Typ-1, Kuroda Normalform

Wie bei den kontextfreien Grammatiken wollen wir auch bei den nichtverkürzenden, d.h. Typ-1, Grammatiken eine spezielle Form finden und nachweisen, dass es zu jeder Typ-1 Grammatik eine äquivalente Grammatik in dieser Form gibt:

Definition: Eine Grammatik ist in *Kuroda Normalform*, wenn alle Regeln von einem der folgenden 4 Typen sind:

$$A \to a,\quad A \to B,\quad A \to BC,\quad AB \to CD$$

Auf den folgenden zwei Folien wollen wir nachweisen, dass es zu jeder Typ-1 Grammatik G mit $\varepsilon \notin L(G)$ eine Typ-1 Grammatik G' in Kuroda Normalform gibt, so dass $L(G) = L(G')$.

Kuroda-Normalform, Beweis des Satzes

Wie gewohnt verwenden wir Pseudo-Terminale, um zu erreichen, dass Terminale nur noch in Regeln der Form $A \to a$ vorkommen. Welche Regeln stören uns jetzt noch? Nur solche:

$$A_1 A_2 \ldots A_m \to B_1 B_2 \ldots B_n \qquad \text{mit } 1 \leq m \leq n \text{ und } n > 2$$

Im Fall $m = 1$ und $n > 2$ können wir dieselbe Konstruktion verwenden wie bei den Typ-2 Normalformen:

$$A_1 \to B_1 B_2 \ldots B_n$$

wird ersetzt durch die folgenden $n - 1$ Regeln:

$$\begin{aligned} A_1 &\to B_1 C_2 \\ C_2 &\to B_2 C_3 \\ &\vdots \\ C_{n-1} &\to B_{n-1} B_n \end{aligned}$$

mit neuen Variablen $C_2, \ldots, C_{n-1}$

Beweis (Abschluss)

Nun fehlt nur noch der Fall $2 \leq m \leq n$ und $n > 2$.

$$A_1 A_2 \ldots A_m \to B_1 B_2 \ldots B_n$$

ersetzen wir in diesem Fall durch die folgenden zwei Regeln:

$$A_1 A_2 \to B_1 D_2$$
$$D_2 A_3 \ldots A_m \to B_2 B_3 \ldots B_n$$

Auch hier ist D_2 eine neue Variable, und es sollte eine leichte Übung sein nachzuweisen, dass diese beiden Regeln die ursprüngliche (rote) Regel gleichwertig ersetzen.

Da die Regel $D_2 A_3 \ldots A_m \to B_2 B_3 \ldots B_n$ kürzer ist als die ursprüngliche Regel, ist klar, dass man induktiv davon ausgehen kann, dass diese neue Regel bereits gleichwertig durch Regeln ersetzbar ist, die der Kuroda Form entsprechen.

Turingmaschinen (TM), bildlich...

Eine Turingmaschine besteht im Wesentlichen aus einem unendlichen Arbeitsband mit Schreib-/Lesekopf und einer unendlichen Zustandskontrolle.

Zeichnen Sie sich bitte selbst ein Bild von diesem Maschinentyp!

Welche Aktion führt eine TM in einem Schritt durch?

In Kenntnis des eigenen Zustands und des vom Schreib-/Lesekopf gescannten Symbols:

1. Schreibe ein (neues) Symbol
2. Ändere den (inneren) Zustand in der Kontrolle
3. Schiebe den Schreib-/Lesekopf um maximal ein Feld nach rechts oder links

Formale Definition der TM

Formal ist eine Turingmaschine ein 7-Tupel:

$$M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$$

Dabei ist:

Z: Die Menge der Zustände (endlich, nicht leer)

Σ: Ein endliches Eingabealphabet ($\Sigma \cap Z = \emptyset$)

Γ: Ein endliches Bandalphabet ($\Gamma \cap Z = \emptyset$, $\Sigma \subseteq \Gamma$)

δ : $Z \times \Gamma \to Z \times \Gamma \times \{L, R, N\}$ die Überführungsfunktion

oder δ : $Z \times \Gamma \to \mathcal{P}(Z \times \Gamma \times \{L, R, N\})$ (NICHTDETERMINISTISCH!)

$z_0 \in Z$: der Startzustand

$\Box \in \Gamma \setminus \Sigma$: das Leersymbol

$E \subseteq Z$: Menge der akzeptierenden Endzustände

Interpretation der Übergänge einer TM

Was bedeutet nun $\delta(z, a) = (z', a', X)$ für $z, z' \in Z$, $a, a' \in \Gamma$ und $X \in \{L, R, N\}$?
(bzw. $(z', a', X) \in \delta(z, a)$ im nichtdeterministischen Fall...)

Wir interpretieren diesen Übergang wie folgt:

Wenn die Turingmaschine sich im Zustand z befindet und der Schreib-/Lesekopf auf dem Arbeitsband ein a scannt, dann ersetzt der Kopf dieses a durch a', die innere Zustandskontrolle nimmt den Zustand z' an, und die Position des Schreib-/Lesekopfes ändert sich wie folgt:

Bei $X = N$ bleibt sie unverändert.
Bei $X = L$ wird der Kopf ein Feld nach links bewegt.
Bei $X = R$ wird der Kopf ein Feld nach rechts bewegt.