Fleißige Bieber

TM-Wettbewerb: Maschinen mit n Zuständen konkurrieren, wer läuft am längesten (Maschinen die nicht halten sind disqualifiziert)
$\rightsquigarrow$ "fleißige Bieber"

Definition (Fleißige Bieber, [Radó '62])

Für jedes Codewort w einer Turing-Maschine M, sei
$s(w) =$ Anzahl Schritte die M (bei leerer Eingabe) macht (0 falls M nicht hält) und
$e(w) =$ Anzahl (nicht-$\square$) Symbole auf dem Band wenn M (bei leerer Eingabe) hält (sonst 0).
Für jedes $n \in \mathbb{N}$ sei $\mathcal{M}_1(n)$ die Menge aller Turing-Maschinen $M = (Z, \Sigma, \Gamma, \delta, \square, z_0, E)$ mit

$$|Z| = n \qquad \Sigma = \{1\} \qquad \Gamma = \Sigma \cup \{\square\}.$$

Wir definieren die Funktionen

$$S_1(n) := \max_{M \in \mathcal{M}_1(n)} s(\langle M \rangle) \qquad E_1(n) := \max_{M \in \mathcal{M}_1(n)} e(\langle M \rangle)$$

Eine Maschine $M \in \mathcal{M}_1(n)$ heißt unärer **fleißiger Bieber** (**busy beaver**) falls M auf leerer Eingabe $E_1(n)$ nicht-$\square$ Symbole auf das Band schreibt und dann hält.

Fleißige Bieber sind Unberechenbar I

Theorem

Die Funktion S ist nicht Turing-berechenbar.

Beweis

Annahme: S ist Turing-berechenbar

$\rightsquigarrow$ S wird von einer Turing-Maschine berechnet.

$\rightsquigarrow$ es gibt $n \in \mathbb{N}$ sodass S von einer Turing-Maschine $M_{BB} \in \mathcal{M}(n)$ berechnet wird.

Sei M' die Maschine, die wie folgt agiert:

1. verdopple die Zahl auf dem Band
2. simuliere M_{BB}
3. wandle das binäre Ausgabewort in unär

Sei n' die Anzahl Zustände von M'.

Sei M'' die Maschine, die wie folgt agiert:

1. erzeuge die Zahl n' auf dem Band
2. simuliere M'

sodass M'' genau $2n'$ Zustände hat.

$\rightsquigarrow$ M'' hat $2n'$ Zustände aber M'' macht bei leerer Eingabe **echt mehr** als $S(2n')$ Schritte $\lightning$

Frage: Überlegen Sie sich den analogen Beweis dafür, dass E nicht Turing-berechenbar ist.

Fleißige Bieber sind Unberechenbar II

Theorem

Die Funktion S wächst (asymptotisch) schneller als jede berechenbare Funktion!

Beweis (Skizze)

Annahme: es gibt berechenbare Funktion f und $n_0 \in \mathbb{N}$, sodass $f(n) > S(n)$ für alle $n > n_0$.
$\rightsquigarrow$ bauen Turing-Maschine M, die entscheidet ob $w \in H_0$ für gegebene Codierung w einer TM mit $\Sigma = \{0, 1\}$ und $\Gamma = \{0, 1, \square\}$ und $n := |Z|$.

1. $n \leq n_0$, dann gib fest verdrahtete Antwort aus (endlich viele)
2. berechne $f(n)$
3. simuliere M_w auf leerem Band höchstens $f(n)$ Schritte
4. gib aus, ob M_w nach höchstens $f(n)$ Schritten hielt

$\rightsquigarrow$ die von M berechnete Funktion ist total und es gilt:

$$\begin{aligned} w \in H_0 &\Leftrightarrow M_w \text{ hält auf leerem Band} \\ &\Leftrightarrow M_w \text{ hält auf leerem Band nach höchstens } f(n) \text{ Schritten} \Leftrightarrow M \text{ gibt } 1 \text{ aus} \end{aligned}$$

$\rightsquigarrow$ M berechnet χ_{H_0} ϟ