Blatt 4 w

Tuesday, May 12, 2020 5:45 PM

Def.: 1.3

$\boxed{L \text{ entscheidbar} \Leftrightarrow M_L \text{ hält immer}}$

(„⇐" hängt davon ab, ob, wie man verwerfen definiert. Ist aber immer möglich!)

A4.1:

a) Bew.: Sei $M_{\leq 42}$ die TM, die $H_{\leq 42}$ entscheidet.

z.: $M_{\leq 42}$ akz. $w \in H_{\leq 42}$ und $M_{\leq 42}$ verw. $w \notin H_{\leq 42}$

(i) Sei $w \in H_{\leq 42}$.

Dann ist w in Form $w = \langle M \rangle$. Die TM M hält nun auf jeder Eingabe nach spätestens 42 Schritten.

[Dies kann umgesetzt werden, indem man sich die gemachten Schritte merkt und bei 42 akzeptiert.

↳ Das merken könnte man mit Zuständen machen: $Q_{\leq 42} = Q_M \times \{ k_1, .., k_{42} \}$]

unabhängig davon ob M akz. oder verw. akzeptiert $M_{\leq 42}$ w.
Hier können wir in keiner Dauerschleife stecken bleiben, da M spätestens nach 42 Schritten hält.

(ii) $w \notin H_{\leq 42}$

Hier ist M mit $w = \langle M \rangle$ eine TM die nicht nach 42 Schritten hält.
Sobald $M_{\leq 42}$ also zählt, dass der 43 Schritt erreicht wurde verwirft $M_{\leq 42}$ w.

Somit $\exists$ TM $M_{\leq 42}$ die $H_{\leq 42}$ entscheidet. □

b) Bew:

Sei M_{ein} die TM die $TAPE_{ein}$ entscheidet.

Problem: Sei M TM mit $\delta(q_0, x) = (q_0, x, N)$, dann wäre es ein Element von $TAPE_{ein}$, aber würde nicht halten. Somit hält M_{ein} auch nicht? $\Leftrightarrow$ $TAPE_{ein}$ nicht entscheidbar?

Verstehe nicht, wie ich das dann zeigen soll...

Modifizieren wir $TAPE_{ein}$ so, dass zusätzlich gilt, dass für $\langle M \rangle w \in TAPE_{ein}$ M auf w hält kann man das zeigen. Von dieser Erweiterung werde ich im folgenden ausgehen.

M_{ein} merkt sich, bei der Simulation von M auf w, die Länge von w und den Index der Position (Dies muss auf einer separaten Spur passieren, da w beliebig lang sein kann und man das mit Zuständen im voraus nicht abfangen kann). Hält M ohne $-|w|, .., |w|$ überschritten zu haben akzeptiert M_{ein} die Eingabe. Überschreitet der Index die Länge wird verworfen.

(i) $x \in TAPE_{ein}$

Dann $x = \langle M \rangle w$ und M_{ein} simuliert M auf w. M hält auf w und überschreitet $-|w|, \dots, |w|$ nicht. $\Rightarrow$ M_{ein} akzeptiert x

(ii) $x \notin TAPE_{ein}$

$\Rightarrow$ M überschreitet $-|w|, .., |w|$ $\Rightarrow$ M_{ein} verwirft x $\square$

c) Sei M_{q_0} TM die L_{q_0} entscheidet.

M_{q_0} durchsucht dann die Eingabe, bis der folgende Zust.-übergang gefunden wird:

$$\delta(q_0, \square) = (q', x, \xi) \text{ mit } q' \in Q \setminus \{q_0\},\ x \in \Gamma$$
$$\text{und } \xi \in \{L, N, R\}$$

(i) Sei $w \in L_{q_0}$.

$\Rightarrow M_{q_0}$ findet einen solchen Übergang und akzeptiert.

(ii) Sei $w \notin L_{q_0}$.

$\Rightarrow M_{q_0}$ findet keinen solchen Übergang und verwirft. $\square$

A4.3:

Wie will man sowas mit diagonalisierung machen?
Hätte jetzt eher versucht ein explizites Gegenbeispiel anzugeben.

A4.4:

$A_{\mathbb{P}} = \{\langle M \rangle \mid M$ entscheidet $\mathbb{P}\}$, also berechnet M die Funktion

$$f(x) = \begin{cases} 1 & \text{, wenn val}(x) \text{ Primzahl} \\ 0 & \text{, sonst} \end{cases}$$

Die Behauptung folgt direkt aus dem Satz von Rice. Da dieser auch durch eine Turing reduction gezeigt wurde kann man das hier analog machen.

(durch Widerspruch)

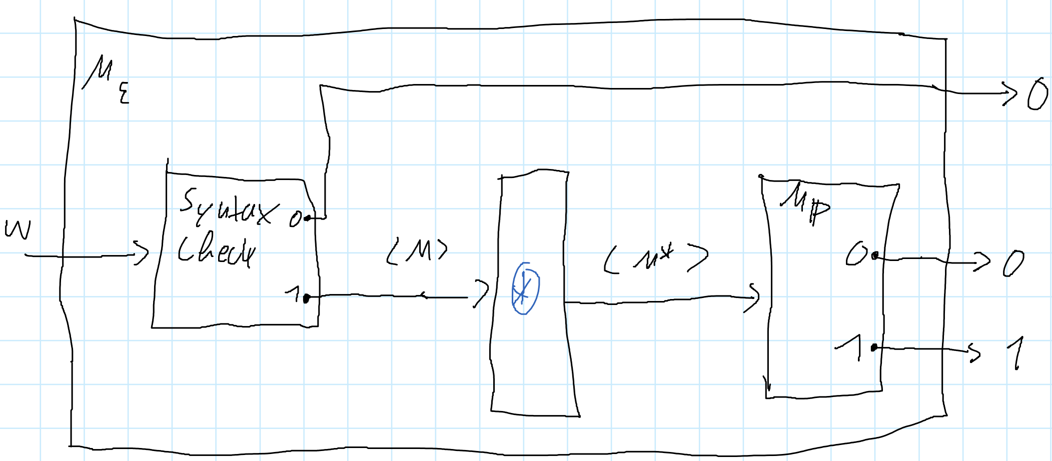
<u>Bew.:</u> Sei M_P die TM die A_P entscheidet.

Wir führen eine Turing-Reduktion des speziellen Halteproblems H_ε auf M_P durch.

Eine wichtige Beobachtung ist es, dass die Sprache P sowohl entscheidbar ist, also insbesondere immer hält.

Wir nennen die Funktion, die P entscheidet f.

Die Turing-Reduktion wird folgendermaßen konstruiert:



(*) Hier wird die Gödelnummer einer TM M^* mit folgendem Verhalten konstruiert:

↳ Erst simuliert M^* das Verhalten von M auf ε

↳ Dann berechnet M^* die Funktion f.

(An dieser Stelle sei noch ausdrücklich darauf hingewiesen, dass (*) M^* nicht simuliert, sondern nur die Gödelnummer einer TM ausgibt die sich wie M^* verhält!)

Nun zeigen wir die Reduktion:

(i) Sei $w \in H_\varepsilon$. Somit wird der Syntax-Check passiert und M^* berechnet f (weil M auf ε hält)

$\Rightarrow \langle M^* \rangle \in A_{\mathbb{P}} \Rightarrow M_{\mathbb{P}}$ akz. $\langle M^* \rangle$

$\Rightarrow M_\varepsilon$ akzeptiert w

(ii) Sei $w \notin H_\varepsilon$.

Dann stimmt entweder die Syntax nicht $\Rightarrow M_\varepsilon$ verwirft w oder M aus $w = \langle M \rangle$ hält nicht auf w.

$\Rightarrow M^*$ berechnet die Fkt. $u: \Sigma^* \to \{\perp\}$ mit $u(x) = \perp$ für alle $x \in \Sigma^*$

↝ Diese Funktion berechnet offensichtlich keinen Primzahltest, also

$\Rightarrow \langle M^* \rangle \notin A_{\mathbb{P}} \Rightarrow M_{\mathbb{P}}$ verwirft $\langle M^* \rangle$

$\Rightarrow M_\varepsilon$ verwirft w.

Damit haben wir bewiesen, dass wenn $A_{\mathbb{P}}$ entscheidbar ist, dann ist H_ε es auch.

↝ Theorem 2.12 sagt allerdings uns, dass H_ε nicht entscheidbar ist!

↝ Somit ist gezeigt, dass $A_{\mathbb{P}}$ auch nicht entscheidbar ist. $\square$

A4.2:

a) Das Programm berechnet $x_2 + x_3$ und schreibt das Ergebnis in x_1

b) —— " —— nix und terminiert nie.

b) ——||———— nix und terminiert nie.

"c)" $\frac{\ln(x)}{\ln(a)} = \log_a(x)$, $w = \log_x(y) \Leftrightarrow x^w = y$

x	y
2	4

$2^1 = 2$
$2^2 = 4$

irgendwie so...

—