Der erweiterte Satz von Rice

Raphael Reitzig Sebastian Wild Markus E. Nebel

13. März 2012

1 Notation:

- Für (partielle) Funktionen f und g definieren wir $f \subseteq g$, »f ist eine Teilfunktion von g«, wenn

  (1) $D_f \subseteq D_g$ und

  (2) $\forall x \in D_f : f(x) = g(x)$

  Wir schreiben $f \subsetneq g$ für $f \subseteq g \,\wedge\, D_f \neq D_g$.

  (Das entspricht genau der Mengeninklusion auf den Graphen der Funktionen und wurde in ähnlicher Weise im Sematik-Kapitel des Buchs für Variablenbelegungen verwendet.)

- Definiere $f \cong g$, "f und g sind äquivalent/gleich", wenn $f \subseteq g$ und $g \subseteq f$.
- Für eine beliebige $n+1$-stellige Funktion g schreiben wir F_i^g für die Funktion definiert durch $F_i^g(\vec{x}) := g(i,\vec{x})$. [1]
- Schreibe $\mathfrak{I}\,P := \left\{ i \in \mathbb{N}_0 : F_i^{\mathscr{G}} \in P \right\}$ für die Indexmenge einer Funktionenklasse P.

2 Der erweiterte Satz von Rice

Der Beweis zum erweiterten Satz von Rice verwendet ein recht nützliches Lemma, das wir hier zuerst betrachten:

[1] Damit ist g eine universelle Funktion für die Klasse $K_n := \{F_i^g : i \in \mathbb{N}_0\}$ und induziert damit eine (nicht injektive) *Gödelisierung* G für K_n.

Lemma Compiler: Sei $\mathscr{G}$ eine Standard-Gödelisierung für K und $g \in K_{n+1}$ eine Funktion. Dann gibt es eine totale Funktion $c \in K_1$ mit

$$\forall i \in \mathbb{N}_0 \quad F^{\mathscr{G}}_{c(i)} \cong F^{g}_{i}.$$

Beweis: Wegen $g \in K$ und $\mathscr{G}$ Standard-Gödelisierung existiert $j \in \mathbb{N}_0$ mit $F^{\mathscr{G}}_j = g$. Damit gibt es nach smn-Eigenschaft für $\mathscr{G}$ eine *totale* Funktion $s_{1,n} \in K_2$ mit

$$\forall i \in \mathbb{N}_0 \quad \forall x \in \mathbb{N}_0, \vec{y} \in \mathbb{N}^n \quad F^{\mathscr{G}}_{s_{1,n}(i,x)}(\vec{y}) = F^{\mathscr{G}}_i(x,\vec{y}) \ .$$

Das gilt natürlich insbesondere für $i = j$, d. h. $F^{\mathscr{G}}_{s_{1,n}(j,x)}(\vec{y}) \cong g(x,\vec{y})$. Setze jetzt $c(i) := s_{1,n}(j,i)$.[2] Dann gilt

$$\forall x \in \mathbb{N}_0, \vec{y} \in \mathbb{N}^n \quad F^{\mathscr{G}}_{c(x)}(\vec{y}) = F^{\mathscr{G}}_{s_{1,n}(j,x)}(\vec{y}) = g(x,\vec{y}) = F^{g}_{x}(\vec{y}).$$

$\square$

Satz (Erweiterter Satz von Rice):
Sei $\mathscr{G}$ eine Standard-Gödelisierung für $\mathscr{R}$ und $A \subseteq \mathbb{N}_0$ eine Menge. Gibt es $f, g \in \mathscr{R}_n$ mit

(1) $f \subsetneq g$,

(2) $\mathfrak{I}\{f\} \subseteq A$, d. h. alle Indizes von f sind in A, und

(3) $\mathfrak{I}\{g\} \cap A = \emptyset$, d. h. kein Index von g ist in A,

so ist A nicht rekursiv aufzählbar.

Beweis: Reduktion auf $\overline{D} = \mathbb{N}_0 \backslash D$ nicht rekursiv aufzählbar, mit $D = \left\{ i : F^{\mathscr{G}}_i(i) \text{ def.} \right\}$.

Annahme: A sei rekursiv aufzählbar.

Wir definieren die Funktion h durch

$$h(x,\vec{y}) := \begin{cases} g(\vec{y}) & \text{für } x \in D \\ f(\vec{y}) & \text{sonst} \end{cases} .$$

Behauptung 1: $h \in \mathscr{R}_{n+1}$

[2] Wir behaupten hier (stillschweigend), dass $c \in K_1$. Das gilt aber nicht für beliebige K, sondern nur solche, die alle konstanten Funktionen enthalten und gegen Einsetzen abgeschlossen sind und gehört selbstredend zu den Voraussetzungen des Lemmas; der Übersichtlichkeit halber dort weggelassen. Anwenden werden wir das Lemma nur für $K = \mathscr{R}$, was diese Anforderung natürlich erfüllt.

Beweis via Timesharing: Wir berechnen abwechselnd $F_x^{\mathscr{G}}(x)$ und $f(\vec{y})$. Terminiert ersteres, berechnen wir anschließend $g(\vec{y})$, ist $f(\vec{y})$ als Erstes berechnet, so geben wir $f(\vec{y})$ aus. Das beschrieben Vorgehen lässt sich beispielsweise auf Turing Maschinen realisieren, ist also berechenbar.

Für den Nachweis, dass wir auch tatsächlich $h(x,\vec{y})$ berechnen unterscheiden wir:

- $x \in D$ und $f(\vec{y})$ undefiniert
  Die Berechnung von $f(\vec{y})$ terminiert nicht, $F_x^{\mathscr{G}}(x)$ schon. Also tun wir das, was $g(\vec{y})$ getan hätte. Egal ob das terminiert oder nicht, es ist das richtige.

- $x \notin D$ und $f(\vec{y}) = z$
  Hier wird sicher $f(\vec{y})$ als einziges fertig, die Ausgabe z ist genau das Gewünschte.

- $x \notin D$ und $f(\vec{y})$ undefiniert
  Hier ist $h(x,\vec{y})$ nicht definiert und unsere Berechnung simuliert abwechselnd zwei nicht terminierende Auswertungen, passt.

- $x \in D$ und $f(\vec{y}) = z$
  Das ist der spannendste Fall, denn wir haben zwei terminierende Berechnungen, von denen wir nicht wissen, welcher zuerst fertig wird. Ist $F_x^{\mathscr{G}}(x)$ zuerst durch, machen wir mit der weiteren Auswertung von $g(\vec{y})$ das Richtige. Ist aber $f(\vec{y})$ zuerst fertig, so ist unsere Ausgabe $z = f(\vec{y})$, aber $h(x,\vec{y}) = g(\vec{y})$. Problem? Nein, denn mit $f \subsetneq g$ und $z \in D_f$ folgt $z \in D_g$ und $z = f(\vec{y}) = g(\vec{y})$.

Die (angedeutete) Turing Maschine berechnet also tatsächlich h. $\square_{\text{[Behauptung 1]}}$

Mit $h \in \mathscr{R}$ und $\mathscr{G}$ Standard-Gödelisierung existiert nun nach Lemma Compiler $c \in \mathscr{T}_1$ mit $\forall i \in \mathbb{N}_0\ F_{c(i)}^{\mathscr{G}} \cong F_i^h$.

Beachte: Die Klasse von Funktionen, für die h eine universelle Funktion ist, ist ziemlich langweilig:

$$\left\{F_i^h : i \in \mathbb{N}_0\right\} = \left\{f, g\right\}. \tag{$*$}$$

Damit haben wir alles zusammen: Sei $x \in \mathbb{N}_0$ beliebig so gilt

$$x \in \overline{D} \quad \underset{\text{Def } h}{\iff} \quad f \cong F_x^h \underset{\text{Compiler}}{\cong} F_{c(x)}^{\mathscr{G}} \quad \underset{(*)}{\iff} \quad c(x) \in A.$$

Nach Annahme ist A rekursiv aufzählbar, d. h. es gibt nach Satz 4.23 eine Funktion $\pi_A \in \mathscr{R}_1$ mit $D_{\pi_A} = A$. Folglich ist auch die Funktion $\pi_{\overline{D}} := \pi_A \circ c$ berechenbar und es gilt $D_{\pi_{\overline{D}}} = \overline{D}$, also wäre – wieder nach 4.23 – $\overline{D}$ rekursiv aufzählbar.
$\lightning$ da wir wissen, dass $\overline{D}$ *nicht* rekursiv aufzählbar ist (Satz 4.24 (b)). $\square$

3 Anwendungsbeispiel

Behauptung: Die Menge $A := \{x \in \mathbb{N}_0 : \exists y \in \mathbb{N}_0\ F_x^{\mathscr{G}_1}(y) \text{ undef.}\}$, d. h. die Menge aller Indizes von berechenbaren, nicht-totalen Funktionen, ist nicht rekursiv aufzählbar.

Beweis: via erweitertem Satz von Rice. Prüfen wir also die Voraussetzungen:

Dazu müssen wir ›passende‹ Funktionen f und g finden. Wir wählen $f = \Phi_1$, die einstellige nirgends definierte Funktion, und $g = C_1^0$, die einstellige konstante 0-Funktion. Wir prüfen:

(1) $f \subsetneq g$
$\emptyset = D_{\Phi_1} = D_f \subsetneq D_g = D_{C_1^0} = \mathbb{N}_0$ und trivialerweise $\forall x \in \emptyset : f(x) = g(x)$.

(2) $\mathfrak{I}\{f\} \subseteq A$
Da $f = \Phi$ nicht total, gilt für jeden Index $x \in \mathfrak{I}\{f\}$: $\exists y : F_x^{\mathscr{G}_1}(y)$ undef. Also ist $x \in A$.

(3) $\mathfrak{I}\{g\} \cap A = \emptyset$
g total, also gilt für beliebiges $x \in \mathfrak{I}\{g\}$: $\nexists y : F_x^{\mathscr{G}_1}(y)$ undef. und damit $x \notin A$.

Damit ist der erweiterte Satz von Rice anwendbar und es folgt, dass A nicht rekursiv aufzählbar ist. $\square$