(h_M - 1)} & \text{falls } M \text{ bei Eingabe } \varepsilon \text{ anhält,} \\ \kappa^{(0)}, \kappa^{(1)}, \kappa^{(2)}, \dots & \text{sonst,} \end{cases}$$

und sei $\kappa^{(t)} = \left(q^{(t)}, p^{(t)}, \beta^{(t)}\right)$.

Wenn der Lauf von M bei Eingabe ε unendlich ist, dann setzen wir $h_M \coloneqq \infty$.

Ergänzungen zu Seite 6.23

Beachte: Es gilt $\Sigma \subset \Gamma$ und $\Box \in \Gamma \setminus \Sigma$, also $m < \ell$ und $\Box \in \{a_{m+1}, \ldots, a_\ell\}$.

Die Berechnungsstruktur von M ist die σ_{UE}-Struktur $\mathfrak{B}_M$ mit

- Universum $B_M := \mathbb{Z}$;
- $\dot{\leq}^{\mathfrak{B}_M} := \leq$ (natürliche Ordnung auf $\mathbb{Z}$);
- für $i \in \mathbb{Z}$ sind $g_{+1}^{\mathfrak{B}_M}(i) := i + 1$ und $g_{-1}^{\mathfrak{B}_M}(i) := i - 1$;
- $\dot{0}^{\mathfrak{B}_M} := 0$;
- für $0 \leq t < h_M$ und $i \in \mathbb{Z}$ sind

$$
\begin{aligned}
f_q^{\mathfrak{B}_M}(t) &:= j \qquad \text{für das } j \text{ mit } q^{(t)} = q_j,\\
f_p^{\mathfrak{B}_M}(t) &:= p^{(t)},\\
f_\beta^{\mathfrak{B}_M}(t,i) &:= j \qquad \text{für das } j \text{ mit } \beta^{(t)}(i) = a_j,
\end{aligned}
$$

  und für alle anderen t sind $f_q^{\mathfrak{B}_M}(t) = -1$, $f_p^{\mathfrak{B}_M}(t) = -1$, und $f_\beta^{\mathfrak{B}_M}(t,i) = -1$;
- $c_0^{\mathfrak{B}_M} := 0$, $c_+^{\mathfrak{B}_M} := j$ für das j mit $q_j = q_+$, $c_-^{\mathfrak{B}_M} := j$ für das j mit $q_j = q_-$, $c_\Box^{\mathfrak{B}_M} := j$ für das j mit $a_j = \Box$.

Plan

Schritt 1

Wir konstruieren einen σ_{UE}-Satz ψ_M, so dass für alle σ_{UE}-Strukturen $\mathfrak{B}$ gilt:

$$\mathfrak{B} \models \psi_M \iff \mathfrak{B} \cong \mathfrak{B}_M.$$

Beobachtung 6.18

M hält bei Eingabe ε genau dann, wenn es ein $t \in B_M$ gibt, so dass $f_q^{\mathfrak{B}_M}(t) \in \{c_+^{\mathfrak{B}_M}, c_-^{\mathfrak{B}_M}\}$.

Schritt 2

Wir setzen

$$\varphi_M := \psi_M \wedge \neg\exists x\big(f_q(x) = c_+ \vee f_q(x) = c_-\big).$$

Dann gilt φ_M genau dann erfüllbar, wenn es kein $t \in \mathbb{B}_M$ gibt, so dass $f_q^{\mathfrak{B}_M}(t) \in \{c_+^{\mathfrak{B}_M}, c_-^{\mathfrak{B}_M}\}$, also genau dann, wenn M bei Eingabe ε nicht hält.

Besserer Plan

Problem

Es gibt keinen σ_{UE}-Satz ψ_M, so dass für alle σ_{UE}-Strukturen $\mathfrak{B}$ gilt:

$$\mathfrak{B} \models \psi_M \quad \Longleftrightarrow \quad \mathfrak{B} \cong \mathfrak{B}_M.$$

Das lässt sich leicht mit Hilfe des Endlichkeitssatzes beweisen.

Schritt 1 (verbessert)

Wir konstruieren einen σ_{UE}-Satz ψ_M, so dass alle σ_{UE}-Strukturen $\mathfrak{B}$, die ψ_M erfüllen, genügend ähnlich zu $\mathfrak{B}_M$ sind, dass wie Schritt 2 trotzdem noch durchführen können.

Diskrete offene Ordnungen

Eine diskrete offene Ordnung auf einer Menge B ist eine totale Ordnung $\leq$ auf B, so dass für alle $b \in B$ gilt:

- b hat einen Nachfolger $c \in B$, d.h., $b < c$ und es gibt kein $b' \in B$, so dass $b < b' < c$;
- b hat einen Vorgänger $a \in B$, d.h., $a < b$ und es gibt kein $b' \in B$, so dass $a < b' < b$.

Man beachte, dass jedes Element in einer totalen Ordnung höchstens einen Vorgänger und einen Nachfolger hat, also in einer diskreten offenen Ordnung genau einen Vorgänger und genau einen Nachfolger.

Beispiel 6.19

Die natürliche Ordnung auf $\mathbb{Z}$ ist eine diskrete offene Ordnung.

Ergänzungen zu Seite 6.27

In dichten Ordnungen wie den natürlichen Ordnungen auf $\mathbb{Q}$ oder $\mathbb{R}$ haben Elemente keine direkten Nachfolger, weil es zwischen zwei Zahlen immer noch eine weitere Zahl gibt.

Wir setzen

$$
\begin{aligned}
\psi_{\mathrm{ord}} \coloneqq\ & \forall x \forall y \big((x \dot{\leq} y \wedge y \dot{\leq} x) \leftrightarrow x \doteq y \big) \\
& \wedge \forall x \forall y \forall z \big(x \dot{\leq} y \wedge y \dot{\leq} z \to x \dot{\leq} z \big) \\
& \wedge \forall x \forall y (x \dot{\leq} y \vee y \dot{\leq} x) \\
& \wedge \forall x \big(x \dot{\leq} g_{+1}(x) \wedge x \dot{\neq} g_{+1}(x) \wedge \forall y (y \dot{\leq} x \vee g_{+1}(x) \dot{\leq} y) \big) \\
& \wedge \forall x \forall y \big(g_{+1}(x) = y \leftrightarrow g_{-1}(y) = x \big).
\end{aligned}
$$

Beobachtung 6.20

Sei $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_{\mathrm{ord}}$. Dann gilt:

(1) $\dot{\leq}^{\mathfrak{B}}$ *ist eine totale Ordnung auf B;*

(2) $g_{+1}^{\mathfrak{B}}$ *und* $g_{-1}^{\mathfrak{B}}$ *sind die zugehörige Nachfolger- und Vorgängerfunktionen.*

Es folgt, dass $\dot{\leq}^{\mathfrak{B}}$ eine diskrete offene Ordnung ist.

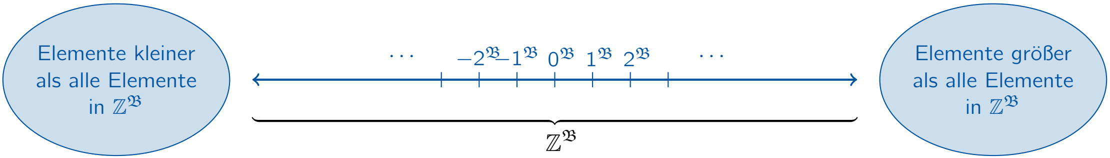
Modelle von ψ_{ord}

Wir definieren rekursiv für alle $i \in \mathbb{Z}$ einen Term η_i:

- $\eta_0 := 0$;
- $\eta_{i+1} := g_{+1}(\eta_i)$ für $i \geq 0$;
- $\eta_{i-1} := g_{-1}(\eta_i)$ für $i \leq 0$.

Sei nun $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_{\text{ord}}$. Für alle $i \in \mathbb{Z}$ sei $i^{\mathfrak{B}} := [\![\eta_i]\!]^{\mathfrak{B}}$, und sei $\mathbb{Z}^{\mathfrak{B}} := \{i^{\mathfrak{B}} \mid i \in \mathbb{Z}\}$.

Bezüglich der Ordnung sieht dann $\mathfrak{B}$ folgendermaßen aus:



Seien

$$\begin{aligned} j_+ &\coloneqq \text{das } j \in \{0,\dots,k\}, \text{ so dass } q_+ = q_{j_+}, \\ j_- &\coloneqq \text{das } j \in \{0,\dots,k\}, \text{ so dass } q_- = q_{j_-}, \\ j_\Box &\coloneqq \text{das } j \in \{0,\dots,\ell\}, \text{ so dass } \Box = a_{j_\Box}. \end{aligned}$$

Wir setzen

$$\psi_{\mathrm{konst}} \coloneqq c_0 \doteq \dot{0} \wedge c_+ \doteq \eta_{j_+} \wedge c_- \doteq \eta_{j_-} \wedge c_\Box \doteq \eta_{j_\Box}.$$

Beobachtung 6.21

Sei $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_{\mathrm{ord}} \wedge \psi_{\mathrm{konst}}$. Dann gilt

$$\begin{aligned} c_0^{\mathfrak{B}} &= 0^{\mathfrak{B}} && \textit{(Index von } q_0\textit{)}, \\ c_+^{\mathfrak{B}} &= j_+^{\mathfrak{B}} && \textit{(Index von } q_+\textit{)}, \\ c_-^{\mathfrak{B}} &= j_-^{\mathfrak{B}} && \textit{(Index von } q_-\textit{)}, \\ c_\Box^{\mathfrak{B}} &= j_\Box^{\mathfrak{B}} && \textit{(Index von } \Box\textit{)}. \end{aligned}$$

Sei $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_{\text{ord}} \wedge \psi_{\text{konst}}$, und sei $b \in B$. Wir setzen

$$q_b^{\mathfrak{B}} := \begin{cases} q_j & \text{falls } f_q^{\mathfrak{B}}(b) = j^{\mathfrak{B}} \text{ für ein } j \in \{0, 1 \ldots, k\}, \\ \text{undefiniert} & \text{sonst,} \end{cases}$$

$$p_b^{\mathfrak{B}} := \begin{cases} i & \text{falls } f_p^{\mathfrak{B}}(b) = i^{\mathfrak{B}} \in \mathbb{Z}^{\mathfrak{B}}, \\ \text{undefiniert} & \text{sonst,} \end{cases}$$

Falls für alle $b' \in B$ ein $j \in \{0, \ldots, m\}$ existiert, so dass $f_\beta(b, b') = j^{\mathfrak{B}}$, so definieren wir $\beta_b^{\mathfrak{B}} : \mathbb{Z} \to \Gamma$ durch

$$\beta_b^{\mathfrak{B}}(i) := a_j \text{ für das } j \text{ mit } f_\beta(b, i^{\mathfrak{B}}) = j^{\mathfrak{B}}.$$

Sonst ist $\beta_b^{\mathfrak{B}}$ undefiniert.

Wenn $q_b^{\mathfrak{B}}$, $q_b^{\mathfrak{B}}$ und $\beta_b^{\mathfrak{B}}$ definiert sind, dann sagen wir: $\mathfrak{B}$ beschreibt zum Zeitpunkt b die Konfiguration $\kappa_b^{\mathfrak{B}} := (q_b^{\mathfrak{B}}, p_b^{\mathfrak{B}}, \beta_b^{\mathfrak{B}})$.

Sonst beschreibt $\mathfrak{B}$ beschreibt zum Zeitpunkt b keine Konfiguration, und $\kappa_b^{\mathfrak{B}}$ ist undefiniert.

Wir setzen

$$\psi_{\text{init}} := f_q(\dot{0}) \doteq c_0 \land f_p(\dot{0}) \doteq \eta_1 \land \forall y\, f_\beta(\dot{0}, y) \doteq c_\Box$$
$$\land \forall x \big(x \mathrel{\dot{\leq}} \eta_{-1} \to (f_q(x) \doteq \eta_{-1} \land f_p(x) \doteq \eta_{-1} \land \forall y\, f_\beta(x, y) \doteq \eta_{-1}) \big)$$

Beobachtung 6.22

Sei $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_{\text{ord}} \land \psi_{\text{konst}} \land \psi_{\text{init}}$. Dann beschreibt $\mathfrak{B}$ zum Zeitpunkt $0^{\mathfrak{B}}$ die Anfangskonfigration von M bei Eingabe ε.

Außerdem gilt für alle $b, b' \in B$ mit $b \mathrel{\dot{\leq}}^{\mathfrak{B}} -1^{\mathfrak{B}}$

$$f_q^{\mathfrak{B}}(b) = f_p^{\mathfrak{B}}(b) = f_\beta^{\mathfrak{B}}(b, b') = -1^{\mathfrak{B}}.$$

Für $q = q_i, q' = q_{i'} \in Q$, $a = a_j, a' = a_{j'} \in \Gamma$ und $d \in \{-1, 0, +1\}$ setzen wir

$$
\begin{aligned}
\chi_{q,a,q',a',d}(x, x') :=\ & \Big(f_q(x) \doteq \eta_i \wedge f_\beta(x, f_p(x)) \doteq \eta_j\Big) \\
& \to \Bigg(f_q(x') \doteq \eta_{i'} \wedge f_\beta(x', f_p(x)) \doteq \eta_{j'} \wedge \forall y \big(y \not\doteq f_p(x) \to f_\beta(x', y) = f_\beta(x, y)\big)\big) \\
& \wedge \left\{ \begin{array}{ll} f_p(x') \doteq g_{-1}(f_p(x)) & \text{falls } d = -1, \\ f_p(x') \doteq f_p(x) & \text{falls } d = 0, \\ f_p(x') \doteq g_{+1}(f_p(x)) & \text{falls } d = +1 \end{array} \right\} \Bigg)
\end{aligned}
$$

Beobachtung 6.23

Sei $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_{\text{ord}} \wedge \psi_{\text{konst}}$, und seien $b, b' \in B$, so dass $\mathfrak{B}$ zum Zeitpunkt b die Konfiguration $\kappa = (q, p, \beta)$ beschreibt. Seien $a := \beta(p)$ und $\delta(q, a) =: (q', a', d)$. Wenn dann

$$\mathfrak{B} \models \chi_{q,a,q',a',d}(b, b')$$

so beschreibt $\mathfrak{B}$ zum Zeitpunkt b' die Nachfolgerkonfiguration κ' von κ.

Nachfolgekonfigurationen (Forts.)

Wir setzen

$$\chi_{\text{nach}}(x, x') := \bigvee_{\substack{q,q' \in Q, a,a' \in \Gamma, d \in \{-1,0,+1\} \\ \text{mit } \delta(q,a)=(q',a',d)}} \chi_{q,a,q',a',d}(x, x'),$$

$$\psi_{\text{nach}} := \forall x \Big(\dot{0} \mathrel{\dot{\leq}} x \to \chi_{\text{nach}} \frac{g_{+1}(x)}{x'} \Big).$$

Beobachtung 6.24

Sei $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_{\text{ord}} \wedge \psi_{\text{konst}}$.

(1) *Seien $b, b' \in B$, so dass $\mathfrak{B} \models \chi_{\text{nach}}(b, b')$. Wenn $\mathfrak{B}$ zum Zeitpunkt b die Konfiguration κ beschreibt, dann beschreibt $\mathfrak{B}$ zum Zeitpunkt b' die Nachfolgerkonfiguration κ' von κ.*

(2) *Gelte $\mathfrak{B} \models \psi_{\text{nach}}$. Dann gilt für alle $t \in \mathbb{Z}$, $t \geq 0$: wenn $\mathfrak{B}$ zum Zeitpunkt $t^{\mathfrak{B}}$ die Konfiguration κ beschreibt, dann beschreibt $\mathfrak{B}$ zum Zeitpunkt $(t+1)^{\mathfrak{B}}$ die Nachfolgerkonfiguration κ' von κ.*

Wir setzen

$$\psi_M := \psi_{\text{ord}} \wedge \psi_{\text{konst}} \wedge \psi_{\text{init}} \wedge \psi_{\text{nach}}.$$

Lemma 6.25

(1) *Sei $\mathfrak{B}$ eine σ_{UE}-Struktur mit $\mathfrak{B} \models \psi_M$. Dann gilt für alle $t \in \mathbb{Z}, 0 \leq t < h_M$:*
$\mathfrak{B}$ beschreibt zum Zeitpunkt $t^{\mathfrak{B}}$ die Konfiguration $\kappa^{(t)}$.

(2) $\mathfrak{B}_M \models \psi_M$.

Ergänzungen zu Seite 6.35

Beweis.

Behauptung (1) folgt aus Beobachtungen 6.22 und 6.24(2).

Behauptung (2) lässt sich leicht entlang der Definitionen der Formeln verifizieren. $\square$

Schritt 2

Wir setzen

$$\varphi_M := \psi_M \wedge \neg\exists x\big(f_q(x) = c_+ \vee f_q(x) = c_-\big).$$

Lemma 6.26

φ_M ist genau dann erfüllbar, wenn M bei Eingabe ε nicht hält.

Beweis.

„$\Longrightarrow$“: Sei $\mathfrak{B} \models \varphi_M$. Wegen Lemma 6.25 gilt dann für $0 \leq t < h_M$, dass $\mathfrak{B}$ zum Zeitpunkt $t^{\mathfrak{B}}$ die Konfiguration $\kappa^{(t)}$ beschreibt.

Angenommen, M hält. Dann ist $h_M \in \mathbb{N}_{>0}$, und $\kappa^{(h_M-1)}$ ist eine Haltekonfiguration. Also gilt für $b := (h_M - 1)^{\mathfrak{B}})$

$$f_q^{\mathfrak{B}}(b) \in \{c_-^{\mathfrak{B}}, c_+^{\mathfrak{B}}\}.$$

Das *widerspricht* $\mathfrak{B} \models \varphi_M$.

„$\Longleftarrow$“: Wenn M bei Eingabe ε nicht hält gilt $\mathfrak{B}_M \models \varphi_M$. $\square$

Beweis von Satz 6.17.

Die Abbildung $M \mapsto \varphi_M$ ist klar berechenbar und damit wegen Lemma 6.26 eine Reduktion von $\overline{\mathrm{H}_\varepsilon}$ auf $\textsc{Erf}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$. Weil $\overline{\mathrm{H}_\varepsilon}$ unentscheidbar ist, ist auch $\textsc{Erf}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$ unentscheidbar. $\square$

6.3 Semi-Entscheidbarkeit und Erfüllbarkeit im Endlichen

Das Allgemeingültigkeitsproblem

Korollar 6.27

$\textsc{Allg}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$ *ist unentscheidbar.*

Satz 6.28

Für alle semi-entscheidbaren σ ist $\textsc{Allg}\big(\mathsf{L}(\sigma)\big)$ semi-entscheidbar.

Beweis des Satzes.

Nach dem Vollständigkeitssatz ist eine Formel $\varphi \in \mathsf{L}(\sigma)$ allgemeingültig, wenn die Sequenz $\vdash \varphi$ ableitbar ist. Eine Semi-Entscheidungsalgorithmus für $\textsc{Allg}\big(\mathsf{L}(\sigma)\big)$ generiert systematisch alle ableitbaren Sequenzen und akzeptiert, wenn er die Sequenz $\vdash \varphi$ für die Eingabeformel φ findet. Ansonsten hält der Algorithmus nie an. $\square$

Korollar 6.29

$\text{ÄQ}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$ *ist unentscheidbar.*

Korollar 6.30

Für alle semi-entscheidbaren σ ist $\text{ÄQ}\big(\mathsf{L}(\sigma)\big)$ semi-entscheidbar.

Zur Semi-Entscheidbarkeit des Erfüllbarkeitsproblems

Korollar 6.31

Für alle semi-entscheidbaren σ ist $\textsc{UErf}\big(\mathsf{L}(\sigma)\big)$ *semi-entscheidbar.*

Korollar 6.32

$\textsc{Erf}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$ *ist nicht semi-entscheidbar.*

Endliche Erfüllbarkeit

Definition 6.33

Eine Formel $\varphi \in \mathsf{L}(\sigma)$ ist endlich erfüllbar, wenn es eine endliche σ-Interpretation gibt, die φ erfüllt.

$\textsc{EndErf}\big(\mathrm{L}(\sigma)\big)$

Eingabe: $\varphi \in \mathsf{L}(\sigma)$.

Frage: Ist φ endlich erfüllbar?

Satz 6.34

Für alle semi-entscheidbaren σ ist $\textsc{EndErf}\big(\mathsf{L}(\sigma)\big)$ semi-entscheidbar.

Unentscheidbarkeit des Endlichen-Erfüllbarkeits-Problems

Satz 6.35 (Satz von Trachtenbrot)

Es gibt eine endliche Symbolmenge σ, so dass $\textsc{EndErf}\big(\mathsf{L}(\sigma)\big)$ *unentscheidbar ist.*

Ergänzungen zu Seite 6.42

Der Beweis des Satzes von Trachtenbrot geht ähnlich wie der Beweis der Unentscheidbarkeit des Erfüllbarkeitsproblems. Hier reduzieren wir H_ε auf $\textsc{EndErf}\big(\mathsf{L}(\sigma_{\mathsf{UE}})\big)$. Wir kodieren wieder die Berechnungen einer Turingamschine M bei Eingabe ε durch eine Struktur $\mathfrak{B}_M$, hier achten wir allerdings darauf, dass die Struktur $\mathfrak{B}_M$ endlich ist, wenn M bei Eingabe ε anhält. Dazu schneiden wir bei den Konfigurationen auf beiden Seiten die unendliche vielen $\square$-Symbole ab. Dann können wir eine endliche Berechnung auch durch eine endliche Struktur beschreiben.