Kapitel 4

Arithmetische Prädikate

Wir wenden uns in den letzten beiden Kapitel den Beziehungen zwischen Rekursionstheorie und Arithmetik zu. Insbesondere interessieren wir uns für Anwendungen rekursionstheoretischer Ergebnisse auf die Arithmetik.

Arithmetische Prädikate entstehen durch arithmetische Komprehension: es sind die Prädikate, die durch Formeln der Zahlentheorie der 1. Stufe definiert werden. Gewöhnlich wählt man in der Zahlentheorie als nicht-logische Grundzeichen die Konstante 0 und die Funktionszeichen *suc* (für den Nachfolger), $+$ und $\cdot$, eventuell noch die Kleiner-Relation $<$ oder die arithmetische Differenz $\dot{-}$. Eine Sprache der Zahlentheorie auf einer solchen sparsamen Grundlage werden wir in 4.3 einführen. Zunächst verfahren wir großzügiger und verwenden wie bisher Namen für alle rekursiven Prädikate und Funktionen in unserer Sprache. Dieses großzügige Verfahren ist dadurch gerechtfertigt, dass alle rekursiven Funktionen und Prädikate auch mit den Ausdrucksmitteln der gewöhnlichen Zahlentheorie definiert werden können. Mehr als das, alle rekursiven Funktionen und Prädikate - und auch nur diese - sind, wie in 4.3 gezeigt wird, sogar repräsentierbar im Robinson-Formalismus, einem bescheidenen Fragment der gewöhnlichen Zahlentheorie.

4.1 Die arithmetische Hierarchie

Die Menge K, definiert durch

$$x \in K \leftrightarrow \exists y \;\; T(x, x, y)$$

ist nicht rekursiv. Die Anwendung von Quantoren auf rekursive Prädikate führt demnach aus der Klasse der rekursiven Prädikate heraus. Wir untersuchen hier die Prädikate, die mittels Quantifikation über Zahlen aus rekursiven

Prädikaten hervorgehen, und fragen, ob durch Vorschalten von immer mehr Quantoren immer neue, kompliziertere Prädikate definierbar werden.

4.1.1 Rekursive Definition der Klassen $\boldsymbol{\Sigma_n}$ und $\boldsymbol{\Pi_n}$.

1. Es sei $\Sigma_0 = \Pi_0$ die Klasse der rekursiven Prädikate.

2. Ein m-stelliges Prädikat P **gehört zu** Σ_{n+1}, P **ist** Σ_{n+1} oder **ein** Σ_{n+1}**-Prädikat**, wenn es ein $(m+1)$-stelliges Prädikat Q in Π_n gibt mit

$$P(\boldsymbol{y}) \leftrightarrow \exists x Q(x, \boldsymbol{y});$$

P **gehört zu** Π_{n+1}, P **ist** Π_{n+1} oder **ein** Π_{n+1}**-Prädikat**, wenn es ein $(m+1)$-stelliges Prädikat Q in Σ_n gibt mit

$$P(\boldsymbol{y}) \leftrightarrow \forall x Q(x, \boldsymbol{y}).$$

Bemerkung. Ein Prädikat P ist genau dann Σ_n (bzw. Π_n), wenn es ein rekursives Prädikat R gibt mit

$$(1) \qquad P(\boldsymbol{y}) \leftrightarrow Q_1 x_1 ... Q_n x_n R(x_1, ..., x_n, \boldsymbol{y}),$$

wobei $Q_1, ..., Q_n$ eine alternierende Folge von n Quantoren ist und der **erste** Quantor Q_1 der Existenzquantor $\exists$ (bzw. der Allquantor $\forall$) ist. Die Σ_1-Prädikate sind nach Satz 2.2.4 genau die r.a. Prädikate. Die doppelte Folge dieser Prädikatsklassen

$$\Sigma_0 = \Pi_0, \quad \begin{array}{ll} \Sigma_1, ..., & \Sigma_n, ... \\ \\ \Pi_1, ..., & \Pi_n, ... \end{array}$$

nennt man die **arithmetische Hierarchie**.

Beispiele. Die T-Prädikate T_n sind Σ_0 und Π_0, wie alle rekursiven Prädikate. K ist Σ_1, aber nicht Π_1, und die Menge R der Gödel-Nummern 1-stelliger total-rekursiver Funktionen,

$$e \in R \leftrightarrow \forall x \exists y T(e, x, y),$$

ist Π_2, wie man der definierenden Formel unmittelbar ansieht. R ist aber nicht Σ_2, was wir hier allerdings nicht zeigen.

Offenbar wachsen diese Klassen zumindest schwach monoton mit wachsendem n:

4.1.2 Lemma

Ist ein Prädikat P Σ_n oder Π_n, so ist P auch Σ_m und Π_m für alle $m > n$.

Beweis durch Einführen überflüssiger Quantoren. Gilt (1) für P und ist z eine neue Variable, so gilt auch

$$\begin{array}{rcll} P(\boldsymbol{y}) & \leftrightarrow & \overline{Q}_1 z Q_1 x_1 ... Q_n x_n & R(\boldsymbol{x}, \boldsymbol{y}) \\ & \leftrightarrow & Q_1 x_1 ... Q_n x_n \overline{Q}_n z & R(\boldsymbol{x}, \boldsymbol{y}), \end{array}$$

wobei $\overline{Q}_i$ im Falle $n = 0$ ein beliebiger und im Falle $n > 0$ der von Q_i verschiedene Quantor ist. Also ist P sowohl Π_{n+1} als auch Σ_{n+1}, und durch Wiederholung dieses Schrittes folgt die Behauptung.

4.1.3 Lemma

Ist das r-stellige Prädikat Q Σ_n (bzw. Π_n), sind $f_1, ..., f_r$ m-stellige rekursive Funktionen, und ist P definiert durch

$$P(\boldsymbol{y}) \leftrightarrow Q(f_1(\boldsymbol{y}), ..., f_r(\boldsymbol{y})),$$

so ist auch P Σ_n (bzw. Π_n).

Beweis durch lokale Einsetzung. Nach Voraussetzung gibt es ein rekursives Prädikat R und ein alternierendes Präfix $Q_1 x_1 ... Q_n x_n$ mit

$$Q(\boldsymbol{z}) \leftrightarrow Q_1 x_1 ... Q_n x_n R(\boldsymbol{x}, \boldsymbol{z}).$$

Das $(n + m)$-stellige Prädikat R' mit

$$R'(\boldsymbol{x}, \boldsymbol{y}) \leftrightarrow R(\boldsymbol{x}, f_1(\boldsymbol{y}), ..., f_r(\boldsymbol{y}))$$

ist dann nach Lemma 1.3.2 rekursiv. Also ist P Σ_n (bzw. Π_n) wegen

$$P(\boldsymbol{y}) \leftrightarrow Q_1 x_1 ... Q_n x_n R'(\boldsymbol{x}, \boldsymbol{y}).$$

4.1.4 Lemma

Es sei $n > 0$. Ist Q Σ_n und gilt

$$P(\boldsymbol{y}) \leftrightarrow \exists x Q(x, \boldsymbol{y}),$$

so ist auch P Σ_n. Ist Q Π_n und gilt

$$P(\boldsymbol{y}) \leftrightarrow \forall x Q(x, \boldsymbol{y}),$$

so ist auch P Π_n.

Beweis durch Quantorenkontraktion. Im ersten Fall gibt es ein Π_{n+1}-Prädikat R mit

$$\begin{array}{lcll} Q(x,\boldsymbol{y}) & \leftrightarrow & \exists z R(z,x,\boldsymbol{y}), & \text{also} \\ P(\boldsymbol{y}) & \leftrightarrow & \exists x \exists z R(z,x,\boldsymbol{y}) & \leftrightarrow \exists x R((x)_0,(x)_1,\boldsymbol{y}). \end{array}$$

Nach Lemma 4.1.3 ist dieser Kern immer noch Π_{n-1}, so dass P Σ_n ist. Im zweiten Fall schließt man entsprechend.

4.1.5 Lemma

Sind die m-stelligen Prädikate P und Q Σ_n (bzw. Π_n), so sind auch die Prädikate $P \vee Q$ und $P \wedge Q$ Σ_n (bzw. Π_n).

Beweis durch Induktion nach n.

1. $n = 0$. Nach Lemma 1.3.12 sind mit P und Q auch $P \vee Q$ und $P \wedge Q$ rekursiv.

2. $n > 0$. P und Q seien Σ_n. Dann gibt es $(m+1)$-stellige Π_{n-1}-Prädikate R und S mit

   $$P(\boldsymbol{y}) \leftrightarrow \exists x R(x,\boldsymbol{y}) \text{ und } Q(\boldsymbol{y}) \leftrightarrow \exists x S(x,\boldsymbol{y}).$$

   Nach Lemma 4.1.3 oder auch 1.3.1 sind die Prädikate R' und S' mit

   $$R'(x,z,\boldsymbol{y}) \leftrightarrow R(x,\boldsymbol{y}) \text{ und } S'(x,z,\boldsymbol{y}) \leftrightarrow S(z,\boldsymbol{y})$$

   Π_{n-1}, so dass nach Induktionsvoraussetzung auch $R \vee S$ und $R' \wedge S'$ Π_{n-1} sind. Dann gilt

   $$\begin{array}{lclcl} (P \vee Q)(\boldsymbol{y}) & \leftrightarrow & P(\boldsymbol{y}) \vee Q(\boldsymbol{y}) & \leftrightarrow & \exists x (R(x,\boldsymbol{y}) \vee S(x,\boldsymbol{y})) \\ & & & \leftrightarrow & \exists x (R \vee S)(x,\boldsymbol{y}) \quad \text{und} \\ (P \wedge Q)(\boldsymbol{y}) & \leftrightarrow & P(\boldsymbol{y}) \wedge Q(\boldsymbol{y}) & \leftrightarrow & \exists x \exists z (R(x,\boldsymbol{y}) \wedge S(z,\boldsymbol{y})) \\ & & & \leftrightarrow & \exists x \exists z (R' \wedge S')(x,z,\boldsymbol{y}). \end{array}$$

   Also ist $P \vee Q$ und nach Lemma 4.1.4 auch $P \wedge Q$ Σ_n. Sind P und Q Π_n-Prädikate, so schließt man entsprechend unter Vertauschung von $\exists, \vee$ mit $\forall, \wedge$.

4.1.6 Lemma

Ist das Prädikat P Σ_n (bzw. Π_n), dann ist das Prädikat $\neg P$ Π_n (bzw. Σ_n).

Beweis. Gilt (1) für P, so gilt

$$\neg P(\boldsymbol{y}) \leftrightarrow \overline{Q}_1 x_1 ... \overline{Q}_n x_n \neg R(\boldsymbol{x}, \boldsymbol{y}),$$

wobei $\overline{Q}_i$ der von Q_i verschiedene Quantor ist. Da mit R nach Lemma 1.3.12 auch $\neg R$ rekursiv ist, folgt die Behauptung.

Bemerkung. Nach den letzten drei Lemmata führen die logischen Partikel nicht aus der arithmetischen Hierarchie heraus. Zu einer Klasse mit notwendig höherem Index führt allenfalls noch das Vorschalten des Quantors, der vom nachfolgenden Quantor verschieden ist. Es stellt sich die Frage, ob sich auch in diesem Fall das Anwachsen des Präfixes beschränken läßt. Diese Frage wird im Hierarchie-Theorem 4.1.8 negativ beantwortet.

Wie die r.a. Mengen nach dem Aufzählungs-Theorem 2.2.4 von einem einzigen 2-stelligen r.a. Prädikat aufgezählt werden wegen

$$x \in W_e \leftrightarrow \exists y T(e, x, y),$$

so gilt hier, dieses Resultat verallgemeinernd:

4.1.7 Arithmetisches Aufzählungs-Theorem.

Zu jedem $n > 0$ und m gibt es ein $(m+1)$-stelliges Σ_n-Prädikat (Π_n-Prädikat) Q, das alle m-stelligen Σ_n-Prädikate (bzw. Π_n-Prädikate) im folgenden Sinne aufzählt: Ein m-stelliges Prädikat P ist genau dann Σ_n (bzw. Π_n), wenn es eine natürliche Zahl e gibt, für die stets gilt:

(2) $\quad P(\boldsymbol{y}) \leftrightarrow Q(e, \boldsymbol{y}).$

Beweis.
1. Fall. Wir betrachten die Σ_n-Prädikate für ungerades n und die Π_n-Prädikate für gerades $n > 0$; das sind genau die Fälle, bei denen in (1) der letzte Quantor ein Existenzquantor $\exists$ ist:

$$P(\boldsymbol{y}) \leftrightarrow Q_1 x_1 ... \exists x_n R(\boldsymbol{x}, \boldsymbol{y})$$

mit alternierendem Präfix und mit r.a. $\exists x_n R(\boldsymbol{x}, \boldsymbol{y})$. Nach dem Aufzählungstheorem 2.2.4 gibt es eine Zahl e mit

$$\exists x_n R(\boldsymbol{x}, \boldsymbol{y}) \leftrightarrow \exists x_n T_{n+m-1}(e, \boldsymbol{y}, \boldsymbol{x}).$$

Definieren wir ein $(m + 1)$-stelliges Prädikat Q durch

(3) $\quad Q(z, \boldsymbol{y}) \leftrightarrow Q_1 x_1 ... \exists x_n T_{m+n-1}(z, \boldsymbol{y}, \boldsymbol{x})$

mit alternierendem Präfix, so ist Q wieder Σ_n, falls n ungerade ist, und Π_n, falls $n > 0$ gerade ist, und für P gilt (2). Also gibt es zu jedem solchen P eine Zahl e mit (2), und umgekehrt liegt für jede Zahl e das durch (2) definierte Prädikat P nach Lemma 4.1.3 in derselben Klasse wie Q.

2. Fall. Wenn das Prädikat P Σ_n ist, falls $n > 0$ gerade ist, und Π_n ist, falls n ungerade ist, wenn also in der Darstellung (1) von P der letzte Quantor ein Allquantor $\forall$ ist, dann fällt dessen Negation $\neg P$ nach Lemma 4.1.6 unter den 1. Fall. Zu P gibt es also eine Zahl e, so dass für das durch (3) definierte Q stets gilt

$$\neg P(\boldsymbol{y}) \leftrightarrow Q(e, \boldsymbol{y}), \text{ also } P(\boldsymbol{y}) \leftrightarrow \neg Q(e, \boldsymbol{y}).$$

Für gerades (ungerades) $n > 0$ ist $\neg Q$ nach Lemma 4.1.6 selbst Σ_n (Π_n) und zählt hiernach genau alle Σ_n- (Π_n-)Prädikate auf. Damit ist der Satz bewiesen.

Die Menge K ist offenbar Σ_1; sie ist nicht Σ_0, weil ihr Komplement $\neg K$ von der Aufzählung $\{W_e | e \in \mathbb{N}\}$ aller Σ_1-Prädikate nicht erfaßt wird und deshalb zwar Π_1, aber nicht Σ_1 ist. In Verallgemeinerung dieser Methode zeigen wir, dass für $n > 0$ nur die in Lemma 4.1.2 angegebenen Inklusionen zwischen den Klassen Σ_n und Π_n gelten.

4.1.8 Arithmetisches Hierarchie-Theorem (Kleene-Mostowski).

Für jedes $n > 0$ gibt es ein einstelliges Σ_n-Prädikat P, das nicht Π_n ist und daher weder Σ_m noch Π_m für alle $m < n$ ist. Das Prädikat $\neg P$ ist dann Π_n, aber nicht Σ_n und daher weder Σ_m noch Π_m für alle $m < n$.

Beweis. Es sei Q ein 2-stelliges Σ_n-Prädikat, das die Klasse aller 1-stelligen Σ_n-Prädikate nach Satz 4.1.7 aufzählt, und P sei definiert durch

$$P(x) \leftrightarrow Q(x, x).$$

Dann ist P ein Σ_n-Prädikat nach Lemma 4.1.3, aber $\neg P$ wird nach dem 1. Cantorschen Diagonalverfahren von Q nicht aufgezählt. Denn wenn es eine Zahl e gäbe, so dass für alle x

$$\neg P(x) \leftrightarrow Q(e, x)$$

wäre, so folgte für $x = e$ der Widerspruch

$$\neg P(e) \leftrightarrow Q(e, e) \leftrightarrow P(e)$$

Also ist $\neg P$ nach dem Aufzählungs-Theorem 4.1.7 kein Σ_n-Prädikat. Nach Lemma 4.1.6 ist $\neg P$ ein Π_n-Prädikat und P kein Π_n-Prädikat. Der Rest folgt mit Lemma 4.1.2

Um die Hierarchie besser darstellen zu können, führen wir die Durchschnitte der "benachbarten" Klassen Σ_n und Π_n ein:

4.1.9 Definition

$\Delta_n := \Sigma_n \cap \Pi_n$, ein Prädikat ist Δ_n, wenn es sich sowohl als Σ_n- als auch als Π_n-Prädikat darstellen läßt.

4.1.10 Korollar

Für $n > 0$ ist Δ_n echt in Σ_n und echt in Π_n enthalten, und Σ_n und Π_n sind echt in Δ_{n+1} enthalten.

Denn das Enthaltensein ist (nach Lemma 4.1.2 klar, und wäre $\Delta_n = \Sigma_n$, so wäre $\Sigma_n \subseteq \Pi_n$, entgegen Satz 4.1.8. Wäre $\Sigma_n = \Delta_{n+1}$, so wäre nach Lemma 4.1.6 $\Pi_n = \Delta_{n+1} = \Sigma_n$ wegen

$$P \in \Pi_n \iff \neg P \in \Sigma_n \iff \neg P \in \Delta_{n+1} \iff P \in \Delta_{n+1},$$

entgegen Satz 4.1.8.

Die arithmetische Hierarchie läßt sich also folgendermaßen veranschaulichen:

$$
\begin{array}{ccccccc}
 & & \Sigma_1 \subset \cdots \subset \Sigma_n & & & & \Sigma_{n+1} \subset \dots \\
 & \cup & & \cap & & \cup & \\
\Delta_0 = \Sigma_0 = \Pi_0 = \Delta_1 & & & & \Delta_{n+1} & & \\
 & \cap & & \cup & & \cap & \\
 & & \Pi_1 \subset \cdots \subset \Pi_n & & & & \Pi_{n+1} \subset \dots
\end{array}
$$

Hierin bezeichnet $\subset$ stets das echte Enthaltensein. Ganz am Anfang ist die Klasse der rekursiven Prädikate trivialerweise gleich $\Delta_0 = \Sigma_0 = \Pi_0$ und nach dem Satz von Post auch gleich Δ_1, aber danach fallen keine zwei Klassen zusammen.

Per Definition ist Δ_n der Durchschnitt von Σ_n und Π_n. In der anderen Richtung lässt das Hierarchie-Theorem offen, ob möglicherweise Δ_{n+1} die Vereinigung von Σ_n und Π_n ist. Das ist für $n > 0$ nicht der Fall:

4.1.11 Korollar

Für $n > 0$ ist $\Sigma_n \cup \Pi_n$ echt in Δ_{n+1} enthalten.

Beweis. Nach dem Hierarchie-Theorem gibt es ein einstelliges Δ_{n+1}-Prädikat P , das nicht Σ_n ist. Dessen Komplement $\neg P$ ist nach 4.1.6 Δ_{n+1} , aber nicht Π_n . Die disjunkte Vereinigung von P und $\neg P$, also die Menge

$$Q = \{\langle 0, x\rangle \mid P(x)\} \cup \{\langle 1, x\rangle \mid \neg P(x)\}$$

ist nach 4.1.3 und 5 auch Δ_{n+1} , aber weder Σ_n noch Π_n . Denn wäre $Q \in \Sigma_n$, so wäre auch

$$P = \{x \mid \langle 0, x\rangle \in Q\} \in \Sigma_n$$

nach 4.1.3, im Widerspruch zur Wahl von P. Ebenso folgte aus $Q \in \Pi_n$, dass $\neg P \in \Pi_n$ und daher nach 4.1.6 wieder $P \in \Sigma_n$ wäre.

4.2 Entbehrlichkeit der primitiven Rekursion

Wie im Anschluss an Lemma 4.1.6 bemerkt, treten alle arithmetischen Prädikate in der arithmetischen Hierarchie auf. Das Umgekehrte folgt unmittelbar aus dem Satz:

Alle rekursiven Prädikate sind arithmetisch.

Beim Beweis dieses Satzes macht das Schema der primitiven Rekursion große Schwierigkeiten, der μ-Operator dagegen nicht. Man sucht deshalb nach einer anderen Definition der μ-rekursiven Funktionen, die mehr Anfangsfunktionen und den μ-Operator im Normalfall, aber nicht die primitive Rekursion verwendet.

4.2.1 Induktive Definition der streng μ-rekursiven Funktionen.

1. Alle Funktionen U_i^n und C_k^n und die Funktionen $+, \cdot$ und $\chi_<$ (charakteristische Funktion der Kleiner-Relation) sind streng μ-rekursiv.
2. Ist h r-stellig und $g_1, ..., g_r$ n-stellig streng μ-rekursiv, so ist auch $h \circ (g_1, ..., g_r)$ n-stellig streng μ-rekursiv.
3. Ist g $(n+1)$-stellig streng μ-rekursiv und gilt
$$\forall \boldsymbol{x} \exists y\ g(\boldsymbol{x}, y) = 0,$$
so ist auch μg streng μ-rekursiv.

Ein Prädikat P heißt **streng μ-rekursiv**, wenn seine charakteristische Funktion χ_P streng μ-rekursiv ist.

Bemerkung. Alle streng μ-rekursiven Funktionen sind also μ-rekursiv, insbesondere total.

Nach 2. und 3. sind die streng μ-rekursiven Funktionen abgeschlossen gegen Komposition und Anwendung des μ-Operators im Normalfall. Zu zeigen bleibt, dass sie auch gegen primitive Rekursion abgeschlossen sind. Die Ergebnisse, die wir in 1.3 für primitiv-rekursive Funktionen bewiesen haben, wollen wir hier für streng μ-rekursive Funktionen gewinnen, u. U. in veränderter Reihenfolge. Die Lemmata 1.3.1 und 2 handeln nur von den U_i^n und der Komposition und gelten deshalb auch hier.

4.2.2 Lemma

Hinzunahme, Vertauschung und Identifikation von Variablen sind streng μ-rekursive Operationen, d. h. ist

$$f(\boldsymbol{x}) = g(\boldsymbol{x}')$$

und jedes x_i' ein x_j, so ist mit g auch f streng μ-rekursiv.

4.2.3 Lemma

Lokale Einsetzung ist eine streng μ-rekursive Operation, d. h. ist

$$f(\boldsymbol{x}) = g(\boldsymbol{x}_1', h(\boldsymbol{x}_2'), \boldsymbol{x}_3')$$

und ist jedes x_{ik}' ein x_j, so ist mit g und h auch f streng μ-rekursiv.

Die Beweise können wörtlich aus 1.3 übernommen werden.

Bemerkung. Diese Ergebnisse gelten auch für Prädikate; ist z. B.

$$P(x, y, z) \leftrightarrow Q(x, h_1(x, y, z), h_2(x, z)),$$

so ist mit Q, h_1 und h_2 auch P streng μ-rekursiv. Zum Beweis betrachtet man χ_P und χ_Q und verwendet zweimal Lemma 4.2.3.

Die Lemmata 4.2.2 und 3 werden im folgenden häufig ohne Erwähnung angewandt.

4.2.4 Lemma

Die aussagenlogischen Operationen sind streng μ-rekursiv; d. h. mit P ist $\neg P$, mit P und Q ist $P \vee Q$ streng μ-rekursiv.

Denn es ist

$$\begin{array}{rcl} \chi_{\neg P}(\boldsymbol{x}) & = & \chi_{<}(0, \chi_P(\boldsymbol{x})) \text{ und} \\ \chi_{P \vee Q}(\boldsymbol{x}) & = & \chi_P(\boldsymbol{x}) \cdot \chi_Q(\boldsymbol{x}). \end{array}$$

Die anderen Junktoren sind mit $\neg$ und $\vee$ definierbar.

4.2.5 Lemma

Die Prädikate $<, \leq$ und $=$ sind streng μ-rekursiv.

Denn $<$ ist nach Definition 4.2.1, und $\leq$ und $=$ sind wegen

$$x \leq y \leftrightarrow \neg y < x \quad \text{und} \quad x = y \leftrightarrow x \leq y \wedge y \leq x$$

nach Lemma 4.2.4 und 4.2.2 streng μ-rekursiv.

4.2.6 Lemma

Beschränkte Quantoren und der beschränkte μ-Operator sind streng μ-rekursive Operationen, d. h. ist

$$\begin{array}{rcl} P(\boldsymbol{x}, y) & \leftrightarrow & (\exists i < y) Q(\boldsymbol{x}, i) \quad \text{bzw.} \\ P(\boldsymbol{x}, y) & \leftrightarrow & (\forall i < y) Q(\boldsymbol{x}, i) \quad , \\ f(\boldsymbol{x}, y) & = & (\mu i < y) Q(\boldsymbol{x}, i) \quad , \end{array}$$

so ist mit Q auch P bzw. f streng μ-rekursiv.

Beweis. Es ist

$$\begin{array}{rcl} (\exists i < y) Q(\boldsymbol{x}, i) & \leftrightarrow & \mu i (Q(\boldsymbol{x}, i) \vee i = y) < y, \\ (\forall i < y) Q(\boldsymbol{x}, i) & \leftrightarrow & \neg \exists i < y \neg Q(\boldsymbol{x}, i) \quad \text{und} \\ (\mu i < y) Q(\boldsymbol{x}, i) & = & \mu i (Q(\boldsymbol{x}, i) \vee i = y)). \end{array}$$

Also sind mit Q auch die definierten Prädikate und Funktionen nach Definition 4.2.1 - der μ-Operator wird nur im Normalfall angewandt - und Lemma 4.2.4 und 5 streng μ-rekursiv.

4.2.7 Lemma

$\dot{-}$ ist streng μ-rekursiv.

Denn es ist $x \dot{-} y = \mu i(y + i = x \vee (x < y \wedge i = 0)$, und die Behauptung folgt mit Lemma 4.2.4 und 5.

Die in 1.3 eingeführte Paarfunktion ist zu kompliziert, sie wächst zu rasch, als dass wir sie an dieser Stelle schon als streng μ-rekursiv erkennen könnten. Wir greifen auf die Bijektion $\pi : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ aus der Bemerkung 1.3.20 zurück. Beim 2. Cantorschen Diagonalverfahren werden jeweils alle Paare (x, y) mit konstanter Summe $s = x + y$ nacheinander nach der Größe von x aufgezählt. Das x-te solche Paar $(x, s \dot{-} x)$ $(x = 0, ..., s)$ erhält dann die Nummer

$$\pi(x, s \dot{-} x) = \frac{1}{2} \cdot s \cdot (s + 1) + x$$

π ist bijektiv, weil es einerseits gerade $s + 1$ Paare mit Komponentensumme s gibt und andererseits ebenfalls $s + 1$ Zahlen, die $\geq \frac{1}{2} \cdot s \cdot (s + 1)$ und $< \frac{1}{2} \cdot (s + 1) \cdot (s + 2)$ sind. Also besitzt π zwei einstellige Inverse.

4.2.8 Definition

Wie bisher bezeichne π die Bijektion von $\mathbb{N} \times \mathbb{N}$ auf $\mathbb{N}$, gegeben durch

$$\pi(x, y) = \frac{1}{2} \cdot (x + y) \cdot (x + y + 1) + x$$

Dann sind zwei einstellige Inverse von π definiert durch

$$\pi_1(\pi(x, y)) = x \quad \text{und} \quad \pi_2(\pi(x, y)) = y$$

4.2.9 Lemma

Die Funktionen π, π_1 und π_2 sind streng μ-rekursiv.

Beweis. Da von zwei aufeinander folgenden Zahlen stets eine gerade ist, gibt es stets (genau) ein z mit

$$2 \cdot z = (x + y) \cdot (x + y + 1) + 2x.$$

Also ist

$$\pi(x, y) = \mu z(2 \cdot z = (x + y) \cdot (x + y + 1) + 2 \cdot x),$$

und π ist nach 4.2.5 streng μ-rekursiv. Umgekehrt ist

$$x + y = \mu s(2 \cdot \pi(x, y) < (s + 1) \cdot (s + 2)),$$

so dass für die streng μ-rekursive Funktion σ mit

$$\sigma(z) = \mu s(2 \cdot z < (s+1) \cdot (s+2))$$
$$\sigma(\pi(x,y)) = x + y$$

ist. Dann können wir aber auch π_1 und π_2 durch

$$\pi_1(z) = z \dot{-} \tfrac{1}{2} \cdot \sigma(z) \cdot (\sigma(z)+1) \quad \text{und}$$
$$\pi_2(z) = \sigma(z) \dot{-} \pi_1(z)$$

streng μ-rekursiv definieren, denn dann ergibt sich

$$\pi_1(\pi(x,y)) = \pi(x,y) \dot{-} \tfrac{1}{2} \cdot (x+y) \cdot (x+y+1) = x,$$
$$\pi_2(\pi(x,y)) = (x+y) \dot{-} x = y.$$

Die Funktion π und ihre Umkehrfunktionen sind hiernach Paar- und Projektionsfunktionen, die sich mit besonders geringen Mitteln angeben lassen, z. B. ohne Kenntnisse über Primzahlen und die Eindeutigkeit der Primfaktorzerlegung.

4.2.10 Lemma

Der ganze Anteil des Quotienten $[-]$ und der Rest *res* sind streng μ-rekursive Funktionen.

Beweis. Es ist

$$\left[\frac{x}{y}\right] = (\mu i < x+1)(\exists j < y)(x = i \cdot y + j)$$

und

$$res(x,y) = x \dot{-} \left[\frac{x}{y}\right] \cdot y.$$

Speziell ist dann $\left[\frac{x}{0}\right] = x+1$ und $res(x,0) = x$. Nach Lemma 4.2.5 und 6 ist $[-]$, nach Lemma 4.2.7 auch *res* streng μ-rekursiv.

Wir ziehen jetzt einen einfachen Sachverhalt aus der elementaren Zahlentheorie heran, nach dem geeignete Tupelmengen durch Restsysteme aufgezählt werden.

Beispiel. Wir vergleichen die Menge T aller Paare (a,b) mit $a < 2, b < 3$ mit der Menge R der von $i = 0, ..., 5$, also von $i < 2 \cdot 3$ erzeugten "Restsysteme" $r_i = (res(i,2), res(i,3))$ modulo 2 und 3:

$$\begin{array}{ll} (0,0) = r_0 & (1,0) = r_3 \\ (0,1) = r_4 & (1,1) = r_1 \\ (0,2) = r_2 & (1,2) = r_5 \end{array}$$

In diesem Fall ist $T = R$. Nehmen wir statt 2 und 3 die Zahlen 2 und 4, so können Paare wie $(0,1), (0,3), (1,0), (1,2)$ nicht als Restsysteme modulo 2 und 4 auftreten; es ist $T \neq R$, weil 2 und 4 nicht teilerfremd sind.

4.2.11 Chinesischer Restsatz

Es seien $d_0, ..., d_n$ paarweise teilerfremde Zahlen. Dann ist die Menge der Restsysteme

$$R := \{(res(c, d_0), ..., res(c, d_n)) | c < d_0 \cdot ... \cdot d_n\}$$

gleich der Tupelmenge

$$T := \{(a_0, ..., a_n) | a_i < d_i \text{ für alle } i \leq n\}.$$

Beweis. Ist ein $d_i = 0$, so sind R und T leer. Sonst ist stets $res(c, d_i) < d_i$, so dass $R \subseteq T$ ist. Zu zeigen bleibt, dass verschiedene $c < d_0 \cdot ... \cdot d_n$ verschiedene Restsysteme $\in R$ definieren, d. h.: Ist $res(c, d_i) = res(c', d_i)$ für alle $i \leq n$, und $c' \leq c < d_0 \cdot ... \cdot d_n$, so ist $c = c'$. Dies ist der Fall, weil dann $res(c - c', d_i) = 0$, also $d_i | c - c'$ für $i \leq n$, gilt. Da die d_i teilerfremd sind, folgt

$$d_0 \cdot ... \cdot d_n | c - c' < d_0 \cdot ... \cdot d_n,$$

also $c - c' = 0$ und $c = c'$. Also enthält R $d_0 \cdot ... \cdot d_n$ Elemente, und das sind ebenso viele wie in T : Die endlichen Mengen R und T stimmen überein.

Dieser Satz wurde von Gödel 1931 verwendet, um Zahlentupel als Restsysteme einer einzigen Zahl darzustellen.

4.2.12 Definition der Gödel'schen β-Funktion.

Die Funktionen β' und β sind definiert durch

$$\begin{aligned} \beta'(x, y, i) &= res(x, y \cdot (i+1) + 1), \\ \beta(z, i) &= \beta'(\pi_1(z), \pi_2(z), i). \end{aligned}$$

4.2.13 Lemma

β' und β sind streng μ-rekursiv. Es ist stets

$$\beta(\pi(x, y), i) = res(x, y \cdot (i+1) + 1).$$

Beweis. β' ist nach Lemma 4.2.10 streng μ-rekursiv; deswegen und nach Lemma 4.2.9 ist auch β streng μ-rekursiv. Die behauptete Gleichung folgt unmittelbar aus den Gleichungen in Lemma 4.2.9.

4.2.14 Satz (Gödel 1931)

Zu jedem n und jedem $(n+1)$-Tupel $a_0, ..., a_n$ gibt es Zahlen c und d, so dass für alle $i \leq n$ gilt

$$\beta(\pi(c, d), i) = a_i.$$

Beweis. Gegeben seien n und $a_0, ..., a_n$. Wir setzen

$$d := (\max \{n, a_0, ..., a_n\})! \text{ und } d_i := d \cdot (i+1) + 1 \text{ für } i \leq n.$$

Wir behaupten, die d_i sind paarweise teilerfremd. Denn angenommen, eine Primzahl p teilt d_i und d_j mit $i < j$. Dann gilt

$$p | d_j - d_i = d \cdot (j - i) \text{ und } 0 < j - i \leq n.$$

Da p prim ist, folgt $p|d$ oder $p|j-i$. Aus $p|j-i$ und $0 < j-i \leq n$ folgt $p \leq n$, also auch $p|d$ nach Definition von d. Aus $p|d$ folgt aber $p|d \cdot (i+1) = d_i - 1$, im Widerspruch zur Annahme $p|d_i$. Also sind die d_i paarweise teilerfremd.

Nach dem chinesischen Restsatz gibt es daher eine Zahl $c < d_0 \cdot ... \cdot d_n$ mit

$$a_i = res(c, d_i) = res(c, d \cdot (i+1) + 1) = \beta(\pi(c, d), i)$$

für alle $i \leq n$. Damit ist der Satz bewiesen.

Bemerkung. Die Funktion β ist hiernach zur Tupel-Dekodierung geeignet, wie die Funktion $x, i \mapsto (x)_i$ in 1.3. Sie zählt die Elemente jedes gegebenen Tupels durch das 2. Argument auf, wenn man das 1. Argument geeignet (und groß genug) wählt. Allerdings hat unter den Voraussetzungen des Satzes die Funktion $i \mapsto \beta(\pi(c, d), i)$ noch Werte $\neq 0$ für Argumente $> n$. Es ist daher zweckmäßig, die Länge n des zu kodierenden Tupels in die Kode-Nummer des Tupels aufzunehmen. Für jedes feste n kann man alternativ zu 1.3 eine n-stellige Kodierfunktion $\underline{\langle}, ..., \underline{\rangle}$ wie folgt definieren:

$$\underline{\langle} x_0, ..., x_{n-1} \underline{\rangle} = \mu z(\beta(z, 0) = n \wedge \beta(z, 1) = x_0 \wedge ... \wedge \beta(z, n) = x_{n-1}).$$

Mit Satz 4.2.14 können wir zeigen, dass auch die primitive Rekursion eine streng μ-rekursive Operation ist.

4.2.15 Satz

Alle μ-rekursiven Funktionen sind streng μ-rekursiv, und umgekehrt.

Beweis. Da $+, \cdot$ und $\chi_<$ primitiv-rekursiv sind, folgt die Umkehrung unmittelbar durch Induktion nach der Definition der streng μ-rekursiven Funktionen.

Für die Hauptrichtung benutzen wir, dass nach dem Normalform-Theorem 2.1 jede μ-rekursive Funktion f eine Darstellung

$$f(\boldsymbol{x}) = U(\mu y T_n(e, \boldsymbol{x}, y))$$

besitzt, wobei der μ-Operator im Normalfall angewandt ist. Nach Lemma 4.2.3 genügt es also, das T-Prädikat T_n und die Funktion U als streng μ-rekursiv nachzuweisen. Wir zeigen allgemeiner durch Induktion nach der Definition der primitiv-rekursiven Funktionen:

Alle primitiv-rekursiven Funktionen sind streng μ-rekursiv.

1. Die Ausgangsfunktionen U_i^n, C_k^n und $suc = + \circ (U_1^1, C_1^1)$ sind streng μ-rekursiv.

2. Mit $h, g_1, ..., g_r$ ist auch $h \circ (g_1, ..., g_r)$ streng μ-rekursiv.

3. Sei $f = Rgh$ und seien g und h streng μ-rekursiv. Wir definieren ein Prädikat P durch

$$P(\boldsymbol{x}, y, z) \leftrightarrow \beta(z, 0) = g(\boldsymbol{x}) \wedge \forall i < y \; \beta(z, i+1) = h(\boldsymbol{x}, i, \beta(z, i)).$$

Dann ist P nach Lemma 4.2.4, 5, 6 und 4.2.13 streng μ-rekursiv, und es gilt:

$$(1) \qquad P(\boldsymbol{x}, y, z) \leftrightarrow \forall i \leq y f(\boldsymbol{x}, i) = \beta(z, i)$$

Denn gilt $P(\boldsymbol{x}, y, z)$, so ist $f(\boldsymbol{x}, 0) = g(\boldsymbol{x}) = \beta(z, 0)$, und aus $i < y$ und $f(\boldsymbol{x}, i) = \beta(z, i)$ folgt

$$f(\boldsymbol{x}, i+1) = h(\boldsymbol{x}, i, f(\boldsymbol{x}, i)) = h(\boldsymbol{x}, i, \beta(z, i)) = \beta(z, i+1),$$

so dass Induktion nach i die rechte Seite von (1) ergibt. Gilt umgekehrt $\forall i \leq y f(\boldsymbol{x}, i) = \beta(z, i)$, so ist

$$\beta(z, 0) = f(\boldsymbol{x}, 0) = g(\boldsymbol{x}),$$

und für $i < y$ ist $i + 1 \leq y$, also

$$\beta(z, i+1) = f(\boldsymbol{x}, i+1) = h(\boldsymbol{x}, i, f(\boldsymbol{x}, i)) = h(\boldsymbol{x}, i, \beta(z, i)).$$

(2) $\forall \boldsymbol{x}, y \exists z P(\boldsymbol{x}, y, z)$

Denn nach Satz 4.2.14 gibt es zu $\boldsymbol{x}, y, f(\boldsymbol{x}, 0), ..., f(\boldsymbol{x}, y)$ Zahlen c, d mit

$$\beta(\pi(c,d), i) = f(\boldsymbol{x}, i) \text{ für } i \leq y.$$

Es gibt also eine Zahl $z = \pi(c,d)$, so dass gilt

$$\forall i \leq y \; \beta(z, i) = f(\boldsymbol{x}, i),$$

und dies ist wegen der Äquivalenz (1) die Behauptung (2).

(3) $f(\boldsymbol{x}, y) = \beta(\mu z P(\boldsymbol{x}, y, z), y)$

Das folgt aus (1). Wegen (2) wird in (3) der μ-Operator im Normalfall angewandt. Demnach ist mit P und β auch f streng μ-rekursiv, und das war zu beweisen.

Also sind alle primitiv-rekursiven Funktionen streng μ-rekursiv.

Damit ist der Satz bewiesen.

Hiernach lassen sich die μ-partiell-rekursiven Funktionen auch ohne das Schema der primitiven Rekursion einführen, wenn man nur die Ausgangsfunktionen etwas anreichert. Das ist ein geeigneter Ausgangspunkt, um die rekursiven Funktionen und Prädikate als in der Arithmetik definierbar, sogar als repräsentierbar nachzuweisen.

4.3 $\Sigma(\mathcal{N})$-Definitionen und Repräsentierbarkeit

Wir können nun die in der Einleitung dieses Kapitels erwähnte Aufgabe lösen, die rekursiven Prädikate - und damit alle Prädikate in der arithmetischen Hierarchie - durch Formeln der Zahlentheorie zu definieren. Hauptgegenstand und Standardmodell der Zahlentheorie ist die **Struktur der natürlichen Zahlen**

$$\mathcal{N} = (\mathbb{N}\,; 0, suc, +, \cdot\,; <)$$

bestehend aus der Menge $\mathbb{N}$ der natürlichen Zahlen, der Zahl Null 0, der Nachfolgerfunktion suc, der Addition $+$, der Multiplikation $\cdot$ und der Kleiner-Relation $<$. Der Formulierung elementarer Sachverhalte in dieser Struktur dient die folgende Sprache.

4.3.1 Definition der Sprache $L(Z)$ der Zahlentheorie.

$L(Z)$ enthält die nicht-logischen Grundzeichen

$$0 \text{ (Null)}\ , suc \text{ (Nachfolger)}\ , +, \cdot, < \ \text{(kleiner)}\ .$$

Als **logische Grundzeichen** verwenden wir

- **freie Variablen**, mitgeteilt durch $a, b, ...,$
- **gebundene Variablen**, mitgeteilt durch $x, y, ...,$
- das **Gleichheitszeichen** $=$
- und die **logischen Partikel** $\neg, \wedge, \vee, \rightarrow, \forall, \exists$, hilfsweise Klammern (,).

Neben den Grundzeichen verwenden wir **Nennzeichen** $*_1, *_2, ...$. Eine n-**stellige Nennform** ist eine endliche Zeichenreihe aus Grundzeichen und höchstens den Nennzeichen $*_1, ..., *_n$. Nennformen werden durch $F, G, t, ...$ mitgeteilt. $F(G_1, ..., G_n)$ bezeichnen die Nennform, die aus F hervorgeht, indem man in F überall $*_i$ durch G_i ersetzt $(i = 1, ..., n)$. Eine Nennform heißt **geschlossen**, wenn in ihr keine freie Variable auftritt.

Terme von $L(Z)$ sind die freien Variablen, die 0, mit t auch $suct$ und mit s und t auch $(s + t)$ und $(s \cdot t)$. Die **Ziffern** sind die Terme

$$0, suc0, sucsuc0, ...,$$

die wir als Namen von natürlichen Zahlen benutzen und, ebenso wie diese, durch $k, l, ...$ mitteilen. Beliebige Terme teilen wir durch $r, s, t, ...$ mit.

Primformeln von $L(Z)$ sind die **Gleichungen** $s = t$ und die **Ungleichungen** $s < t$ mit Termen s, t.

Die **Formeln** von $L(Z)$ werden aus Primformeln mit den logischen Partikeln bei geeigneter Klammersetzung gebildet.

$(A \leftrightarrow B)$ steht für $((A \rightarrow B) \wedge (B \rightarrow A))$, und $\forall x < tF(x)$ ist Abkürzung für $\forall x(x < t \rightarrow F(x))$. Klammern werden fortgelassen, wenn es nicht zu Missverständnissen führt. Dabei binden die einstelligen Partikel $\neg, \forall, \exists$ am stärksten, $\wedge$ und $\vee$ binden stärker als $\rightarrow$ und $\leftrightarrow$, und bei gleichartigen Junktoren ist Rechtsklammerung zu ergänzen: $A \rightarrow B \rightarrow C$ steht für $(A \rightarrow (B \rightarrow C))$.

Die geschlossenen Formeln und Terme aus $L(Z)$ werden in natürlicher, klassischer Weise in der Struktur $\mathcal{N}$ interpretiert. Ist C eine geschlossene Formel aus $L(Z)$, die in $\mathcal{N}$ **gilt** (oder, was dasselbe ist, in $\mathcal{N}$ **wahr** ist), so schreiben wir $\mathcal{N} \models C$.

Wir interessieren uns in diesem Paragraphen hauptsächlich für spezielle Formeln, die schon äußerlich etwas mit Σ_1-Prädikaten zu tun haben. In diesen Formeln darf die Negation nur vor Primformeln, der Allquantor nur in beschränkter Form $\forall x < t$ und die Implikation sonst gar nicht auftreten.

4.3.2 Induktive Definition der $\boldsymbol{\Sigma(\mathcal{N})}$-Formeln.

1. Primformeln $s = t$ und $s < t$ sowie ihre Negationen $\neg s = t$ und $\neg s < t$ sind $\Sigma(\mathcal{N})$-Formeln.
2. Mit A und B sind auch $A \wedge B$ und $A \vee B$ $\Sigma(\mathcal{N})$-Formeln.
3. Mit $F(a)$ sind auch $\forall x < tF(x)$ und $\exists xF(x)$ $\Sigma(\mathcal{N})$-Formeln.

Es sind diese $\Sigma(\mathcal{N})$-Formeln, die wir zur Definition der Σ_1- bzw. r.a. Prädikate heranziehen wollen.

4.3.3 Definition

Eine n-stellige geschlossene Nennform F aus $L(Z)$ nennen wir eine **$\boldsymbol{\Sigma(\mathcal{N})}$ -Definition** eines n-stelligen Prädikats P, wenn

(i) $F(a_1, ..., a_n)$ eine $\Sigma(\mathcal{N})$-Formel ist und

(ii) für alle $k_1, ..., k_n \in \mathbb{N}$ gilt

(1) $P(k_1, ..., k_n) \Leftrightarrow \mathcal{N} \models F(k_1, ..., k_n)$.

P ist **$\boldsymbol{\Sigma(\mathcal{N})}$** oder ein **$\boldsymbol{\Sigma(\mathcal{N})}$ -Prädikat**, wenn es eine $\Sigma(\mathcal{N})$-Definition von P gibt. P ist $\boldsymbol{\Delta}(\mathcal{N})$, wenn sowohl P als auch $\neg P$ $\Sigma(\mathcal{N})$ sind.

4.3.4 Lemma

Alle $\Sigma(\mathcal{N})$-Prädikate sind rekursiv aufzählbar:

$$\Sigma(\mathcal{N}) \subseteq \Sigma_1$$

Beweis. Sei F eine $\Sigma(\mathcal{N})$-Definition eines Prädikats P. Wir induzieren nach dem Aufbau der $\Sigma(\mathcal{N})$-Formel $F(a_1, ..., a_n)$. Wir setzen $\mathbf{a} := a_1, ..., a_n$ und $\mathbf{k} := k_1, ..., k_n$.

1. Ist $F(\mathbf{a})$ eine Primformel aus $L(Z)$ oder deren Negation, so definiert F durch (1) offenbar ein primitiv-rekursives P.

2. F sei $F_1 \wedge F_2$ oder $F_1 \vee F_2$. Nach Induktionsvoraussetzung definieren die F_i $(i = 1, 2)$ r. a. Prädikate P_i. Dann ist $P = P_1 \cap P_2$ bzw. $P = P_1 \cup P_2$, und P ist wieder r. a. nach Lemma 4.1.5.

3. F sei $\forall x < tG(x, *_1, ..., *_n)$. G ist eine $\Sigma(\mathcal{N})$-Definition von einem Prädikat Q, das nach Induktionsvoraussetzung r. a. ist. Also gibt es eine partiell-rekursive Funktion g, deren Definitionsbereich *dom* $g = Q$ ist. Da die Term-Nennform t eine primitiv-rekursive Funktion (nämlich ein Polynom) definiert, ist dann auch die partielle Funktion f mit

   $$f(\mathbf{k}) \simeq \sum_{i<t(\mathbf{k})} g(i, \mathbf{k})$$

   partiell-rekursiv, so dass $dom f$ r.a. ist. Wegen

   $$k \in dom f \Leftrightarrow \forall i < t(\mathbf{k})Q(i, \mathbf{k}) \Leftrightarrow \mathcal{N} \models \forall x < t(\mathbf{k})G(x, \mathbf{k}) \Leftrightarrow P(\mathbf{k})$$

   ist also $P = dom f$ r.a.

4. F sei $\exists xG(x, *_1, ..., *_n)$. Nach Induktionsvoraussetzung definiert G ein r.a.Prädikat. Nach Lemma 2.2.4 definiert dann auch F ein r.a. Prädikat.

Durch Induktion nach $F(\mathbf{a})$ ist das Lemma damit vollständig bewiesen. Wegen des Satzes 2.2.5 von Post können wir ergänzen:

Korollar. Alle $\Delta(\mathcal{N})$-Prädikate sind rekursiv:

$$\Delta(\mathcal{N}) \subseteq \Delta_1$$

4.3.5 Lemma

$\Sigma(\mathcal{N})$-definierbare Graphen totaler Funktionen sind schon $\Delta(\mathcal{N})$.

Denn ist G eine $\Sigma(\mathcal{N})$-Definition des Graphen einer totalen Funktion f, so ist wegen

$$f(\mathbf{k}) \neq l \Leftrightarrow \mathcal{N} \models \exists z(G(\mathbf{k}, z) \wedge \neg z = l)$$

auch dessen Komplement $\Sigma(\mathcal{N})$-definierbar.

Durch Lemma 4.3.4 ist die Wahl der Bezeichnung "$\Sigma(\mathcal{N})$-Definition" in gewisser Weise gerechtfertigt. Folgende Umkehrung dieses Lemmas nimmt allerdings eine Schlüsselstellung in diesem Paragraphen ein.

4.3.6 Satz

Der Graph jeder streng μ-rekursiven Funktion f besitzt eine $\Sigma(\mathcal{N})$-Definition G_f.

Beweis durch Induktion nach der Definition der streng μ-rekursiven Funktionen.

1.
   1. $U_i^n(k_1, ..., k_n) = l \ \Leftrightarrow\ \mathcal{N} \models k_i = l$. Also ist $*_i = *_{n+1}$ eine $\Sigma(\mathcal{N})$-Definition von $Graph(U_i^n)$.
   2. $C_k^n(k_1, ..., k_n) = l \ \Leftrightarrow\ \mathcal{N} \models k = l$. Also ist $k = *_{n+1}$ eine $\Sigma(\mathcal{N})$-Definition von $Graph(C_k^n)$.
   3. $*_1 + *_2 = *_3$ und $*_1 \cdot *_2 = *_3$ sind offenbar $\Sigma(\mathcal{N})$-Definitionen von $Graph(+)$ und $Graph(\cdot)$.
   4. $\chi_<(k, l) = m \ \Leftrightarrow\ \mathcal{N} \models (k < l \wedge m = 0) \vee (\neg k < l \wedge m = 1)$. Also ist $(*_1 < *_2 \wedge *_3 = 0) \vee (\neg *_1 < *_2 \wedge *_3 = 1)$ eine $\Sigma(\mathcal{N})$-Definition von $Graph(\chi_<)$.

2. Sei $f = h \circ (g_1, ..., g_r)$. Wir schreiben wieder $\mathbf{k}$ für $k_1, ..., k_n$. Nach Induktionsvoraussetzung gibt es $\Sigma(\mathcal{N})$-Definitionen G_i von $Graph(g_i)$ $(i = 1, ..., r)$ und G_h von $Graph(h)$. Dann ist

   $Graph(f)(\mathbf{k}, l) \ \Leftrightarrow\ h(g_1(\mathbf{k}), ..., g_r(\mathbf{k})) = l$

   $\Leftrightarrow$ es gibt $m_1, ..., m_r$, so dass $g_i(\mathbf{k}) = m_i$ und $h(m_1, ..., m_r) = l$

   $\Leftrightarrow \mathcal{N} \models \exists z_1 ... \exists z_r (G_1(\mathbf{k}, z_1) \wedge ... \wedge G_r(\mathbf{k}, z_r) \wedge G_h(z_1, ..., z_r, l))$.

   Diese letzte Formel ist eine $\Sigma(\mathcal{N})$-Formel $G_f(\mathbf{k}, l)$, so dass G_f eine $\Sigma(\mathcal{N})$-Definition von $Graph(f)$ ist.

3. Sei $f = \mu g$. Nach Induktionsvoraussetzung gibt es eine $\Sigma(\mathcal{N})$-Definition G_g von $Graph(g)$. Dann ist

   $Graph(f)(\mathbf{k}, l) \ \Leftrightarrow\ \mu g(\mathbf{k}) = l$

   $\Leftrightarrow\ g(\mathbf{k}, l) = 0$ und für alle $i < l$ ist $g(\mathbf{k}, i) \neq 0$

   $\Leftrightarrow \mathcal{N} \models G_g(\mathbf{k}, l, 0) \wedge \forall x < l \ \ \exists z (G_g(\mathbf{k}, x, z) \wedge \neg z = 0$.

   Diese letzte Formel ist eine $\Sigma(\mathcal{N})$-Formel $G_f(\mathbf{k}, l)$, so dass G_f eine $\Sigma(\mathcal{N})$-Definition von $Graph(f)$ ist.

Damit ist der Satz vollständig bewiesen.

4.3.7 Korollar

Die Graphen μ-rekursiver Funktionen sind $\Delta(\mathcal{N})$.

Denn ist f μ-rekursiv, so ist f nach Satz 4.2.15 streng μ-rekursiv. Dann besitzt $Graph(f)$ nach Satz 4.3.6 eine $\Sigma(\mathcal{N})$-Definition, ist also $\Sigma(\mathcal{N})$. Weil f total ist, ist $Graph(f)$ nach Lemma 4.3.5 dann schon $\Delta(\mathcal{N})$.

4.3.8 Korollar

Die rekursiven Prädikate sind genau die $\Delta(\mathcal{N})$-Prädikate:

$$\Delta_1 = \Delta(\mathcal{N})$$

Beweis. Sei P rekursiv, also χ_P streng μ-rekursiv. Nach Satz 4.3.6 besitzt $Graph(\chi_P)$ eine $\Sigma(\mathcal{N})$-Definition F. Dann ist

$$\begin{array}{lcccl} P(\mathbf{k}) & \Leftrightarrow & \chi_P(\mathbf{k}) & = & 0 \Leftrightarrow \mathcal{N} \models F(\mathbf{k}, 0) \quad \text{und} \\ \neg P(\mathbf{k}) & \Leftrightarrow & \chi_P(\mathbf{k}) & = & 1 \Leftrightarrow \mathcal{N} \models F(\mathbf{k}, 1). \end{array}$$

Also ist $F(*_1, ..., *_n, i)$ eine $\Sigma(\mathcal{N})$-Definition von P, falls $i = 0$, und von $\neg P$, falls $i = 1$. – Die andere Richtung ist das Korollar zu Lemma 4.3.4.

4.3.9 Korollar

Die r.a. Prädikate sind genau die $\Sigma(\mathcal{N})$-Prädikate:

$$\Sigma_1 = \Sigma(\mathcal{N}).$$

Beweis. Sei P r.a. Dann gibt es nach dem Aufzählungs-Theorem 2.2.4 eine rekursive Funktion f, so dass für $\mathbf{k} \in \mathbb{N}^n$

$$P(k) \Leftrightarrow \exists y\ f(\mathbf{k}, y) = 0.$$

$Graph(f)$ besitzt nach Satz 4.3.6 eine $\Sigma(\mathcal{N})$-Definition G_f. Dann ist

$$P(\mathbf{k}) \Leftrightarrow \mathcal{N} \models \exists y G_f(\mathbf{k}, y, 0),$$

so dass $\exists y G_f(*_1, ..., *_n, y, 0)$ eine $\Sigma(\mathcal{N})$-Definition von P ist. – Die Rückrichtung ist Lemma 4.3.4.

Bemerkung. Dieses Ergebnis läßt sich wesentlich verschärfen. Als $\exists(\mathcal{N})$-Formeln kann man die $\Sigma(\mathcal{N})$-Formeln bezeichnen, in denen kein beschränkter Allquantor $\forall x < t$ auftritt. Die $\exists(\mathcal{N})$-Prädikate sind dann als spezielle

$\Sigma(\mathcal{N})$-Prädikate r.a. Wie Matijasevic 1971 aber gezeigt hat, gilt auch die Umkehrung, also insgesamt:

Die r.a. Prädikate sind genau die $\exists(\mathcal{N})$-Prädikate:

$$\Sigma_1 = \exists(\mathcal{N}).$$

Der tiefliegende Beweis verwendet zahlentheoretische Hilfsmittel und knüpft nicht an unseren Satz 4.3.6 an. Für unsere Ziele verwenden wir den Satz von Matijasevic im folgenden nicht.

Mit Korollar 4.3.8 und 9 können wir die Diskussion über arithmetische Prädikate vom Anfang dieses Kapitels präzisieren und abschließen.

4.3.10 Definition

Ein Prädikat P ist **arithmetisch**, wenn es eine geschlossene Nennform F aus der Sprache $L(Z)$ gibt, so dass (1) für alle $k_1, ..., k_n \in \mathbb{N}$ gilt.

Die $\Sigma(\mathcal{N})$-Prädikate sind offenbar spezielle arithmetische Prädikate.

4.3.11 Satz

In der arithmetischen Hierarchie treten genau die sämtlichen arithmetischen Prädikate auf:

$$P \text{ arithmetisch} \quad \Leftrightarrow \quad P \in \bigcup \{\Sigma_n | n \in \mathbb{N}\}$$

Denn tritt P in der Hierarchie auf, so hat P eine Definition

$$P(\boldsymbol{y}) \longleftrightarrow Q_1 x_1 ... Q_n x_n R(\boldsymbol{x}, \boldsymbol{y})$$

mit alternierenden Quantoren $Q_1, ..., Q_n$ und rekursivem Kern R, der nach 4.3.8 arithmetisch, sogar $\Delta(\mathcal{N})$ ist. Also ist auch P arithmetisch. Ist umgekehrt P arithmetisch, so folgt durch pränexe Umformungen und Quantorenzusammenfassungen mit den Lemmata 4.1.4 bis 6, dass P in der Hierarchie auftritt. Also ist die arithmetische Hierarchie eine Klasseneinteilung genau der arithmetischen Prädikate "nach ihrer Kompliziertheit oberhalb des Rekursiven".

Wir wollen nun eine Verbindung zwischen der Gültigkeit in $\mathcal{N}$ und der Herleitbarkeit in geeigneten zahlentheoretischen Systemen herstellen. Es zeigt sich, dass wahre $\Sigma(\mathcal{N})$-Formeln schon in sehr schwachen Teilsystemen der Zahlentheorie herleitbar sind. Ein besonders zweckmäßiges und elegantes System dieser Art wurde von Raphael Robinson angegeben und wird deshalb hier als Robinson-Formalismus ROB bezeichnet.

4.3.12 Definition des Robinson-Formalismus ROB.

Die Sprache von ROB ist die Sprache $L(Z)$ der Zahlentheorie. Die nichtlogischen Axiome von ROB sind

Ax 1.	$\neg\, suc\, a = 0$	Ax 2.	$suc\, a = suc\, b \to a = b$
Ax 3.	$a + 0 = a$	Ax 4.	$a + suc\, b = suc(a + b)$
Ax 5.	$a \cdot 0 = 0$	Ax 6.	$a \cdot suc\, b = a \cdot b + a$
Ax 7.	$\neg\, a < 0$	Ax 8.	$a < suc\, b \leftrightarrow a < b \vee a = b$

$$Ax\ 9.\ a < b \vee a = b \vee b < a.$$

Wir verwenden einen **Herleitungsbegriff** $\vdash$, für den der Korrektheits- und Vollständigkeitssatz der klassischen Prädikatenlogik (mit Identität) gilt:

$$ROB \vdash A \ \Leftrightarrow\ A \text{ gilt in jedem Modell von } ROB.$$

Bemerkung. ROB ist endlich axiomatisiert und enthält kein Schema der vollständigen Induktion. Die ersten acht Axiome legen rekursiv die Werte bzw. die Wahrheit von $suc, +, \cdot, <$ fest, wobei in der linken Spalte jeweils der Nullfall und in der rechten Spalte jeweils der Nachfolgerfall behandelt wird. Allein Ax 9, das die Linearität von $<$ festlegt, fällt aus diesem Rahmen. - Die logischen Axiome und Grundschlussregeln, die den Herleitungsbegriff bestimmen, brauchen hier im einzelnen nicht fixiert zu werden. Man halte sich aber vor Augen, dass ROB eine sehr schwache Theorie ist. In $\mathcal{N}$ selbstverständliche Tatsachen wie $\neg a < a, \neg suca = a$ oder $0 + a = a$ sind (mit freier Variablen a) in ROB nicht herleitbar. Für geschlossene Formeln dieser Art, die nur Aussagen über Ziffern machen, leistet ROB aber doch erstaunlich viel.

4.3.13 Lemma

Für Ziffern k, l, m gilt stets:

1. Aus $\mathcal{N} \models\ k = l$ folgt $ROB \vdash\ k = l$.
2. Aus $\mathcal{N} \models\ \neg k = l$ folgt $ROB \vdash\ \neg k = l$.
3. Aus $\mathcal{N} \models\ k < l$ folgt $ROB \vdash\ k < l$.
4. Aus $\mathcal{N} \models\ \neg k < l$ folgt $ROB \vdash\ \neg k < l$.
5. Aus $\mathcal{N} \models\ k + l = m$ folgt $ROB \vdash\ k + l = m$.
6. Aus $\mathcal{N} \models\ k \cdot l = m$ folgt $ROB \vdash\ k \cdot l = m$.

Beweis. Wir können hier stets "$\mathcal{N} \models$" fortlassen, weil wir etwa die Ziffer k und die natürliche Zahl $k \in \mathbb{N}$ in der Bezeichnung ohnehin nicht unterscheiden.

1. Ist $k = l$, so stimmen k und l als Ziffern überein, und $\vdash k = l$ folgt durch Substitution aus $\vdash a = a$.

   Wir beweisen 2. bis 6. durch Induktion nach der Länge der Ziffer l, eine Induktion auf der Metastufe.

2. Wir betrachten zunächst den Fall $k > l$, so dass k eine Ziffer $suc\ m$ ist. $\vdash \neg suc\ m = 0$ folgt aus $Ax1$. Sei nun $k > suc\ l$. Dann ist $m > l$, und nach Induktionsvoraussetzung ist $\vdash \neg m = l$. Aus $Ax2$ folgt mit Kontraposition $\vdash \neg suc\ m = suc\ l, q.e.d.$

   Ist $k < l$, so ist $l > k$, also $\vdash \neg l = k$, wie eben gezeigt. Dann folgt $\vdash \neg k = l$ mit der Symmetrie der Gleichheit.

3. $k < 0$ tritt nicht ein. Sei $k < suc\ l$, also $k < l$ oder $k = l$. Aus $k < l$ folgt $\vdash k < l$ nach Induktionsvoraussetzung, und aus $k = l$ folgt $\vdash k = l$ nach 1. In beiden Fällen ist $\vdash k < l \vee k = l$, woraus mit $Ax8 \vdash k < suc\ l$ folgt.

4. $\vdash \neg k < 0$ folgt aus $Ax7$. Sei nun $\neg\ k < suc\ l$, also $\neg k < l$ und $\neg k = l$. Dann folgt $\vdash \neg k < l$ nach Induktionsvoraussetzung und $\vdash \neg k = l$ nach 2., also $\vdash \neg(k < l \vee k = l)$, woraus wieder mit $Ax8 \vdash \neg\ k < suc\ l$ folgt.

5. Ist $k + 0 = m$, so stimmen k und m überein, und $\vdash k + 0 = m$ folgt aus $Ax3$. Sei nun $k + suc\ l = m$. Dann ist m eine Ziffer $suc\ n$, und es ist $k + l = n$. Es folgt $\vdash k + l = n$ nach Induktionsvoraussetzung, also auch $\vdash suc(k + l) = m$. Hieraus und aus dem Einsetzungsfall $\vdash k + suc\ l = suc(k + l)$ von $Ax4$ folgt mit der Transitivität der Gleichheit $\vdash k + suc\ l = m$.

6. wird analog zu 5. mit $Ax5$ und $Ax6$ bewiesen.

4.3.14 Lemma

Sei t ein geschlossener Term. Dann gibt es genau eine Ziffer l, so dass $\mathcal{N} \models t = l$, und für dieses l ist $ROB \vdash t = l$.

Beweis durch Induktion nach dem Aufbau von t.

1. t ist 0. Dann ist $l \equiv 0$, und es ist $\vdash 0 = 0$.

2. t ist $suc\ s$. Nach Induktionsvoraussetzung gibt es genau ein l mit $s = l$, und es ist $\vdash s = l$. Dann ist $suc\ s = suc\ l$ auch nur für $suc\ l$, und es ist $\vdash suc\ s = suc\ l$.

3. t ist $r+s$. Nach Induktionsvoraussetzung gibt es genau ein k und l mit $r = k$ und $s = l$, und es ist $\vdash r = k$ und $\vdash s = l$. Für das eindeutig bestimmte $m = k + l$ ist dann $r + s = m$, und nach Lemma 4.3.13, 5. ist $\vdash k + l = m$, so dass mit Gleichheitsschlüssen $\vdash r + s = m$ folgt.

4. Für $t \equiv r \cdot s$ folgt die Behauptung analog zu 3. mit Lemma 4.3.13, 6.

4.3.15 Lemma

$ROB \vdash F(0) \to \ldots \to F(k-1) \to a < k \to F(a)$.

Beweis durch Induktion nach der Länge von k.

1. $\vdash a < 0 \to F(a)$ folgt aussagenlogisch aus $Ax7$.

2. Aus dem Gleichheitssatz der Prädikatenlogik folgt

$$\vdash F(k) \to a = k \to F(a).$$

Dies ergibt aussagenlogisch mit der Induktionsvoraussetzung

$$\vdash F(0) \to \ldots \to F(k) \to (a < k \vee a = k) \to F(a).$$

Mit $Ax8$ folgt die Behauptung.

4.3.16 $\Sigma(\mathcal{N})$- Vollständigkeit von ROB

Für geschlossene $\Sigma(\mathcal{N})$-Formeln C gilt:

$$\text{Aus } \mathcal{N} \models C \text{ folgt } ROB \vdash C.$$

Beweis durch Induktion nach dem Aufbau von C.

1. C ist eine Primformel $s = t$ oder $s < t$ oder eine negierte Primformel $\neg s = t$ oder $\neg s < t$ mit geschlossenen Termen s, t. Dann gibt es nach Lemma 4.3.14 Ziffern k und l mit $s = k$ und $t = l$ und $k = l, k < l, \neg k = l$ bzw. $\neg k < l$. Nach Lemma 4.3.14 ist $\vdash s = k$ und $\vdash t = l$, und nach Lemma 4.3.13, 1. bis 4. ist $\vdash k = l, \vdash k < l, \vdash \neg k = l$ bzw. $\vdash \neg k < l$. Mit Gleichheitsschlüssen folgt jeweils $\vdash C$.

2. C ist $A \wedge B$. Dann gelten A und B in $\mathcal{N}$, und nach Induktionsvoraussetzung ist $\vdash A$ und $\vdash B$, also auch $\vdash C$.

3. Für $C \equiv A \vee B$ schließt man analog.

4. C ist $\forall x < tF(x)$. Dann ist t ein geschlossener Term, und nach Lemma 4.3.14 gibt es eine Ziffer k, so dass $t = k$ ist und $\vdash t = k$, und für alle $i < k$ gilt $F(i)$ in $\mathcal{N}$. Nach Induktionsvoraussetzung ist für $i < k \vdash F(i)$, so dass mit Lemma 4.3.15 $\vdash a < k \to F(a)$, also auch $\vdash \forall x < kF(x)$ und wegen $\vdash t = k$ auch $\vdash C$ folgt.

5. C ist $\exists xF(x)$. Dann gibt es eine Ziffer k, für die $F(k)$ in $\mathcal{N}$ gilt. Nach Induktionsvoraussetzung ist $\vdash F(k)$, woraus $\vdash C$ folgt.

Bemerkung. Die Umkehrung von Satz 4.3.16:

$$\text{Aus } ROB \vdash C \text{ folgt } \mathcal{N} \models C$$

ergibt sich (mit dem Korrektheitssatz der Prädikatenlogik) aus der Tatsache, dass die Struktur $\mathcal{N}$ ein Modell von ROB ist.

4.3.17 Definition

Eine geschlossene Nennform F **repräsentiert** ein Prädikat P **schwach**, wenn für alle $k_1, ..., k_n \in \mathbb{N}$ gilt:

$$P(k_1, ..., k_n) \iff ROB \vdash F(k_1, ..., k_n).$$

4.3.18 Satz über schwache Repräsentierbarkeit.

Alle rekursiv aufzählbaren Prädikate sind schwach repräsentierbar durch ihre $\Sigma(\mathcal{N})$-Definition.

Beweis. Sei P r.a. Nach 4.3.9 besitzt P eine $\Sigma(\mathcal{N})$-Definition F. Mit $\mathbf{k} := k_1, ..., k_n$ ergibt sich aus Satz 4.3.16 und der letzten Bemerkung

$$P(\mathbf{k}) \iff \mathcal{N} \models F(\mathbf{k}) \iff ROB \vdash F(\mathbf{k}),$$

weil die Formeln $F(\mathbf{k})$ geschlossene $\Sigma(\mathcal{N})$-Formeln sind.

Dieser Satz hängt, wie bemerkt, daran, dass $\mathcal{N}$ ein Modell von ROB ist. Er läßt sich direkt nur auf solche Erweiterungen von ROB übertragen, von denen $\mathcal{N}$ (oder eine Expansion von $\mathcal{N}$) ein Modell ist. Das Problem liegt schon in der Definition 4.3.17: Das Nicht-Zutreffen von P auf $\mathbf{k}$ äußert sich nur in der Nicht-Herleitbarkeit von $F(\mathbf{k})$ in ROB, die sich selbstverständlich nicht auf beliebige Erweiterungen von ROB überträgt. Erhalten bliebe dagegen die Herleitbarkeit der Negation von $F(\mathbf{k})$. Dieser Gesichtspunkt führt uns direkt zur starken Repräsentierbarkeit.

4.3.19 Definition

Eine geschlossene Nennform F **repräsentiert** ein Prädikat P **stark**, wenn für alle $k_1, ..., k_n \in \mathcal{N}$ gilt:

$$\begin{array}{lcll} P(k_1,...,k_n) & \Rightarrow & ROB \vdash & F(k_1,...,k_n) \quad \text{und} \\ \neg P(k_1,...,k_n) & \Rightarrow & ROB \vdash & \neg F(k_1,...,k_n). \end{array}$$

F **repräsentiert** eine Funktion f **stark**, wenn F den Graphen von f stark repräsentiert.

Hier zählt ROB also nicht nur die $\mathbf{k} := k_1, ..., k_n$ auf, auf die ein stark repräsentierbares Prädikat P zutrifft, sondern ROB entscheidet sogar von jedem $\mathbf{k}$, ob $P(\mathbf{k})$ oder $\neg P(\mathbf{k})$. Es ist also nur von rekursiven Prädikaten und Funktionen zu erwarten, dass sie stark repräsentierbar sind. Vorher müssen wir von der $\Sigma(\mathcal{N})$-Definition G_f, die wir in Satz 4.3.6 zu streng μ-rekursiven f konstruiert haben, noch in ROB beweisen, dass sie tatsächlich Graphen sind.

4.3.20 Lemma

Ist f streng μ-rekursiv und G_f die $\Sigma(\mathcal{N})$-Definition des Graphen von f nach Satz 4.3.6, so ist in ROB die Rechtseindeutigkeit von G_f herleitbar:

$$ROB \vdash G_f(a_1,...,a_n,b) \wedge G_f(a_1,...,a_n,b') \rightarrow b = b'$$

Beweis durch Induktion nach der Definition der streng μ-rekursiven Funktionen.

1. Für $f = U_i^n, C_k^n, +, \cdot$ ist G_f durch Gleichungen gegeben; dann ist die Behauptung trivial. Für $f = C_k^n$ ist etwa $G_f :\equiv k = *_{n+1}$, und es ist $\vdash k = b \wedge k = b' \rightarrow b = b'$.

2. Für $f = \chi_<$ ist $G_f(a,b,c) :\equiv (a < b \wedge c = 0) \vee (\neg a < b \wedge c = 1)$. Dann ist aussagenlogisch

   $$\begin{array}{lllll} \vdash & a < b & \rightarrow G_f(a,b,c) & \rightarrow c = 0 & \text{und} \\ \vdash & \neg a < b & \rightarrow G_f(a,b,c) & \rightarrow c = 1, & \text{woraus} \\ \vdash & (a < b \vee \neg a < b) & \rightarrow (G_f(a,b,c) \wedge G_f(a,b,c')) & \rightarrow c = c' & \end{array}$$

   und somit die Behauptung folgt.

3. Sei $f = h \circ (g_1, ..., g_r)$. Nach Induktionsvoraussetzung gibt es $\Sigma(\mathcal{N})$-Definitionen G_i von $Graph(g_i)$ $(i = 1, ..., r)$ und G_h von $Graph(h)$, für die (mit $\mathbf{a} := a_1, ..., a_n$ und $\mathbf{c} := c_1, ..., c_r$)

   $$\begin{array}{llllllll} \vdash & G_i(\mathbf{a}, c_i) & \wedge & G_i(\mathbf{a}, c_i') & \rightarrow & c_i & = & c_i' \quad \text{und} \\ \vdash & G_h(\mathbf{c}, b) & \wedge & G_h(\mathbf{c}, b') & \rightarrow & b & = & b' \quad \text{ist.} \end{array}$$

Hieraus folgt

$$\begin{aligned}
&\vdash (G_1(\mathbf{a}, c_1) \wedge ... \wedge G_r(\mathbf{a}, c_r) \wedge G_h(\mathbf{c}, b)) \\
&\wedge(G_1(\mathbf{a}, c_1') \wedge ... \wedge G_r(\mathbf{a}, c_r') \wedge G_h(\mathbf{c}', b')) \\
&\rightarrow c_1 = c_1' \wedge ... \wedge c_r = c_r' \wedge b = b'
\end{aligned}$$

Hierin sind o. E. die freien Variablen c_i, c_j' paarweise und von $a_1, ..., a_n, b, b'$ verschieden gewählt. Dann folgt mit Quantorenschlüssen

$$\begin{aligned}
&\vdash \exists z_1 ... \exists z_r (G_1(\mathbf{a}, z_1) \wedge ... \wedge G_r(\mathbf{a}, z_r) \wedge G_h(\mathbf{z}, b)) \\
&\wedge \exists z_1 ... \exists z_r (G_1(\mathbf{a}, z_1) \wedge ... \wedge G_r(\mathbf{a}, z_r) \wedge G_h(\mathbf{z}, b')) \rightarrow b = b',
\end{aligned}$$

und das ist die Behauptung für das G_f aus Satz 4.3.6.

4. Sei $f = \mu g$. Nach Induktionsvoraussetzung gibt es eine $\Sigma(\mathcal{N})$-Definition G_g von $Graph(g)$ mit

$$\vdash G_g(\mathbf{a}, b, c) \wedge G_g(\mathbf{a}, b, c') \rightarrow c = c'.$$

Hieraus folgt für

$$G_f(\mathbf{a}, b) :\equiv G_g(\mathbf{a}, b, 0) \wedge \forall x < b \exists z (G_g(\mathbf{a}, x, z) \wedge \neg z = 0)$$

$$\vdash G_f(\mathbf{a}, b) \rightarrow G_g(\mathbf{a}, b, 0) \wedge \forall x < b \neg G_g(\mathbf{a}, x, 0),$$

also

$$\vdash G_f(\mathbf{a}, b) \wedge G_f(\mathbf{a}, b') \rightarrow \neg b' < b \wedge \neg b < b',$$

woraus mit $Ax9$ die Behauptung folgt.

Damit ist das Lemma vollständig bewiesen. Während sich die Existenz eines Funktionswertes, d. h. $\exists y G_f(t, y)$, in ROB i. a. nur für geschlossene t herleiten läßt (vgl. Sätze 4.3.16 und 18), haben wir hier seine Eindeutigkeit für freie Variablen t allein mit logischen Gesetzen und $Ax9$ hergeleitet.

4.3.21 Satz über starke Repräsentierbarkeit.

Alle rekursiven Funktionen und Prädikate sind stark repräsentierbar.

Beweis. Sei f rekursiv. Dann ist f nach Satz 4.2.15 streng μ-rekursiv, und $Graph(f)$ hat eine $\Sigma(\mathcal{N})$-Definition G_f gemäß der Konstruktion in Satz 4.3.6. Nach Satz 4.3.18 gilt dann für $\mathbf{k} := k_1, ..., k_n$

$$f(\mathbf{k}) = l \Leftrightarrow ROB \vdash G_f(\mathbf{k}, l)$$

und ebenso

$$\begin{array}{rcl} f(\mathbf{k}) \neq l & \Leftrightarrow & \text{es gibt ein } m \in \mathbb{N} : f(\mathbf{k}) = m \text{ und } m \neq l \\ & \Leftrightarrow & ROB \vdash G_f(\mathbf{k}, m) \wedge \neg m = l \quad \text{für ein } m \\ & \Rightarrow & ROB \vdash \neg G_f(\mathbf{k}, l) \quad \text{nach Lemma 4.3.20} \end{array}$$

Also wird f von seiner $\Sigma(\mathcal{N})$-Definition G_f sogar stark repräsentiert.

Sei nun P ein rekursives Prädikat. Dann ist $f = \chi_P$ rekursiv, und G_f repräsentiert f stark, also

$$\begin{array}{lcll} P(\mathbf{k}) & \Leftrightarrow & f(\mathbf{k}) = 0 & \Leftrightarrow ROB \vdash G_f(\mathbf{k}, 0) \quad \text{und} \\ \neg P(\mathbf{k}) & \Leftrightarrow & f(\mathbf{k}) \neq 0 & \Leftrightarrow ROB \vdash \neg G_f(\mathbf{k}, 0). \end{array}$$

Also wird P von $G_f(*_1, ..., *_n, 0)$ stark repräsentiert.

Nach diesem Satz ist die starke Repräsentierbarkeit eine weitere adäquate Präzisierung des intuitiven Berechenbarkeitsbegriffs. Der Robinson-Formalismus ROB erweist sich als eine Alternative zu den Registermaschinen aus Kapitel 1. Er entspricht also einem Maschinen-Konzept insgesamt. Die $\Sigma(\mathcal{N})$-Definitionen F, die in günstigen Fällen Funktionen stark repräsentieren, entsprechen den Maschinen, die auch nur in analogen günstigen Fällen Funktionen berechnen. Und die einzelnen Herleitungen in ROB von Formeln $F(\mathbf{k}, l)$ entsprechen den einzelnen Rechnungen auf einer Maschine mit Eingabe $\mathbf{k}$ und einem Ergebnis, aus dem l abzulesen ist.

Der Repräsentierbarkeitssatz, nach dem alle streng μ-rekursiven Funktionen repräsentierbar sind, entspricht also dem Satz 1.2.19, nach dem alle μ-rekursiven Funktionen (normiert) berechenbar sind. Und ebenso, wie man die Umkehrung von Satz 1.2.19 erst über eine Arithmetisierung des Maschinen-Konzepts in Kleene's Normalform-Theorem 2.1.11 erhält, erhält man die Umkehrung des Repräsentierbarkeits-Satzes erst über eine Arithmetisierung des Formalismus ROB. Dann haben wir folgende Berechenbarkeitsbegriffe für totale Funktionen als äquivalent nachgewiesen: Eine totale Funktion ist zugleich

- Register-berechenbar
- normiert berechenbar
- μ-rekursiv
- streng μ-rekursiv
- $\Delta(\mathcal{N})$-definierbar

- stark repräsentierbar.

Entsprechend sind Prädikate zugleich

- rekursiv aufzählbar
- Σ_1
- $\Sigma(\mathcal{N})$
- schwach repräsentierbar.

Die letzte Charakterisierung gestattet die Anwendung der Rekursionstheorie auf mathematische Theorien, die Erweiterungen des Robinson-Formalismus sind. Solche arithmetische Systeme werden im letzten Kapitel behandelt.