Unvollständigkeit und Unentscheidbarkeit

Skript zum Proseminar
Die Grenzen der Berechenbarkeit
von
Thomas Piecha

Dieses Skript beruht nahezu vollständig auf dem von *René Gazzari* erstellten Skript zu der von *Peter Schroeder-Heister* zuletzt im Sommersemester 2009 gehaltenen Vorlesung "Mathematische Logik II: Gödelsche Unvollständigkeitssätze". Es wurden einige Korrekturen, Modifikationen und Ergänzungen vorgenommen.

Sommersemester 2020

Universität Tübingen
Fachbereich Informatik und
Carl Friedrich von Weizsäcker Center

Skript zum Proseminar für Studenten der Informatik und des Studium Professionale, gehalten im Sommersemester 2020.

Inhaltsverzeichnis

1 Überblick

Wir behandeln die beiden Gödelschen Unvollständigkeitssätze (§§ 2-8), zu denen wir hier zunächst einen kurzen Überblick geben. In § 9 wird auf die Unentscheidbarkeit der Prädikatenlogik eingegangen.

Erster Unvollständigkeitssatz. Der *erste Unvollständigkeitssatz* (Gödel, 1931) zeigt, dass die Theorie der *Peano-Arithmetik* PA unter Voraussetzung ihrer Konsistenz unvollständig ist. Das bedeutet, dass es in der Sprache $\mathscr{L}_{\mathsf{PA}}$ der Arithmetik einen Satz G gibt, so dass weder G noch seine Negation $\neg G$ beweisbar ist. Das heißt, es gilt sowohl $\mathsf{PA} \nvdash G$ als auch $\mathsf{PA} \nvdash \neg G$.

Wahrheit. Wir werden Aussagen $A \in \mathscr{L}_{\mathsf{PA}}$ als *wahr* bezeichnen, die im Standardmodell $\mathbb{N}$ von PA, d. h. den gewohnten natürlichen Zahlen mit Addition und Multiplikation, gültig sind. Eine Aussage A ist also wahr, falls $\mathbb{N} \vDash A$. In dieser Terminologie sagt der erste Unvollständigkeitssatz aus, dass es wahre arithmetische Sätze gibt, die in PA nicht beweisbar sind.

Vollständigkeit und Unvollständigkeit. Die *Unvollständigkeit von* PA darf keinesfalls mit der *Vollständigkeit des Kalküls* verwechselt werden. Es gilt für alle $A \in \mathscr{L}_{\mathsf{PA}}$: $\mathsf{PA} \vDash A \iff \mathsf{PA} \vdash A$.
Der *Vollständigkeitssatz für die Prädikatenlogik* erster Stufe steht in keinem Widerspruch zur Unvollständigkeit einer erststufigen Theorie (hier PA). Während sich der Vollständigkeitssatz allgemein auf den Konsequenzbegriff und damit auf Gültigkeit in *allen Modellen* bezieht, bezieht sich der Unvollständigkeitssatz, wenn man ihn semantisch formuliert, auf ein *einziges Modell*, nämlich das *Standardmodell* $\mathbb{N}$.

Beweisidee. Der Beweis der Unvollständigkeit gelingt dadurch, dass man in PA über die *Beweisbarkeit in* PA sprechen kann. Dies erreicht man durch *Codierung der (informellen) Arithmetik* mithilfe *primitiver Rekursion*. Primitiv-rekursive Funktionen und Prädikate lassen sich ihrerseits in PA *repräsentieren*. Das bedeutet, dass sich Formeln in der Sprache der Arithmetik finden lassen, die sich analog zu den Funktionen und Prädikaten verhalten.
Damit kann man in PA den Satz $G \in \mathscr{L}_{\mathsf{PA}}$ konstruieren, der seine eigene Nichtbeweisbarkeit (in PA) aussagt. Für diesen Satz kann (metasprachlich) bewiesen werden, dass weder er selbst noch seine Negation in PA beweisbar ist. Es gilt also $\mathsf{PA} \nvdash G$ und $\mathsf{PA} \nvdash \neg G$. Damit ist PA eine unvollständige Theorie.
Der Beweis der Unvollständigkeit erfolgt hier zwar für PA, kann aber auf beliebige (rekursiv aufzählbare) Axiomensysteme übertragen werden, die ausdrucksstark genug sind, um über ihre eigene Beweisbarkeit zu reden.

Zweiter Unvollständigkeitssatz. Der *zweite Unvollständigkeitssatz* (Gödel, 1931) besagt dann insbesondere, dass in PA, wieder unter Voraussetzung ihrer Konsistenz, die Konsistenz von PA nicht bewiesen werden kann. Allgemeiner lässt sich sagen: Für ein konsistentes, höchstens rekursiv aufzählbares Axiomensystem kann es keinen Konsistenzbeweis geben, der nur auf die dort zur Verfügung stehenden Mittel zurückgreift.
So benötigt Gentzen (1936) in seinen Beweisen der Widerspruchsfreiheit von PA transfinite Induktion bis zur Ordinalzahl ε_0. Es ist $\varepsilon_0 := \sup\{\omega, \omega^\omega, \omega^{\omega^\omega}, \dots\}$ die kleinste Ordinalzahl α, so dass $\omega^\alpha = \alpha$. Diese *transfinite Induktion* kann nicht mehr in PA dargestellt werden.

Alternativer Umgang mit Unvollständigkeit. Es ist möglich, den Beweisbegriff (hier gegeben durch den Kalkül des natürlichen Schließens) so zu modifizieren, dass genau die in $\mathbb{N}$ wahren Aussagen bewiesen werden können. Betrachte hierzu die folgende Schlussregel:

Definition 1.1 (ω-Regel) Kann man für eine Formel $A(x) \in \mathscr{L}_{\mathsf{PA}}$ jede Einsetzung einer natürlichen Zahl (eventuell durch einen jeweils eigenen Beweis) ableiten, so kann man schon auf die Allaussage $\forall x A(x)$ schließen. *ω-Regel*

Im Kalkül des natürlichen Schließens kann diese infinitäre, auch *unendliche Induktion* genannte Schlussregel wie folgt dargestellt werden:

$$\frac{A(0) \quad A(1) \quad A(2) \quad \dots}{\forall x A(x)}\ \omega\text{-Regel}$$

Die ω-Regel besagt, dass es ausreicht, den Standardteil des Universums zu betrachten, um die Wahrheit einer Allaussage festzustellen. Ersetzt man das Induktionsschema (d. h. das formale Analogon zur vollständigen Induktion) durch die ω-Regel, so ergibt sich eine vollständige, sogar endlich axiomatisierte Theorie.
Dies wird erkauft durch einen Kalkülbegriff, der nicht mehr endlich ist. Des Weiteren ist der resultierende Kalkül in Bezug auf die Standardsemantik nicht mehr korrekt, da mehr Sätze in ihm bewiesen werden können als im unveränderten Kalkül.

Historische Einordnung. Die Gödelschen Unvollständigkeitssätze sind ein Rückschlag für das *Hilbertsche Programm*, das sich (unter anderem) zum Ziel gesetzt hatte, die Mathematik allein aus sich selbst heraus durch mathematisch-logische Widerspruchsfreiheitsbeweise zu rechtfertigen. Dennoch sollte erwähnt werden, dass erst dieses Programm die Arbeit von Kurt Gödel ermöglicht hat.

Zur Terminologie. Wir verwenden den Terminus *Peano-Arithmetik* (PA) für die erststufige Arithmetik mit Induktionsschema, weil es sich so eingebürgert hat. Historisch ist diese Terminologie nicht ganz korrekt, da Peano (1889) auf wesentliche Ideen von Dedekind

(1888) zurückgreift, die wiederum auch schon bei Frege (1879, 1884) vorhanden waren. Das für PA charakteristische Induktionsschema taucht (in nicht mengentheoretischer Fassung) weder bei Frege, Dedekind noch Peano, sondern erstmals bei Hilbert (1922) auf.

Das Skript folgt in weiten Teilen § 8 in D. van Dalen, *Logic and Structure. Fifth Edition*, Springer, 2013. Weitere Quellen und zur Vertiefung empfohlene Werke finden sich im Literaturverzeichnis. Es wird darauf verzichtet, die Verwendung der Quellen immer kenntlich zu machen.

2 Formale Arithmetik

Die Gödelschen Unvollständigkeitssätze beziehen sich auf die *formale Theorie der Arithmetik* PA, die in der Logik erster Stufe formuliert wird. Wir führen diese Theorie nun formal ein und diskutieren einige ihrer grundlegenden Eigenschaften.

2.1 Einführung der formalen Theorie

Definition 2.1 Die *formale Sprache $\mathscr{L}_{\mathsf{PA}}$ erster Stufe der Arithmetik* (mit den logischen Zeichen $\bot, \land, \to, \forall, =$) hat die folgenden nicht-logischen Zeichen im Alphabet:

formale Sprache $\mathscr{L}_{\mathsf{PA}}$ der Arithmetik

(i) eine Individuenkonstante: 0;

(ii) ein einstelliges Funktionszeichen: S (*successor/Nachfolger*);

(iii) drei zweistellige Funktionszeichen: $+$, $\cdot$ und $\uparrow$ (für die Exponentiation).

Es stehen keine Relationszeichen zur Verfügung.
Mit $\Sigma_{\mathsf{PA}} := \{\bot, \land, \to, \forall, 0, S, +, \cdot, \uparrow, =, (,), x_0, x_1, \ldots\}$ wird das Alphabet der Sprache $\mathscr{L}_{\mathsf{PA}}$ bezeichnet. Mit $\mathrm{TERM}_{\mathscr{L}_{\mathsf{PA}}}$ bezeichnen wir die Menge der Terme in $\mathscr{L}_{\mathsf{PA}}$.
Formeln sind definiert wie üblich.

Definition 2.2 (Axiome von PA) Der formalen Theorie PA der Arithmetik liegt das folgende Axiomensystem zugrunde:

Axiome von PA

Anordnung der Zahlen. Axiome für die 0 und für die Nachfolgerfunktion:

(P1) $\forall x : S(x) \neq 0$;
(P2) $\forall x \forall y : S(x) = S(y) \to x = y$.

Induktionsschema. Für jede Formel $A \in \mathscr{L}_{\mathsf{PA}}$ mit $\mathrm{FV}(A) = \{x\}$ (d. h. in A kommt exakt x als freie Variable vor), die Aussage:

(IND) $(A(0) \land \forall x(A(x) \to A(S(x)))) \to \forall x A(x)$.

Addition. Rekursive Definitionsgleichungen für die Addition:

(A1) $\forall x : x + 0 = x$;
(A2) $\forall x \forall y : x + S(y) = S(x + y)$.

Multiplikation. Rekursive Definitionsgleichungen für die Multiplikation:

(M1) $\forall x : x \cdot 0 = 0$;
(M2) $\forall x \forall y : x \cdot S(y) = (x \cdot y) + x$.

Exponentiation. Rekursive Definitionsgleichungen für die Exponentiation:

(E1) $\forall x : x \uparrow 0 = S(0)$;
(E2) $\forall x \forall y : x \uparrow S(y) = (x \uparrow y) \cdot x$.

Definition 2.3 (Theorie PA) Die *Theorie* PA ist der deduktive Abschluss (Ded) Axiomensystems für PA, d. h.

Theorie PA

$$\mathsf{PA} := \mathrm{Ded}((\mathrm{P1}),(\mathrm{P2}),(\mathrm{IND}),(\mathrm{A1}),\ldots,(\mathrm{E2})).$$

Bemerkungen. (i) Das Induktionsschema (IND) ist für jede Einsetzung einer Formel $A \in \mathscr{L}_{\mathsf{PA}}$ mit der freien Variable x ein eigenes Axiom; da es abzählbar-unendlich viele derartige Formeln in $\mathscr{L}_{\mathsf{PA}}$ gibt, ist PA abzählbar-unendlich axiomatisiert.

Um dieses Schema durch ein einzelnes Axiom auszudrücken, wird Logik zweiter Stufe benötigt; in dieser kann über Teilmengen des Universums quantifiziert werden.

(ii) Üblicherweise wird PA ohne genuine Exponentiation betrachtet. Wir nehmen die Exponentiation jedoch hinzu, um später auf die *Gödelsche β-Funktion* (zur Erzeugung von Anfangsstücken beliebiger Zahlenfolgen) bei der *Repräsentation primitiv-rekursiver Funktionen* verzichten zu können.

Gödelsche β-Funktion

Matiyasevich (1970) hat im Rahmen seiner (negativen) Lösung des 10. Hilbertschen Problems (Lösbarkeit diophantischer Gleichungen) gezeigt, dass die Exponentiation in der gewöhnlichen PA (d. h. nur mit $+$ und $\cdot$) definierbar ist. Dies ist ein weiteres Argument dafür, hier auf den technischen Aufwand zu verzichten und die Exponentiation genuin zur Arithmetik hinzuzunehmen.

Beispiel. (Sätze der formalen Arithmetik) Die bekannten Rechenregeln der Arithmetik lassen sich formal aus den Axiomen von PA ableiten. Wir erwähnen insbesondere:

Kommutativität von Addition und Multiplikation:

$$\mathsf{PA} \vdash \forall xy : x + y = y + x \quad \text{und} \quad \mathsf{PA} \vdash \forall xy : x \cdot y = y \cdot x$$

Kürzungsregel für die Addition und Multiplikation:

$$\mathsf{PA} \vdash x + z = y + z \to x = y \quad \text{und} \quad \mathsf{PA} \vdash x \cdot S(z) = y \cdot S(z) \to x = y$$

Nichtexistenz von Zwischenzahlen:

$$\mathsf{PA} \vdash \neg(x < y \wedge y < S(x))$$

Dabei ist $x < y$ eine abkürzende Schreibweise für $\exists z : x + S(z) = y$.

Beweise für derartige Aussagen erfolgen in einem Beweiskalkül. Wir verwenden den *Kalkül des natürlichen Schließens* (siehe Gentzen, 1935; Prawitz, 1965).
Falls in der Mathematik die entsprechenden Aussagen der Arithmetik mit vollständiger Induktion bewiesen werden, wird bei einer formalen Ableitung das Induktionsschema

Kalkül des natürlichen Schließens

verwendet. Beweise haben dann den folgenden Aufbau:

$$\dfrac{\dfrac{\dfrac{\dfrac{\begin{matrix}\mathscr{D}_1 \\ A(0)\end{matrix} \qquad \begin{matrix}\mathscr{D}_2 \\ \forall x(A(x) \to A(S(x)))\end{matrix}}{A(0) \land (A(x) \to A(S(x)))} \qquad A(0) \land (\forall x A(x) \to A(S(x))) \to \forall x A(x)}{\forall x A(x)}\ (\mathrm{IND})}{A(x)}}{}$$

Es ist $A \in \mathscr{L}_{\mathsf{PA}}$ eine geeignete Formel mit genau einer freien Variable x. So ist etwa im Beweis der Kommutativität der Addition $A(x) \doteq \forall y : x + y = y + x$ (wir verwenden das metasprachliche Zeichen "$\doteq$" um syntaktische Gleichheit auszudrücken). Dabei kann es durchaus vorkommen, dass schon der Induktionsanfang $\mathsf{PA} \vdash A(0)$ durch eine eigene Induktion bewiesen werden muss.

2.2 Repräsentation von Zahlen

In der Sprache $\mathscr{L}_{\mathsf{PA}}$ der Arithmetik steht lediglich für die Zahl $0 \in \mathbb{N}$ ein eigenes Namenszeichen zur Verfügung. Mithilfe der *Repräsentation der natürlichen Zahlen in der Arithmetik* kann in der Sprache $\mathscr{L}_{\mathsf{PA}}$ auch über die anderen natürlichen Zahlen gesprochen werden. Wir führen diese Repräsentation im Folgenden ein und stellen erste Zusammenhänge zwischen geschlossenen Termen t der Sprache $\mathscr{L}_{\mathsf{PA}}$ und den natürlichen Zahlen $n \in \mathbb{N}$ dar.

Notation. (Repräsentation von Zahlen) Die (echten) natürlichen Zahlen können in der Arithmetik durch Terme repräsentiert werden. Wir schreiben für jede Zahl $n \in \mathbb{N}$ den Term

$$\overline{n} :\doteq S^n(0) :\doteq \underbrace{S(\ldots S}_{n\text{-mal}}(0)).$$

Formal ist die Repräsentation natürlicher Zahlen die folgende Abbildung:

$$\overline{\cdot} : \mathbb{N} \to \mathrm{TERM}_{\mathscr{L}_{\mathsf{PA}}} : n \mapsto \overline{n} \doteq \begin{cases} 0 & \text{falls } n = 0, \\ S(\overline{m}) & \text{falls } n = m + 1. \end{cases}$$

Bemerkungen. *Interpretation in $\mathbb{N}$:* Da $\mathbb{N}$ ein Modell der Arithmetik ist (d. h. $\mathbb{N} \vDash \mathsf{PA}$), folgt sofort, dass jeder Term $\overline{n}$ in der Struktur $\mathbb{N} = \{\mathbb{N}, +, \cdot, \uparrow, 0\}$ (unter jeder Belegung v) durch die natürliche Zahl $n \in \mathbb{N}$ ausgewertet wird. Es gilt also: $[\![\overline{n}]\!]_v^{\mathbb{N}} = n \in \mathbb{N}$.

Voraussetzung der Konsistenz: In semantischen Kontexten, in denen über das Standardmodell $\mathbb{N}$ gesprochen wird, wird die Konsistenz von PA immer vorausgesetzt und muss deshalb nicht eigens zur Voraussetzung gemacht werden.

Dies wird beim Beweis des Gödelschen Unvollständigkeitssatzes anders sein, da dieser rein syntaktisch erfolgt. Die Konsistenz von PA muss dann als Voraussetzung gefordert werden.

Proposition 2.4 (Verträglichkeit der Repräsentation) *Die Repräsentation der natürlichen Zahlen verträgt sich mit den einzelnen Verknüpfungen. Es gilt also für alle $n, m \in \mathbb{N}$:*

(i) $\mathsf{PA} \vdash \overline{n} + \overline{m} = \overline{n+m}$;

(ii) $\mathsf{PA} \vdash \overline{n} \cdot \overline{m} = \overline{n \cdot m}$;

(iii) $\mathsf{PA} \vdash \overline{n} \uparrow \overline{m} = \overline{n \uparrow m}$.

Beweis. Durch vollständige Induktion über $m \in \mathbb{N}$.
Exemplarisch für (i):

$m = 0$: Mit $n + 0 = n$ und $\overline{0} \simeq 0$ triviale Anwendung von (A1).

Induktionsvoraussetzung (IV): Es gelte $\mathsf{PA} \vdash \overline{n} + \overline{m} = \overline{n+m}$.

$m + 1$: Mit (A2) gilt $\mathsf{PA} \vdash \overline{n} + S(\overline{m}) = S(\overline{n} + \overline{m})$.

Mit IV und Substitutivität gilt $\mathsf{PA} \vdash \overline{n} + S(\overline{m}) = S(\overline{n+m})$.

Also $\mathsf{PA} \vdash \overline{n} + \overline{m+1} = \overline{n+m+1}$. QED

Definition 2.5 (geschlossene/offene Terme) Ein Term t heißt *geschlossen*, falls keine Variablen in t vorkommen, andernfalls *offen*. *geschlossen/offen*

Proposition 2.6 (Auswertung geschlossener Terme) *Zu jedem geschlossenen Term t der Sprache $\mathscr{L}_{\mathsf{PA}}$ gibt es eine natürliche Zahl $n \in \mathbb{N}$, so dass $\mathsf{PA} \vdash t = \overline{n}$ gilt.*

Beweis. Durch Induktion über dem Aufbau von t:

$t :\simeq 0$: Es gilt $\mathsf{PA} \vdash 0 = \overline{0}$.

IV: $\mathsf{PA} \vdash t = \overline{n}$ und $\mathsf{PA} \vdash s = \overline{m}$ für $n, m \in \mathbb{N}$.

$S(t)$: Aus der IV folgt mit Substitutivität $\mathsf{PA} \vdash S(t) = S(\overline{n})$.

Also $\mathsf{PA} \vdash S(t) = \overline{l}$ für $l = n + 1 \in \mathbb{N}$.

$t + s$: Aus der IV folgt mit Substitutivität $\mathsf{PA} \vdash t + s = \overline{n} + \overline{m}$.

Mit Proposition 2.4 folgt $\mathsf{PA} \vdash t + s = \overline{n+m}$.

Also $\mathsf{PA} \vdash t + s = \overline{l}$ für $l = n + m \in \mathbb{N}$.

$t \cdot s$ und $t \uparrow s$: Werden analog zu $t + s$ gezeigt. QED

Korollar 2.7 (Vollständigkeit für atomare Aussagen) *Für alle geschlossenen Terme t, s der Sprache $\mathscr{L}_{\mathsf{PA}}$ gilt:* $\mathsf{PA} \vdash t = s \iff \mathbb{N} \vDash t = s$.

Beweis. "$\Longrightarrow$": Gilt unmittelbar, da $\mathbb{N}$ ein Modell von PA ist.
"$\Longleftarrow$": Angenommen, $\mathbb{N} \vDash t = s$. Dann gilt für jede Belegung v:

$$[\![t]\!]_v^{\mathbb{N}} = [\![s]\!]_v^{\mathbb{N}}. \tag{$\star$}$$

Da t und s geschlossene Terme sind, gibt es mit Proposition 2.6 Zahlen $n, m \in \mathbb{N}$ mit $\mathsf{PA} \vdash t = \overline{n}$ und $\mathsf{PA} \vdash s = \overline{m}$. Da $\mathbb{N} \vDash \mathsf{PA}$, gilt auch $\mathbb{N} \vDash t = \overline{n}$ und $\mathbb{N} \vDash s = \overline{m}$. Damit gilt für eine beliebige Belegung v:

$$[\![t]\!]_v^{\mathbb{N}} = [\![\overline{n}]\!]_v^{\mathbb{N}} = n \quad \text{und} \quad [\![s]\!]_v^{\mathbb{N}} = [\![\overline{m}]\!]_v^{\mathbb{N}} = m.$$

Mit $(\star)$ folgt, dass $n = m$ ist, also $\overline{n} \simeq \overline{m}$. Betrachte die folgende Ableitung:

$$\dfrac{\dfrac{\dfrac{\mathsf{PA}}{t = \overline{n}} \qquad \dfrac{}{\overline{n} = \overline{m}}}{t = \overline{m}} \qquad \dfrac{\dfrac{\mathsf{PA}}{s = \overline{m}}}{\overline{m} = s}}{t = s}$$

Damit gilt $\mathsf{PA} \vdash t = s$. QED

2.3 Beschränkte Quantifikation

Wir führen den Begriff der *beschränkten Quantifikation* für die lineare Ordnung $\leq$ ein. Man kann dies auch auf andere Kontexte übertragen. So wird in der Sprache der Mengenlehre das Zeichen $\leq$ durch das Relationszeichen $\in$ ersetzt und ein analoger Begriff der beschränkten Quantifikation verwendet. Mittels beschränkter Quantifikation führen wir die Σ-Δ-*Hierarchie* für $\mathscr{L}_{\mathsf{PA}}$-Formeln ein. Dies ermöglicht es, das Resultat zur Vollständigkeit atomarer Aussagen für bestimmte komplexe Aussagen zu verallgemeinern.

Notation. (Beschränkte Quantifikation) Es wird die folgende Notation für *beschränkte Quantifikation* (in der formalen Sprache $\mathscr{L}_{\mathsf{PA}}$) verwendet: *beschränkte Quantifikation*

Allquantor: $\forall x \leq y : A$ steht für die Formel $\forall x(x \leq y \to A)$.

Existenzquantor: $\exists x \leq y : A$ steht für die Formel $\exists x(x \leq y \land A)$.

Dabei ist $x \leq y$ eine abkürzende Schreibweise für die Formel $\exists z : x + z = y$ (mit geeigneter Variable z).

Satz 2.8 (Aussagenlogische Reduzierbarkeit) *Die beschränkte Quantifikation lässt sich in der Arithmetik auf die Aussagenlogik reduzieren. Das bedeutet für alle Formeln $A \in \mathscr{L}_{\mathsf{PA}}$ und alle $n \in \mathbb{N}$:*

(i) $\mathsf{PA} \vdash \forall x \leq \overline{n} : A(x) \leftrightarrow A(\overline{0}) \land \ldots \land A(\overline{n})$;

(ii) $\mathsf{PA} \vdash \exists x \leq \overline{n} : A(x) \leftrightarrow A(\overline{0}) \lor \ldots \lor A(\overline{n})$.

Beweisskizze. Der formale Beweis beider Aussagen ist recht aufwändig. Wir geben einige

Hinweise: Der Beweis (z. B. der ersten Aussage) erfolgt prinzipiell durch Induktion über der Größe der Schranke $n \in \mathbb{N}$.
Zum Beweis des Induktionsanfangs (und auch für den Induktionsschritt) werden Hilfsaussagen benötigt. Zum Beispiel:

$$\mathsf{PA} \vdash \forall x \forall y (x + y = 0 \to x = 0).$$

Dies kann mithilfe des Induktionsschemas für $A(y) :\simeq \forall x(x + y = 0 \to x = 0)$ gezeigt werden.
Die Argumentation im Induktionsanfang (der eigentlichen Induktion über den natürlichen Zahlen) kann dadurch vereinfacht werden, dass quantorenlogisch allgemeingültige Aussagen als Regeln des Kalküls verwendet werden. So kann man Ableitungen der folgenden Form verwenden:

$$\frac{\neg \forall x A}{\exists x \neg A}$$

Dies kann einerseits durch die Vollständigkeit des Kalküls gerechtfertigt werden. Andererseits kann dieser Übergang auch rein syntaktisch gerechtfertigt werden, indem die einzelnen Regeln schematisch bewiesen werden. Weiterhin kann man solche Umformungen auch lediglich in Teilformeln durchführen. Dies wird formal durch das Ersetzungstheorem gerechtfertigt. QED

Definition 2.9 (Klassifikation von Formeln) Formeln $A \in \mathscr{L}_{\mathsf{PA}}$ können nach dem Vorkommen von Quantoren *klassifiziert* werden:

(i) A ist eine Δ_0-*Formel*, falls in A höchstens beschränkte Quantoren vorkommen. Δ_0-*Formel*

(ii) A ist eine Σ_1-*Formel*, falls $A \simeq \langle \exists x_k \rangle B$ ist. Dabei ist $\langle \exists x_k \rangle$ ein Block von k (auch unbeschränkten) Existenzquantoren und B eine Δ_0-Formel. Σ_1-*Formel*

(iii) A ist eine Σ_1^*-*Formel*, falls A eine Σ_1-Formel ist, und der Block der Existenzquantoren die Länge 1 hat. Σ_1^*-*Formel*

Diese Formeln werden auch *strikte* Σ_1-*Formeln* genannt. *strikte* Σ_1-*Formel*

Proposition 2.10 (Vollständigkeit für Δ_0-Aussagen) *Für jede Δ_0-Aussage $A \in \mathscr{L}_{\mathsf{PA}}$ gilt:*

$$\mathsf{PA} \vdash A \quad \textit{oder} \quad \mathsf{PA} \vdash \neg A.$$

Beweis. Durch vollständige Induktion über dem Aufbau von Formeln.

A atomar: Für $A \simeq \bot$ gilt $\mathsf{PA} \vdash \neg\bot$. Sei also $A :\simeq t = s$ für zwei Terme t, s.

1. Fall ($\mathbb{N} \vDash t = s$): Mit Korollar 2.7 gilt: $\mathsf{PA} \vdash t = s$.

2. Fall ($\mathbb{N} \nvDash t = s$): Es gibt natürliche Zahlen $n, m \in \mathbb{N}$ mit:

$$\mathbb{N} \vDash t = \overline{n}, \quad \mathbb{N} \vDash s = \overline{m} \quad \text{und} \quad n \neq m.$$

Ohne Einschränkung ist $m < n$. Also $m + (k + 1) = n$ für ein $k \in \mathbb{N}$. Mit Fall 1 gilt:

$$\mathsf{PA} \vdash t = \overline{n}, \quad \mathsf{PA} \vdash s = \overline{m} \quad \text{und} \quad \mathsf{PA} \vdash \overline{m} + S(\overline{k}) = \overline{n}.$$

Betrachte folgende (verkürzt dargestellte) Ableitung:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[t = s]^1}{\overline{n} = \overline{m}}}{\overline{m} + S(\overline{k}) = \overline{m} + 0}}{S(\overline{k}) = 0} \qquad \dfrac{(\mathrm{P1})}{S(\overline{k}) \neq 0}}{\bot}}{t \neq s}\ (1)$$

Also $\mathsf{PA} \vdash t \neq s$.

A ist aussagenlogische Zusammensetzung: trivial.

A wird quantifiziert: Es muss nur beschränkte Quantifikation betrachtet werden. Diese Formeln sind logisch äquivalent zu endlichen aussagenlogischen Kombinationen quantorenfreier Formeln mit geschlossenen Termen. QED

Bemerkungen. (i) In den Beweis gingen die Hilfsbehauptung aus Proposition 2.4 sowie einige weitere Rechenregeln (z. B. die Kürzungsregel) ein.

(ii) Insbesondere wurde die beschränkte Quantifikation auf die Aussagenlogik zurückgeführt.

Proposition 2.11 (Vollständigkeit für Σ_1-Aussagen) *Es gilt für alle Σ_1-Aussagen $B \in \mathscr{L}_{\mathsf{PA}}$:* $\mathbb{N} \vDash B \iff \mathsf{PA} \vdash B$.

Beweis. "$\Longleftarrow$": $\mathbb{N}$ ist ein Modell von PA.
"$\Longrightarrow$": Die Behauptung ist für Δ_0-Aussagen mit Proposition 2.10 einfach zu zeigen.
Sei nun B eine Σ_1-Aussage. Ohne Einschränkung ist B eine strikte Σ_1-Formel (Übung).
Also $B :\simeq \exists x A(x)$ mit einer Δ_0-Formel A. Damit:

$$\begin{aligned} \mathbb{N} \vDash \exists x A(x) &\Longrightarrow \text{Es gibt ein } n \in \mathbb{N} \text{ mit: } \mathbb{N} \vDash A(\overline{n}) && \text{(Auswertung von Formeln)} \\ &\Longrightarrow \mathsf{PA} \vdash A(\overline{n}) \text{ für ein } n \in \mathbb{N} && (A(\overline{n}) \text{ ist eine } \Delta_0\text{-Aussage}) \\ &\Longrightarrow \mathsf{PA} \vdash \exists x A(x). \end{aligned}$$

QED

2.4 ω-Begriffe

Nun werden die allgemeinen syntaktischen Begriffe der *Konsistenz* und *Vollständigkeit* auf die Verhältnisse der Arithmetik angepasst. Dies erfolgt mithilfe der Repräsentation der natürlichen Zahlen.

Definition 2.12 (ω-Begriffe) Sei $T \subseteq \mathscr{L}_{\mathsf{PA}}$ eine Theorie in der Sprache der Arithmetik.

(i) T heißt *ω-vollständig*, falls für jede Formel $A \in \mathscr{L}_{\mathsf{PA}}$ mit $\mathrm{FV}(A) = \{x\}$ gilt: $T \vdash \exists x A(x) \implies$ Es gibt ein $n \in \mathbb{N}$ mit: $T \vdash A(\overline{n})$. *ω-vollständig*

(ii) T heißt *ω-konsistent*, falls es keine Formel $A \in \mathscr{L}_{\mathsf{PA}}$ mit $\mathrm{FV}(A) = \{x\}$ gibt, so dass Folgendes gilt: $T \vdash \exists x A(x)$, und für alle $n \in \mathbb{N}$: $T \vdash \neg A(\overline{n})$. *$\omega$-konsistent*

Bemerkung. Es ist ω die mengentheoretische Bezeichnung von $\mathbb{N}$ als Ordinalzahl (mit der natürlichen Ordnung). Damit verdeutlicht das Zeichen "ω" die Einschränkung der allgemeineren syntaktischen Begriffe auf den Standardbereich des Universums eines PA-Modells. In Nichtstandardmodellen von PA ist dies eine echte Teilmenge des Universums.

Proposition 2.13 (ω-Konsistenz) *Die ω-Konsistenz einer Theorie $T \subseteq \mathscr{L}_{\mathsf{PA}}$ impliziert schon ihre Konsistenz.*

Beweis. Ist T inkonsistent, folgt die ω-Inkonsistenz von T im Kalkül mit der Regel *ex falso quodlibet*. QED

Bemerkung. Die Umkehrung gilt im Allgemeinen nicht. Der erste Gödelsche Unvollständigkeitssatz liefert hierfür ein Gegenbeispiel. Der dort konstruierte *Gödel-Satz G* hat die Form $\neg\exists x B(x)$, wobei für jedes $n \in \mathbb{N}$ gilt: $\mathsf{PA} \vdash B(\overline{n})$. *Gödel-Satz G*
Damit ist $\mathsf{PA} \cup \{\neg G\}$ eine konsistente, aber nicht ω-konsistente Erweiterung der Arithmetik (sofern PA selbst konsistent ist).

Korollar 2.14 (zur Vollständigkeit von Σ_1-Aussagen) *Falls* PA *konsistent ist, gilt für alle Δ_0-Formeln $A \in \mathscr{L}_{\mathsf{PA}}$ (mit $\mathrm{FV}(A) = \{x\}$) ω-Konsistenz.*

Beweis. Sei A eine Δ_0-Formel, so dass für alle $n \in \mathbb{N}$ gilt:

$$\begin{aligned}
\mathsf{PA} \vdash \neg A(\overline{n}) &\implies \text{Für jedes } n \in \mathbb{N} \text{ gilt: } \mathbb{N} \vDash \neg A(\overline{n}) \\
&\implies \mathbb{N} \vDash \forall x \neg A(x) \qquad \text{(Auswertung von Formeln in Strukturen)} \\
&\implies \mathbb{N} \nvDash \neg\forall x \neg A(x) \mathrel{\hat{=}} \exists x A(x) \\
&\implies \mathsf{PA} \nvdash \exists x A(x) \qquad (\text{da } A(x) \in \Delta_0, \text{ also } \exists x A(x) \in \Sigma_1^*)
\end{aligned}$$

QED

3 Rekursionstheorie

Es soll in der formalen Arithmetik PA über die (informelle) Arithmetik gesprochen werden. Dies gelingt dadurch, dass die Begriffe der Arithmetik in geeigneter Codierung als *(primitiv-)rekursive Funktionen* und *Prädikate* dargestellt werden. Diese Funktionen und Prädikate können ihrerseits in der formalen Arithmetik repräsentiert werden. Wir stellen die dazu notwendigen Grundlagen der *primitiven* Rekursionstheorie und der *rekursiven Codierung endlicher Folgen* vor.

3.1 Primitive Rekursion

In der Rekursionstheorie werden die gewohnten natürlichen Zahlen samt ihrer Arithmetik als gegeben vorausgesetzt. Insbesondere steht die Nachfolgerfunktion zur Verfügung. Es wird hier mit x' der Nachfolger von x bezüglich der üblichen Ordnung in der Menge $\mathbb{N}$ der natürlichen Zahlen bezeichnet. Es gilt also $x' = x + 1$.

Notation. Zur leichteren Unterscheidung der verschiedenen Sprachen (formale und informelle Arithmetik) werden die folgenden Konventionen getroffen:

(i) Für natürliche Zahlen werden Variablen $x, y \in \mathbb{N}$ verwendet; sie werden zugleich in der formalen Sprache $\mathscr{L}_{\mathsf{PA}}$ der Arithmetik verwendet.

Für die Stellenzahl von Funktionen und Relationen und für die Länge von Tupeln werden die Variablen $k, l, n, m \in \mathbb{N}$ verwendet.

(ii) Prädikate (Relationen) $R \subseteq \mathbb{N}^n$ werden *kursiv* geschrieben, z. B. $\mathit{Prim}(x)$. Die entsprechenden Formeln in der Sprache $\mathscr{L}_{\mathsf{PA}}$ der Arithmetik schreiben wir normal und überstrichen, z. B. $\overline{\mathrm{Prim}}(x)$.

Funktionen werden analog behandelt.

Definition 3.1 (Grundfunktionen) Die folgenden Funktionen heißen *Grundfunktionen* und sind *primitiv-rekursiv*: *Grundfunktionen* *primitiv-rekursiv*

Nachfolger-Funktion: $N : \mathbb{N} \to \mathbb{N} : x \mapsto x'$.

Null- (oder Zero-)Funktion: $Z := 0$ (Z wird aufgefasst als 0-stellige Funktion).

Identitätsfunktion/Projektion:

Für alle $1 \leq k \leq n \in \mathbb{N}$: $I^n_k : \mathbb{N}^n \to \mathbb{N} : \vec{x} = \langle x_1, \ldots, x_n \rangle \mapsto x_k$.

(Ist die Stellenzahl n aus dem Kontext klar, wird statt I^n_k auch I_k geschrieben.)

Definition 3.2 (Komposition) Sei g eine m-stellige Funktion und seien $h_1, \ldots, h_m$ jeweils n-stellige primitiv-rekursive Funktionen ($n, m \in \mathbb{N}$). Dann ist die n-stellige Funktion f mit

$$f(\vec{x}) := g(h_1(\vec{x}), \ldots, h_m(\vec{x}))$$

die *Komposition von $h_1, \ldots, h_m$ in g*. (Die Funktion f ist dann ebenfalls primitiv-rekursiv.) *Komposition*

Definition 3.3 (Rekursion) Sei g eine n-stellige und h eine $(n+2)$-stellige *primitiv-rekursive* Funktion ($n \in \mathbb{N}$). Dann entsteht die $(n+1)$-stellige Funktion f mit

$$f(\vec{x}, 0) := g(\vec{x}) \quad \text{und} \quad f(\vec{x}, y') := h(\vec{x}, y, f(\vec{x}, y))$$

durch *Rekursion mit Anfangsfunktion g und Rekursionsfunktion h*. (Die Funktion f ist dann ebenfalls primitiv-rekursiv). *Rekursion*

Definition 3.4 (Primitiv-rekursive Funktionen) Die Menge der *primitiv-rekursiven Funktionen* ist die kleinste Menge aller Funktionen, die in endlich vielen Schritten aus den Grundfunktionen mithilfe von Komposition und Rekursion konstruiert werden können. *primitiv-rekursive Funktionen*

Satz 3.5 (Anzahl der primitiv-rekursiven Funktionen)
Die Menge aller primitiv-rekursiven Funktionen ist abzählbar.

Beweis. Übung. Ideen, welche die Abzählbarkeit einer aussagenlogischen Sprache zeigen, können verwendet werden. QED

Definition 3.6 (Primitiv-rekursive Relation) Eine n-stellige Relation $R \subseteq \mathbb{N}^n$ heißt *primitiv-rekursiv*, falls ihre *charakteristische Funktion* *primitiv-rekursive Relation*

$$\chi_R : \mathbb{N}^n \to \{0,1\} \subseteq \mathbb{N} : \vec{x} \mapsto \begin{cases} 1 & \text{falls } \vec{x} \in R, \\ 0 & \text{sonst,} \end{cases}$$

primitiv-rekursiv ist.

Es werden einige Standardbeispiele für primitiv-rekursive Funktionen und Relationen gegeben. Sie werden immer wieder für die Konstruktion komplexerer primitiv-rekursiver Funktionen und Prädikate benötigt.

Beispiele. Folgende Funktionen sind primitiv-rekursiv:

(i) Die n-stellige Konstantenfunktion: Für alle $n, k \in \mathbb{N}$: $C_k^n : \mathbb{N}^n \to \mathbb{N} : \vec{x} \mapsto k$.

(Sofern die Stellenzahl n aus dem Kontext ersichtlich ist, schreiben wir wie bei der Projektion anstelle von C_k^n auch C_k.)

(ii) Die Grundrechenarten: Addition, Multiplikation, Exponentiation und Fakultät.

(iii) Die (totale) Vorgänger-Funktion: $V(x) = \begin{cases} x-1 & \text{falls } x > 0, \\ 0 & \text{sonst.} \end{cases}$

(iv) Die (totale) Subtraktion: $(x \dot{-} y) = \begin{cases} x-y & \text{falls } x \geq y, \\ 0 & \text{sonst.} \end{cases}$

(v) Signum: $sg(x) = \begin{cases} 0 & \text{falls } x = 0, \\ 1 & \text{sonst.} \end{cases}$

(vi) Kosignum: $\widetilde{sg}(x) = \begin{cases} 1 & \text{falls } x = 0, \\ 0 & \text{sonst.} \end{cases}$

Es ist $\widetilde{sg}(x) = 1 \dot{-} sg(x)$.

(vii) Absolute Differenz: $|x - y| = \begin{cases} x - y & \text{falls } x \geq y, \\ y - x & \text{sonst.} \end{cases}$

(viii) Maximumsfunktion: $\max(x, y) = \begin{cases} x & \text{falls } x \geq y, \\ y & \text{sonst.} \end{cases}$

Damit auch für $1 < n \in \mathbb{N}$: $\max(x_0, \ldots, x_n) = \max(\max(x_0, \ldots, x_{n-1}), x_n)$.

(ix) Endliche Summen und Produkte: Für primitiv-rekursives $f : \mathbb{N}^{n+1} \to \mathbb{N}$:

$$Sum(\vec{x}, y) := \sum_{k=0}^{y} f(\vec{x}, k) \quad \text{und} \quad Prod(\vec{x}, y) := \prod_{k=0}^{y} f(\vec{x}, k).$$

Theorem 3.7 (Beschränkter μ-Operator) *Sei B eine n-stellige Funktion und R eine $(n+1)$-stellige Relation, beide primitiv-rekursiv. Dann ist die n-stellige Funktion f mit*

$$\begin{aligned} f(\vec{x}) &= \mu z(z < B(\vec{x}))R(\vec{x}, z) \\ &= \begin{cases} \min\{y < B(\vec{x}) \mid \langle \vec{x}, y \rangle \in R\} & \textit{falls das Minimum existiert}, \\ B(\vec{x}) & \textit{sonst}, \end{cases} \end{aligned}$$

primitiv-rekursiv.

Beweis. Betrachte P mit: $Prod(\vec{x}, y) = \begin{cases} 1 & \text{für jedes } z < y \text{ gilt } \langle \vec{x}, z \rangle \notin R, \\ 0 & \text{sonst.} \end{cases}$

Also

$$Prod(\vec{x}, 0) = 1 \quad \text{und} \quad Prod(\vec{x}, y') = \widetilde{sg}(\chi_R(\vec{x}, y)) \cdot G(\vec{x}, y).$$

Prod ist primitiv-rekursiv, und f mit

$$f(\vec{x}) := \sum_{z < N(B(\vec{x}))} Prod(\vec{x}, y)$$

erfüllt alle Anforderungen. QED

Bemerkung. Der beschränkte μ-Operator kann anstelle der Rekursion zur Definition der *primitiven Rekursion* verwendet werden. Gelegentlich wird zur Definition des beschränkten μ-Operators in der *sonst*-Klausel der Funktionswert 0 gefordert.

Definition 3.8 (Beschränkte Quantifikation) Der Begriff der *beschränkten Quantifikation* wird analog zur beschränkten Quantifikation in der formalen Sprache $\mathscr{L}_{\mathsf{PA}}$ in die Metasprache der informellen Arithmetik übernommen. *beschränkte Quantifikation*

Allquantor: $\forall(x \in \mathbb{N} \leq y) : A$ steht für $\forall x \in \mathbb{N}$: Gilt $x \leq y$, dann auch A.

Existenzquantor: $\exists(x \in \mathbb{N} \leq y) : A$ steht für $\exists x \in \mathbb{N}$ mit $x \leq y$ und A.

Die Darstellung von Relationen mithilfe der beschränkten Quantifikation vereinfacht die Angabe der primitiv-rekursiven charakteristischen Funktion.

Beispiele. Folgende Relationen sind primitiv-rekursiv:

(i) Falls $R, S \subseteq \mathbb{N}^n$ primitiv-rekursiv ist ($n \in \mathbb{N}$), dann auch

$$R^c, \quad R \cap S, \quad R \cup S, \quad R \backslash S.$$

(ii) Falls $R \subseteq \mathbb{N}^n$ und $S \subseteq \mathbb{N}^m$ primitiv-rekursiv sind ($n, m \in \mathbb{N}$), dann ist auch

$$R \times S = \{\langle r, s\rangle \mid r \in R, s \in S\} \subseteq N^{n+m}.$$

primitiv-rekursiv.

(iii) Standardrelationen: Die zweistelligen Relationen $\leq$, $<$, $\geq$, $>$ und $=$ sind primitiv-rekursiv.

(iv) Teilbarkeit: $x|y \;:\Longleftrightarrow\; \exists(z \in \mathbb{N} \leq y) : x \cdot z = y$.
(Die Zahl $x \in \mathbb{N}$ teilt die Zahl $y \in \mathbb{N}$.)

(v) Primzahl-Eigenschaft: $Prim(x) \;:\Longleftrightarrow\; 1 < x \wedge \forall(y \in \mathbb{N} < x) : (y|x \to y = 1)$.
(Die Zahl $x \in \mathbb{N}$ ist eine Primzahl.)

(vi) Nachbarprimzahlen:

$$\begin{aligned} &SuccPrim(x, y) \;:\Longleftrightarrow \\ &\qquad x < y \wedge Prim(x) \wedge Prim(y) \wedge \forall(z \in \mathbb{N} < y) : (x < z \to \neg Prim(z)) \end{aligned}$$

(Die Zahlen x und y sind aufeinanderfolgende Primzahlen.)

Bemerkung. Zur Beschreibung der beiden Relationen *Prim* und *SuccPrim* wurden zur leichteren Lesbarkeit logische Zeichen der formalen Sprache verwendet. Dessen ungeachtet wurden Relationen definiert, also Teilmengen der natürlichen Zahlen (bzw. von $\mathbb{N}^2$).

Es werden einige weitere, komplexere Beispiele für primitive Rekursion gegeben.

Theorem 3.9 (Fallunterscheidung) *Funktionen, die durch endliche, primitiv-rekursive Fallunterscheidungen entstehen, sind primitiv-rekursiv. Das heißt, die n-stellige Funktion*

$f : \mathbb{N}^n \to \mathbb{N}$ mit

$$f : \vec{x} \mapsto \begin{cases} f_1(\vec{x}) & \text{falls } P_1(\vec{x}), \\ \vdots & \vdots \\ f_k(\vec{x}) & \text{falls } P_k(\vec{x}), \end{cases}$$

ist primitiv-rekursiv, falls alle Funktionen $f_1, \dots, f_k$ und alle Relationen $P_1, \dots, P_k$ primitiv-rekursiv sind, und zudem für jedes Tupel $\vec{x} \in \mathbb{N}^n$ genau eine der Relationen zutrifft (Wohldefiniertheit der Fallunterscheidung).

Bemerkung. Da primitiv-rekursive Relationen unter endlicher Vereinigung und Komplement abgeschlossen sind, kann die letzte Klausel durch eine *sonst*-Klausel ersetzt werden. Des Weiteren sind Relationen $R \subseteq \mathbb{N}^n$, die nur auf endlich viele Tupel $\vec{x} \in \mathbb{N}^n$ zutreffen, primitiv-rekursiv. (Definiere die charakteristische Funktion mithilfe der endlichen Fallunterscheidung.)

Theorem 3.10 (Primzahlfolge) *Die Folge aller Primzahlen $p : \mathbb{N} \to \mathbb{N}$ ist primitiv-rekursiv. Anstelle von $p(n)$ wird im Folgenden auch p_n geschrieben.*

Beweis. $p_0 = 2$ und $p_{n+1} = \mu z(z \leq p_n! + 1)(Prim(z) \wedge p_n < z)$ QED

Bemerkung. Primzahlfolgen werden wir später noch etwas anders definieren, da wir nicht die primitiv-rekursive Gödelsche β-Funktion zur Codierung endlicher Tupel natürlicher Zahlen verwenden. An entsprechender Stelle wird erneut darauf hingewiesen.

Theorem 3.11 (Exponentenfunktion) *Die zweistellige Funktion $Exp(x, n)$, die angibt, mit welchem Exponent die n-te Primzahl in der Primfaktorzerlegung von x vorkommt, ist primitiv-rekursiv. Das heißt,*

$$Exp(x, n) = y \;:\Longleftrightarrow\; x = p_n^y \cdot m \quad \text{und} \quad p_n \nmid m.$$

Also

$$Exp(x, n) = \mu z(z \leq x)(p_n^{N(z)} \nmid x).$$

Bemerkung. Für die (später folgende) Repräsentation primitiv-rekursiver Funktionen in der formalen Arithmetik ist es wesentlich, dass die Exponentenfunktion 2-stellig ist und hier nicht eine unendliche Folge verschiedener 1-stelliger Funktionen definiert wurde. Zur Definition der Exponentenfunktion wurde die Exponentiation (die zusätzlich in die formale Arithmetik aufgenommen wurde) und die Primzahlfolge verwendet.

Theorem 3.12 (Wertverlaufsrekursion, abhängig von beschränkten Funktionen)
Seien folgende Funktionen, alle primitiv-rekursiv, gegeben:

(i) $g : \mathbb{N}^n \to \mathbb{N}$ *n-stellig;*

(ii) $r_1, \ldots, r_m : \mathbb{N}^{n+1} \to \mathbb{N}$ $(n+1)$*-stellig und beschränkt (d. h. es gilt* $r_k(\vec{x}, y) \leq y$ *für alle* $x_1, \ldots, x_n, y \in \mathbb{N}$*);*

(iii) $h : \mathbb{N}^{n+m+1} \to \mathbb{N}$ $(n+m+1)$*-stellig.*

Dann entsteht die Funktion $f : \mathbb{N}^{n+1} \to \mathbb{N}$ *mit*

$$f(\vec{x}, 0) = g(\vec{x}) \quad \textit{und} \quad f(\vec{x}, y') = h(\vec{x}, y, f(\vec{x}, r_1(\vec{x}, y)), \ldots, f(\vec{x}, r_m(\vec{x}, y))))$$

durch Wertverlaufsrekursion und ist primitiv-rekursiv.

Beweisskizze. Man definiert zunächst eine Hilfsfunktion, in der alle vorherigen Funktionswerte der Reihe nach gespeichert werden. Anschließend wird f aus der Hilfsfunktion mithilfe der Exponentenfunktion rekonstruiert. Dabei wird f an der letzten Stelle der Hilfsfunktion codiert. QED

Bemerkung. Spezialfälle:

(i) Für $m = 0$ ist f die zu h analoge Funktion, welche zunächst an der Stelle 0 einen beliebigen Funktionswert und an den folgenden Stellen der Reihe nach alle Werte von h (um eine Stelle verschoben) hat.

(ii) Für $m = 1$ und $r_1 = I_{n+1}$ ist die Wertverlaufsrekursion die gewöhnliche Rekursion.

Theorem 3.13 (Fibonacci-Zahlen) *Die Folge* $Fib(n)$ *der Fibonacci-Zahlen ist primitiv-rekursiv.*

Beweisskizze. Durch Codierung der beiden vorherigen Funktionswerte in der Hilfsfunktion (es wird *Fib* dann an erster Stelle codiert) oder durch Wertverlaufsrekursion. QED

3.2 Rekursive Funktionen und Prädikate

Rekursive Funktionen können als totale Funktionen oder als partielle Funktionen definiert werden. Letzteres ist aus Sicht der theoretischen Informatik sinnvoll, da hier z. B. nichtterminierende Turingmaschinen behandelt werden. Ersteres entspricht eher dem mathematischen Empfinden.

Definition 3.14 (Unbeschränkter μ-Operator) Sei $g(\vec{x}, y)$ eine $n+1$-stellige Funktion, so dass es für alle $\vec{x} \in \mathbb{N}^n$ ein $y \in \mathbb{N}$ gibt mit $g(\vec{x}, y) = 0$ $(\star)$. Dann entsteht die Funktion $f : \mathbb{N}^n \to \mathbb{N}$ mit *unbeschränkter μ-Operator*

$$f(\vec{x}) = \mu z(g(\vec{x}, z) = 0) = \min\{z \in \mathbb{N} \mid g(\vec{x}, y) = 0\}$$

durch den (unbeschränkten) μ-Operator aus g.

Definition 3.15 (Rekursive Funktionen) Die Menge der *rekursiven Funktionen* ist die *rekursive Funktionen* kleinste Menge von Funktionen, die alle Grundfunktionen enthält und unter Komposition und dem μ-Operator abgeschlossen ist.

Bemerkungen. *Partielle rekursive Funktionen.* Verzichtet man auf die Bedingung $(\star)$ in der Definition des μ-Operators, so können durch den μ-Operator partielle Funktionen entstehen. Man erhält so die *partiellen rekursiven Funktionen*. Dabei müssen geeignete Bedingungen angegeben werden, an welchen Stellen diese partiellen Funktionen definiert sind. Diese Bedingungen müssen dann auch für die Komposition formuliert werden. Zum Beispiel: Partielle Funktionen sind genau an jenen Stellen definiert, an denen alle Teilfunktionen definiert sind.

Ackermannfunktion. Die Menge der rekursiven Funktionen ist eine echte Erweiterung der Menge der primitiv-rekursiven Funktionen. Die Ackermannfunktion ist ein Beispiel für eine (totale) rekursive Funktion, die nicht primitiv-rekursiv ist. Es lässt sich nämlich mit ein wenig Aufwand nachweisen, dass die Ackermannfunktion schneller wächst als jede primitiv-rekursive Funktion.

Definition 3.16 (Rekursive Relationen) Eine Relation $R \subseteq \mathbb{N}^n$ heißt rekursiv, falls ihre charakteristische Funktion $\chi_R : \mathbb{N}^n \to \{0, 1\} \subset \mathbb{N}$ rekursiv ist. *rekursive Relationen*

3.3 Rekursive Aufzählbarkeit

Ein für die Gödelschen Unvollständigkeitssätze wesentlicher Begriff ist *rekursive Aufzählbarkeit*. Denn die Unvollständigkeitssätze beziehen sich ausschließlich auf Theorien, deren Axiomatisierungen rekursiv aufzählbar sind.

Definition 3.17 (Rekursiv aufzählbar) Eine Relation $R \subseteq \mathbb{N}$ heißt *rekursiv aufzählbar*, falls sie leer ist oder es eine (totale) rekursive Funktion $f : \mathbb{N} \to \mathbb{N}$ gibt mit *rekursiv aufzählbar*

$$R = \{f(x) \mid x \in \mathbb{N}\}.$$

Beispiel. Jede rekursive Relation $\emptyset \neq R \subseteq \mathbb{N}$ ist auch rekursiv aufzählbar. Es gilt nämlich: Ist R rekursiv, dann auch ihre charakteristische Funktion χ_R. Betrachte

$$f : \mathbb{N} \to \mathbb{N} : x \mapsto \begin{cases} x & \text{falls } \chi_R(x) = 1, \\ \mu z(\chi_R(z) = 1) & \text{sonst.} \end{cases}$$

Offenbar ist f rekursiv, und es gilt $R = \{f(x) \mid x \in \mathbb{N}\}$.

Satz 3.18 (Rekursive Aufzählbarkeit) *Eine Relation $R \subseteq \mathbb{N}$ ist genau dann rekursiv, wenn R und ihr Komplement R^c jeweils rekursiv aufzählbar sind.*

3.4 Codierung von endlichen Tupeln

Zur Beschreibung arithmetischer Objekte müssen mehrere Informationen zugleich behandelt werden. So werden etwa zur Beschreibung einer komplexen Formel die unmittelbare(n) Teilformel(n) und das Hauptkonnektiv benötigt. In der (formalen) Arithmetik spricht man lediglich über einzelne Zahlen. Dies deutet an, dass man einen Weg benötigt, endliche Tupel (beliebiger Länge) durch einzelne (natürliche) Zahlen (primitiv-rekursiv) zu codieren. Dies wird im Folgenden skizziert.

Definition 3.19 (Menge aller endlichen Folgen) $\mathbb{N}^{<\omega} := \bigcup\{\mathbb{N}^n \mid 0 \neq n \in \mathbb{N}\}$ bezeichnet die *Menge aller endlichen Folgen* von natürlichen Zahlen. *Menge aller endlichen Folgen*

Definition 3.20 (Codierung endlicher Folgen) Die *Codierung* $\langle\!\langle\cdot\rangle\!\rangle : \mathbb{N}^{<\omega} \to \mathbb{N}$ *endlicher Folgen* ist definiert durch die Abbildung *Codierung endlicher Folgen*

$$\vec{a} = \langle a_0, a_1, \ldots, a_n \rangle \mapsto \langle\!\langle \vec{a} \rangle\!\rangle := p_0^{a_0+1} \cdot p_1^{a_1+1} \cdot \ldots \cdot p_n^{a_n+1},$$

für Primzahlen $p_0, \ldots, p_n$.

Bemerkungen. (i) Die Codierung $\langle\!\langle\cdot\rangle\!\rangle$ ist injektiv: Es gibt keine natürliche Zahl $n \in \mathbb{N}$, die Codezahl von zwei verschiedenen Folgen $\vec{a}, \vec{b}$ ist. (Das heißt, $\vec{a} \neq \vec{b} \implies \langle\!\langle \vec{a} \rangle\!\rangle \neq \langle\!\langle \vec{b} \rangle\!\rangle$.)

(ii) Die Codierung $\langle\!\langle\cdot\rangle\!\rangle$ ist nicht surjektiv: Eine natürliche Zahl $n \in \mathbb{N}$ ist dann keine Codezahl, wenn es zwei Primzahlen p, q mit $p < q$ gibt, so dass zwar q, nicht aber p, die Zahl n teilt. (Betrachte z. B. $10 = 2^1 \cdot 3^0 \cdot 5^1$ für $p = 3$ und $q = 5$.)

(iii) Um aus einer Codezahl $n \in \mathbb{N}$ die codierte Folge zu rekonstruieren, muss man n in seine Primfaktoren zerlegen, dann feststellen, mit welcher Potenz die einzelnen Primzahlen in n vorkommen, und davon jeweils 1 abziehen. Dies ist möglich, da die Primfaktorzerlegung in den natürlichen Zahlen eindeutig und (primitiv-rekursiv) berechenbar ist.

Satz 3.21 (Rekursivität der Codierung) *Folgende Prädikate und Funktionen sind primitiv-rekursiv:*

Codezahl/Code: *Codezahl/Code*

$$Seq(m) \;:\Longleftrightarrow\; m \neq 0 \wedge \forall (p, q \leq m) : (Prim(p) \wedge Prim(q) \wedge p \leq q \wedge q|m \to p|m).$$

(Die Zahl m ist Codezahl/Code einer endlichen Folge.)

Länge einer Codezahl: $length(m) := \begin{cases} n & \textit{falls } m = \langle\!\langle \vec{a} \rangle\!\rangle \textit{ für ein } \vec{a} \in \mathbb{N}^n \\ 0 & \textit{sonst.} \end{cases}$ *Länge einer Codezahl*

(Es ist $length(x)$ *die Länge der durch m codierten Folge* $\vec{a}$*.)*

Decodierung der k-ten Stelle: $(m)_k :=\begin{cases} a_k & \textit{falls } m = \langle\!\langle \vec{a} \rangle\!\rangle \textit{ für ein } \vec{a} \in \mathbb{N}^n \textit{ und } k < n \\ 0 & \textit{sonst.} \end{cases}$ *Decodierung*

(Es ist $(m)_k$ das k-te Argument der durch m codierten Folge $\vec{a}$.)

Konkatenation zweier Tupel: *Konkatenation von Tupeln*

$$n \circ m := \begin{cases} \langle\!\langle \langle a_0, \ldots, a_n, b_0, \ldots, b_m \rangle \rangle\!\rangle & \textit{falls } n = \langle\!\langle \langle a_0, \ldots, a_n \rangle \rangle\!\rangle \\ & \textit{und } m = \langle\!\langle \langle b_0, \ldots, b_m \rangle \rangle\!\rangle, \\ 0 & \textit{sonst.} \end{cases}$$

Beweis. Verbleibt hier ohne Beweis. QED

Bemerkung. Die Decodierung ist im Wesentlichen die Exponentenfunktion. Gelegentlich wird die Exponentenfunktion wie die Decodierung notiert.

4 Codierung der Arithmetik

Nun skizzieren wir, wie die Begrifflichkeit der Arithmetik in rekursiven Begriffen codiert wird. Die Objekte der Arithmetik (Zahlen, Formeln und Beweise) werden durch (eindeutige) Zahlen, Funktionen und Relationen beschrieben, mit denen man (ohne inhaltlichen Bezug) rechnen kann.
Nach erfolgter Codierung der Arithmetik werden diese Begriffe in die formale Arithmetik PA durch Repräsentation übertragen (siehe § 5). Dies ermöglicht es uns, formale Beweise über die Arithmetik in der Arithmetik zu führen.
Die Codierung der Arithmetik erfolgt durch die *Gödelfunktion* $\ulcorner\urcorner$, die den einzelnen Objekten eine natürliche Zahl zuordnet. Der Code $\ulcorner\xi\urcorner$ eines Objekts ξ wird auch *Gödel-Zahl* von ξ genannt, die Codierung auch *Gödelisierung*.

Gödelfunktion

Gödel-Zahl

4.1 Codierung der Sprache $\mathscr{L}_{\mathsf{PA}}$

Mit den bisher zur Verfügung gestellten Mitteln kann in einem ersten Schritt die formale Sprache $\mathscr{L}_{\mathsf{PA}}$ codiert werden.

Definition 4.1 (Codierung von Zeichen) Jedem Zeichen $\alpha \in \Sigma_{\mathsf{PA}}$ des Alphabets von $\mathscr{L}_{\mathsf{PA}}$ wird eine Primzahl als sein *Code* $\ulcorner\alpha\urcorner$ nach folgender Tabelle zugeordnet:

Code (Zeichen)

α	$\perp$	$\wedge$	$\rightarrow$	$\forall$	0	S	$+$	$\cdot$	$\uparrow$	$=$	$($	$)$	x_n
$\ulcorner\alpha\urcorner$	2	3	5	7	11	13	17	19	23	29	31	37	p_{12+n}

(Mit 2 als 0-ter Primzahl p_0 ist p_{12+n} die $(12+n)$-te Primzahl, für $n \geq 0$.)

Bemerkung. Anstatt die Zeichen fortlaufend mit allen natürlichen Zahlen zu nummerieren, weist der Code jedem Zeichen die nächste Primzahl zu. Dies ermöglicht es, die Term- und Formelstruktur der Sprache $\mathscr{L}_{\mathsf{PA}}$ in die Codierung aufzunehmen.

Definition 4.2 (Codierung von Termen) Die Terme der Sprache $\mathscr{L}_{\mathsf{PA}}$ werden rekursiv über ihrem Aufbau (d. h. der Termstruktur entsprechend) *codiert*:

Code (Terme)

(i) Individuenkonstanten und Variablen: sind schon codiert.

(ii) Für n-stellige Funktionszeichen: $\ulcorner f(t_1,\ldots,t_n)\urcorner := \langle\!\langle\langle\ulcorner f\urcorner, \ulcorner(\urcorner\ulcorner t_1\urcorner,\ldots,\ulcorner t_n\urcorner,\ulcorner)\urcorner\rangle\rangle\!\rangle$.

Bemerkungen. Funktionen werden in unserer formalen Sprache prinzipiell als Präfix, mit Klammern und ohne Kommas notiert (z. B. $+(nm)$ statt $n+m$). Die gewohnten Schreibweisen dienen der Leseerleichterung, und es wird dementsprechend codiert.

Beispiele. Codierung von Termen:

(i) $\ulcorner S(0)\urcorner = \langle\!\langle\langle\ulcorner S\urcorner, \ulcorner(\urcorner\ulcorner 0\urcorner, \ulcorner)\urcorner\rangle\rangle\!\rangle = 2^{13+1}\cdot 3^{31+1}\cdot 5^{11+1}\cdot 7^{37+1}$

(ii) $\ulcorner S(S(0))\urcorner = \langle\!\langle\langle\ulcorner S\urcorner, \ulcorner(\urcorner\ulcorner S(0)\urcorner, \ulcorner)\urcorner\rangle\rangle\!\rangle = 2^{13+1}\cdot 3^{31+1}\cdot 5^{\ulcorner S(0)\urcorner+1}\cdot 7^{37+1}$

Definition 4.3 (Codierung von Formeln) Die Formeln der Sprache $\mathscr{L}_{\mathsf{PA}}$ werden rekursiv über ihrem Aufbau (d. h. der Formelstruktur entsprechend) *codiert*. Seien dabei t_1, t_2 beliebige Terme und die Formeln $A, B \in \mathscr{L}_{\mathsf{PA}}$ schon codiert.

Code (Formeln)

(i) Das Falsum $\perp$ ist schon codiert ($\ulcorner \perp \urcorner = 2$).

(ii) $\ulcorner t_1 = t_2 \urcorner := \langle\!\langle \langle \ulcorner t_1 \urcorner, \ulcorner = \urcorner, \ulcorner t_2 \urcorner \rangle \rangle\!\rangle$.

(iii) $\ulcorner (A \wedge B) \urcorner := \langle\!\langle \langle \ulcorner (\urcorner, \ulcorner A \urcorner, \ulcorner \wedge \urcorner, \ulcorner B \urcorner, \ulcorner) \urcorner \rangle \rangle\!\rangle$.

(iv) $\ulcorner (A \to B) \urcorner := \langle\!\langle \langle \ulcorner (\urcorner, \ulcorner A \urcorner, \ulcorner \to \urcorner, \ulcorner B \urcorner, \ulcorner) \urcorner \rangle \rangle\!\rangle$.

(v) $\ulcorner (\forall x_k\, A) \urcorner := \langle\!\langle \langle \ulcorner (\urcorner, \ulcorner \forall \urcorner, \ulcorner x_k \urcorner, \ulcorner A \urcorner, \ulcorner) \urcorner \rangle \rangle\!\rangle$.

Beispiel. Codierung der Formel $A :\simeq 0 = S(0) \in \mathscr{L}_{\mathsf{PA}}$:

$$\ulcorner A \urcorner = \langle\!\langle \langle \ulcorner 0 \urcorner, \ulcorner = \urcorner, \ulcorner S(S(0)) \urcorner \rangle \rangle\!\rangle = 2^{11+1} \cdot 3^{29+1} \cdot 5^{\ulcorner S(S(0)) \urcorner + 1}$$

In der Sprache $\mathscr{L}_{\mathsf{PA}}$ kommen keine Relationszeichen vor. Entsprechend müssen diese auch nicht codiert werden. Damit wurde die gesamte Sprache der Arithmetik codiert: Jedem objektsprachlichem Ausdruck entspricht nun eine natürliche Zahl.

4.2 Codierung logischer Begriffe

Im Folgenden wird skizziert, wie die üblichen Begriffe der Logik durch primitiv-rekursive Funktionen und Prädikate ausgedrückt werden können. Für eine exakte Definition der Ausdrücke siehe z. B. van Dalen (2013), S. 246ff.

Definition 4.4 (Prädikate für das Alphabet) Folgende *Prädikate* bestimmen den Typ codierter Zeichen:

Prädikate für codierte Zeichen

(i) $Const(n) \;:\Longleftrightarrow\; n = \ulcorner 0 \urcorner = 11.$

(Es ist n Codezahl einer Individuenkonstante.)

(ii) $Var(n) \;:\Longleftrightarrow\; \exists (k \in \mathbb{N} < n) : (n = p_{k+12}).$

(Es ist n Codezahl einer Variable.)

(iii) $Fn1(n) \;:\Longleftrightarrow\; n = \ulcorner S \urcorner = 13.$

(Es ist n Codezahl eines einstelligen Funktionszeichens.)

(iv) $Fn2(n) \;:\Longleftrightarrow\; n = \ulcorner + \urcorner \vee n = \ulcorner \cdot \urcorner \vee n = \ulcorner \uparrow \urcorner.$

(Es ist n Codezahl eines zweistelligen Funktionszeichens.)

(v) $Fn(n) \;:\Longleftrightarrow\; Fn1(n) \vee Fn2(n).$

(Es ist n Codezahl eines Funktionszeichens.)

Proposition 4.5 *Die eben definierten Prädikate sind primitiv-rekursiv und leisten das Gewünschte.*

Beweis. *Const* $\subseteq \mathbb{N}$: Die charakteristische Funktion

$$\chi_{Const} : \mathbb{N} \to \mathbb{N} : x \mapsto \chi_{=}(x, C_{11})$$

der Relation *Const* ist offenbar primitiv-rekursiv. Es gilt weiter $n \in Const$ genau dann, wenn $n = \ulcorner 0 \urcorner$, da 11 nach unserer Tabelle gerade die Codezahl für die Konstante 0 ist. Für die restlichen Prädikate erfolgt der Beweis analog. QED

Mit den so definierten Prädikaten lassen sich dann sukzessive die höheren Prädikate und Funktionen für die Sprache $\mathscr{L}_{\mathsf{PA}}$ durch primitiv-rekursive Funktionen und Prädikate ausdrücken.

Definition 4.6 (Höhere Begriffe) Folgende *Prädikate und Funktionen* drücken höhere syntaktische Begriffe aus (wobei gemäß Decodierung für Ausdrücke n von $\mathscr{L}_{\mathsf{PA}}$ mit $(n)_k$ die k-te Komponente von n, für $k \geq 0$, bezeichnet wird): *Begriffe für Codezahlen*

(i) $$\begin{aligned} Term(n) \;:\Longleftrightarrow\; & Const(n) \vee Var(n) \\ & \vee (Seq(n) \wedge length(n) = 4 \wedge Fn1((n)_0) \\ & \quad \wedge (n)_1 = \ulcorner (\urcorner \wedge Term((n)_2) \wedge (n)_3 = \ulcorner) \urcorner) \\ & \vee (Seq(n) \wedge length(n) = 5 \wedge Fn2((n)_0) \\ & \quad \wedge (n)_1 = \ulcorner (\urcorner \wedge Term((n)_2) \wedge Term((n)_3) \wedge (n)_4 = \ulcorner) \urcorner). \end{aligned}$$

(Die Zahl n ist Codezahl eines Terms.)

Analog kann definiert werden:

(ii) *Form*(n). Die Zahl n ist Codezahl einer Formel.

(iii) *Axiom*(n). Die Zahl n ist Codezahl eines der Axiome von PA.

(iv) *Free*(n,m). Die Zahl n ist Codezahl einer freien Variable in einem Term t mit Codezahl $\ulcorner t \urcorner = m$ oder einer Formel A mit Codezahl $\ulcorner A \urcorner = m$.

(v) Der Substitutionsoperator

$$Sub(n,m,l) = \begin{cases} \ulcorner A[t/x] \urcorner & \text{falls } n = \ulcorner A \urcorner \text{ für eine Formel } A \in \mathscr{L}_{\mathsf{PA}}, \\ & m = \ulcorner t \urcorner \text{ für einen Term } t \text{ und} \\ & l = \ulcorner x \urcorner \text{ für eine Variable } x, \\ 0 & \text{sonst.} \end{cases}$$

Proposition 4.7 *Die so definierten Prädikate und Funktionen für höhere Begriffe sind primitiv-rekursiv und leisten das Gewünschte.*

Beweis. Exemplarisch wird das Vorgehen für die einstellige Relation *Term* $\subseteq \mathbb{N}$ skizziert, um eine Idee von den Ausdrucksmöglichkeiten innerhalb der primitiven Rekursion zu bekommen (Rest als Übung).

Zunächst werden einige Hilfsfunktionen definiert, die alles Unwesentliche bei der

Definition von *Term* zusammenfassen und so eine übersichtliche Konstruktion der charakteristischen Funktion erlauben. Sei dazu $\chi := \chi_=$ die charakteristische Funktion der Gleichheit. Definiere damit:

$$\chi_0 : \mathbb{N} \to \mathbb{N} : n \mapsto \max\{\chi_{Var}(n), \chi_{Const}(n)\}$$

$$\chi_1 : \mathbb{N} \to \mathbb{N} : n \mapsto \chi_{Seq}(n) \cdot \chi(length(n), 4) \cdot \chi((n)_1, \ulcorner (\urcorner) \cdot \chi((n)_3, \ulcorner) \urcorner) \cdot \chi((n)_0, \ulcorner S \urcorner)$$

$$\chi_2 : \mathbb{N} \to \mathbb{N} : n \mapsto \chi_{Seq}(n) \cdot \chi(length(n), 5) \cdot \chi((n)_1, \ulcorner (\urcorner) \cdot \chi((n)_4, \ulcorner) \urcorner) \\ \cdot sg(\chi((n)_0, \ulcorner + \urcorner) + \chi((n)_0, \ulcorner \cdot \urcorner) + \chi((n)_0, \ulcorner \uparrow \urcorner))$$

Nach Konstruktion der einzelnen Funktionen ist sofort klar:

(i) $\chi_0(n)$ ist genau dann 1, wenn n Codezahl einer Variable oder der Konstanten 0 ist.

(ii) $\chi_1(n)$ ist genau dann 1, wenn n eine Codezahl der Länge 4 ist, wobei die Komponente 1 von n eine öffnende und die Komponente 3 von n eine schließende Klammer codiert, und falls die Komponente 0 von n zudem das Zeichen "S" codiert.

(iii) $\chi_2(n)$ ist genau dann 1, wenn n eine Codezahl der Länge 5 ist, wobei die Komponente 1 von n eine öffnende und die Komponente 4 von n eine schließende Klammer codiert, und falls die Komponente 0 von n entweder das Zeichen "$+$" oder "$\cdot$" oder "$\uparrow$" codiert.

Damit hat χ_{Term} im Wesentlichen die folgende Form:

$$\chi_{Term} : \mathbb{N} \to \mathbb{N} : n \mapsto \max\{\chi_0(n), \chi_1(n) \cdot \chi_{Term}((n)_2), \chi_2(n) \cdot \chi_{Term}((n)_2) \cdot \chi_{Term}((n)_3)\}.$$

Es ist nachzuweisen, dass χ_{Term} primitiv-rekursiv ist. Dies erfolgt mithilfe der Wertverlaufsrekursion:

(i) $g : \to \mathbb{N} :\mapsto 0$.

(ii) $r_1 : \mathbb{N} \to \mathbb{N} : n \mapsto (n)_2$, $r_2 : \mathbb{N} \to \mathbb{N} : n \mapsto (n)_2$, $r_3 : \mathbb{N} \to \mathbb{N} : n \mapsto (n)_3$.

(iii) $h : \mathbb{N}^4 \to \mathbb{N} : \langle x_0, x_1, x_2, x_3 \rangle \mapsto \max\{\chi_0(x_0), \chi_1(x_0) \cdot x_1, \chi_2(x_0) \cdot x_2 \cdot x_3\}$.

(iv) $f(0) := g$, $f(n+1) := h(n, f(r_1(n)), f(r_2(n)), f(r_3(n)))$.

(v) $\chi_{Term}(n) = f(N(n))$. QED

4.3 Codierung von Beweisen

Ableitungen werden ebenfalls rekursiv codiert, ihrem Aufbau entsprechend. Dabei lassen sich Ableitungen in Abhängigkeit der letzten verwendeten Schlussregel des Kalküls durch die folgenden Informationen beschreiben:

(i) Einführungs- oder Beseitigungsregel, zugehöriger Junktor;

(ii) vorherige Teilableitungen und Konklusion.

Diese Informationen werden in der codierenden Folge mitcodiert.

Definition 4.8 (Codierung von Ableitungen) Die *Codierung von Ableitungen* ist wie folgt aufgebaut:

Codierung von Ableitungen

(i) An 0-ter Stelle wird die Art der letzten Regel codiert:

(a) Die Einführung einer Annahme wird mit 0 codiert.

(b) Schlussregeln mit einem geordneten Paar von Zahlen:

In diesem Paar steht zuerst, ob die angewandte Regel eine Einführungs- oder Beseitigungsregel ist (codiert durch 0 oder 1). An nächster Stelle steht das zugehörige logische Zeichen (Junktor oder Quantor), codiert durch die jeweilige Gödelzahl als Zeichen.

(ii) An den folgenden Stellen werden die bisherigen Teilableitungen, aus denen die Ableitung besteht, codiert.

(iii) An letzter Stelle wird die Konklusion der Ableitung codiert.

Beispiele. Folgende Beispiele sollen die Codierung von Ableitungen verdeutlichen:

(i) Sei $\mathscr{D}$ die Einführung der Annahme A: $\mathscr{D} :\simeq A$. Dann ist $\ulcorner\mathscr{D}\urcorner = \langle 0, \ulcorner A\urcorner\rangle$.

(ii) Sei $\mathscr{D}$ der Form: $\mathscr{D} :\simeq \dfrac{\begin{matrix}\mathscr{D}_1 & \mathscr{D}_2\\ A & B\end{matrix}}{A \wedge B}$.

Dann ist $\ulcorner\mathscr{D}\urcorner := \langle\langle 0, \ulcorner\wedge\urcorner\rangle, \ulcorner\mathscr{D}_1\urcorner, \ulcorner\mathscr{D}_2\urcorner, \ulcorner A \wedge B\urcorner\rangle$.

(iii) Die verbleibenden Ableitungen werden analog codiert.

Prädikate für Ableitungen. Wie schon bei Termen und Formeln, werden die benötigten Begriffe durch primitiv-rekursive Funktionen und Prädikate ausgedrückt.

Behandlung von Annahmen. Der Kalkül des natürlichen Schließens erlaubt die Löschung offener Annahmen bei Anwendung bestimmter Regeln, wobei die Löschung jedoch nicht vorgenommen werden muss, sondern kann. Dies ist schwer in Form rekursiver Prädikate zu berechnen.
Für das Gödelsche Programm ist dies aber unproblematisch. Es soll dort nämlich über Beweisbarkeit gesprochen werden. Dazu genügt es, einen Ableitungsbaum zu finden, der die Beweisbarkeitsaussage belegt. Damit kann hier ohne Einschränkung angenommen werden, dass alle Annahmen, die gelöscht werden dürfen, tatsächlich gelöscht werden. Damit muss zur Bestimmung der Annahmenmenge $Hyp(\mathscr{D})$ einer Ableitung $\mathscr{D}$ lediglich darüber Buch geführt werden, welche Annahmen eingeführt wurden und welche Annahmen gelöscht werden dürfen. Man codiert die eingeführten Annahmen in einer endlichen Folge von Zahlen, aus der bei Löschung einer Annahme die entsprechende Zahl entfernt wird.

Definition 4.9 (Beweisprädikat) Schließlich kann man das für die Unvollständigkeitssätze zentrale, primitiv-rekursive *Beweisprädikat* $Bew(n,m)$ definieren, das wahr ist, falls n Codezahl einer Ableitung $\mathscr{D}$ ist, m Codezahl einer Formel A ist, und die Ableitung $\mathscr{D}$ die Formel A (unter Verwendung der Axiome von PA) beweist. Es gilt also:

Beweisprädikat

$$Bew(n,m) \;:\Longleftrightarrow\; n = \ulcorner \begin{array}{c} A_1,\dots,A_k \\ \mathscr{D} \\ A \end{array} \urcorner,\; m = \ulcorner A \urcorner \text{ und } Axiom(\ulcorner A_l \urcorner) \text{ für } 1 \leq l \leq k.$$

Bemerkung. Das hier definierte Beweisprädikat $Bew(n,m)$ darf nicht mit dem in Gödel (1931) definierten Prädikat $\mathrm{Bew}(x)$ verwechselt werden. Letzteres drückt aus, dass x eine beweisbare Formel ist; dieses Prädikat ist nicht primitiv-rekursiv.

Proposition 4.10 *Für alle $n,m \in \mathbb{N}$ gilt: $Bew(n,m) \iff n = \ulcorner \mathscr{D} \urcorner$ für eine Ableitung $\mathscr{D}$, $m = \ulcorner A \urcorner$ für eine Formel A, und die Ableitung $\mathscr{D}$ ist ein Beweis für die Formel A in* PA. *Also* $\mathsf{PA} \vdash A$.

Beweis. Der technisch aufwändige Beweis wird hier weggelassen. QED

5 Repräsentation von Funktionen und Prädikaten

Wir behandeln die Repräsentation von (primitiv-rekursiven) Funktionen und Prädikaten in der Arithmetik. Zunächst wird geklärt, was in diesem Zusammenhang Repräsentation bedeutet. Anschließend wird gezeigt, dass die primitiv-rekursiven Funktionen und Prädikate repräsentiert werden können. Da die informelle Arithmetik schon durch primitiv-rekursive Begriffe codiert wurde, werden wir damit gezeigt haben, dass die informelle Artihmetik in der formalen Arithmetik repräsentiert werden kann.

5.1 Repräsentierbarkeit

Wir führen die Repräsentierbarkeit von Funktionen und Relationen in der Theorie PA ein. Die folgenden Definitionen lassen sich analog auf beliebige Theorien $T \subseteq \mathscr{L}_{\mathsf{PA}}$ der Sprache der Arithmetik übertragen.
Für Funktionen, die in der formalen Arithmetik PA repräsentiert werden sollen, ist eine ziffernweise Repräsentation durch einem Term ausreichend.

Definition 5.1 (Repräsentation von Funktionen durch Terme) Ein Term t_f mit $\mathrm{FV}(t) \subseteq \{x_1, \ldots, x_k\}$ *repräsentiert* eine k-stellige Funktion f in der Theorie PA, falls für alle k-Tupel $\vec{n} \in \mathbb{N}^k$ gilt:

Repräsentation durch Terme

$$\mathsf{PA} \vdash t_f(\overline{n_1}, \ldots, \overline{n_k}) = \overline{f(n_1, \ldots, n_k)}.$$

Bemerkungen. (i) Dabei ist: $t_f(\overline{n_1}, \ldots, \overline{n_k}) \simeq t_f[\overline{n_1}/x_1, \ldots, \overline{n_k}/x_k]$.

Es wird dabei *nicht* vorausgesetzt, dass tatsächlich alle Variablen x_i im Term t_f vorkommen.

(ii) Bei vielen Funktionen, wie etwa der Signum-Funktion, ist eine derartige Repräsentation nicht möglich. Stattdessen kann eine Funktion auch durch eine Formel A_f repräsentiert werden.

Definition 5.2 (Repräsentation von Funktionen durch Formeln)
Eine Formel A_f mit $\mathrm{FV}(A) \subseteq \{x_0, \ldots, x_k\}$ *repräsentiert* eine k-stellige Funktion f in der Theorie PA, falls für alle k-Tupel $\vec{n} \in \mathbb{N}^k$ gilt:

Repräsentation durch Formeln

(i) Existenz: $\mathsf{PA} \vdash A_f(\overline{f(n_1, \ldots, n_k)}, \overline{n_1}, \ldots, \overline{n_k})$;

(ii) Rechtseindeutigkeit: $\mathsf{PA} \vdash A_f(y, \overline{n_1}, \ldots, \overline{n_k}) \to y = \overline{f(n_1, \ldots, n_k)}$.

Bemerkung. Beide Bedingungen lassen sich auch in einer einzigen Formel zusammenfassen:

$$\mathsf{PA} \vdash A_f(y, \overline{n_1}, \ldots, \overline{n_k}) \leftrightarrow y = \overline{f(n_1, \ldots, n_k)}$$

Proposition 5.3 (Repräsentation durch Formeln) *Falls eine Funktion durch einen Term repräsentiert werden kann, dann kann sie auch durch eine Formel repräsentiert werden.*

Beweis. Sei eine Funktion f durch einen Term t_f repräsentiert. Dann repräsentiert $A_f :\simeq (t_f = x_0)$ die Funktion f als Formel. QED

Definition 5.4 (Repräsentation von Prädikaten) Eine Formel A_P *repräsentiert* ein k-stelliges Prädikat P in der Theorie PA, falls für alle k-Tupel $\vec{n} \in \mathbb{N}^k$ gilt:

Repräsentation von Prädikaten

(i) $\langle n_1, \dots, n_k \rangle \in P \implies \mathsf{PA} \vdash A_P(\overline{n_1}, \dots, \overline{n_k})$;

(ii) $\langle n_1, \dots, n_k \rangle \notin P \implies \mathsf{PA} \vdash \neg A_P(\overline{n_1}, \dots, \overline{n_k})$.

Bemerkung. In der Definition ist wesentlich, dass im Fall (ii) ($\langle \vec{n} \rangle \notin P$) die Beweisbarkeit der Negation verlangt wird und nicht die (wesentlich) schwächere Nichtableitbarkeit $\mathsf{PA} \nvdash A_P(\overline{\vec{n}})$.

Proposition 5.5 (Repräsentierbarkeit von Prädikaten) *Ein Prädikat P ist genau dann repräsentierbar, wenn seine charakteristische Funktion χ_P repräsentierbar ist.*

Beweis. Hier ohne Beweis. Siehe van Dalen, 2013, Lemma 8.5.4. QED

5.2 Repräsentation primitiver Rekursion

Im Folgenden soll gezeigt werden, dass alle primitiv-rekursiven Funktionen und Prädikate in PA repräsentierbar sind. Da die charakteristische Funktion eines primitiv-rekursiven Prädikats nach Definition primitiv-rekursiv ist, genügt es mit Proposition 5.5, die Repräsentierbarkeit lediglich für primitiv-rekursive Funktionen zu zeigen. Dies geschieht mithilfe einiger Hilfssätze.

Hilfssatz 5.6 (Anfangsfunktionen) *Alle Anfangsfunktionen sind in* PA *repräsentierbar.*

Beweis. Für jedes $k \in \mathbb{N}$ gilt:

(i) N (*Nachfolger-Funktion*): $t_N(x_1) :\simeq S(x_1)$;

(ii) C_k (*k-stellige Null-Funktion*): $t_{C_k}(\vec{x}) :\simeq 0$;

(iii) U_k^i (*k-stellige Projektion der i-ten Stelle*): $t_{U_k^i}(\vec{x}) :\simeq x_i$.
Es gilt: $t_{U_k^i}(\overline{n_1}, \dots, \overline{n_k}) \simeq \overline{n_i}$.
Daraus folgt: $\mathsf{PA} \vdash x_i(\overline{n_1}, \dots, \overline{n_k}) = \overline{n_i}$.

Damit sind alle Anfangsfunktionen durch Terme repräsentierbar. QED

Hilfssatz 5.7 (Komposition) *Die Komposition von Funktionen erhält die Repräsentierbarkeit in* PA.

Beweis. Sei g eine m-stellige Funktion, die durch die Formel A_g repräsentiert wird, und seien für $1 \leq i \leq m$ die k-stelligen Funktionen h_i gegeben, welche jeweils durch eine Formel B_i repräsentiert werden. Das bedeutet:

(i) $g(p_1, \ldots, p_m) = q \implies \begin{cases} \mathsf{PA} \vdash A_g(\overline{q}, \overline{p_1}, \ldots, \overline{p_m}), \\ \mathsf{PA} \vdash A_g(y, \overline{p_1}, \ldots, \overline{p_m}) \to y = \overline{q}. \end{cases}$

(ii) Für jedes $1 \leq i \leq m$:

$$h_i(n_1, \ldots, n_k) = l_i \implies \begin{cases} \mathsf{PA} \vdash B_i(\overline{l_i}, \overline{n_1}, \ldots, \overline{n_k}), \\ \mathsf{PA} \vdash B_i(y, \overline{n_1}, \ldots, \overline{n_k}) \to y = \overline{l_i}. \end{cases}$$

Sei nun $f(\vec{n}) := g(h_1(\vec{n}), \ldots, h_m(\vec{n})) = r$.
Betrachte folgende Formel (mit neuen Variablen $y_1, \ldots, y_k$):

$$\sigma(x_0, x_1, \ldots, x_k) :\simeq \exists y_1 \ldots, y_m(B_1(y_1, x_1, \ldots, x_k)) \wedge \ldots \wedge B_m(y_m, x_1, \ldots, x_k) \wedge A_g(x_0, \vec{y})).$$

Damit ist:

$$\sigma(\overline{r}, \overline{n_1}, \ldots, \overline{n_k}) \simeq \exists y_1 \ldots, y_m(B_1(y_1, \overline{n_1}, \ldots, \overline{n_k})) \wedge \ldots \wedge B_m(y_m, \overline{n_1}, \ldots, \overline{n_k}) \wedge A_g(\overline{r}, \vec{y})).$$

Man kann zeigen: $\begin{cases} \mathsf{PA} \vdash \sigma(\overline{r}, \overline{n_1}, \ldots, \overline{n_k}), \\ \mathsf{PA} \vdash \sigma(y, \overline{n_1}, \ldots, \overline{n_k}) \to y = \overline{r}. \end{cases}$

Damit ist die Komposition repräsentierbar. QED

Bemerkung. Für die Repräsentierbarkeit der primitiven Rekursion wird die Repräsentierbarkeit der *Decodierung* in der Sprache $\mathscr{L}_{\mathsf{PA}}$ benötigt. Diese benötigt ihrerseits unverzichtbar die Repräsentierbarkeit der Primzahlfolge. Dafür kann nicht auf die bekannte Definition der Primzahlfolge zurückgegriffen werden, da diese gerade primitive Rekursion verwendet, die hier aber noch nicht repräsentiert ist. *Decodierung*

Deshalb werden zuerst einige einfache Prädikate repräsentiert. Mit deren Hilfe können dann bestimmte *geeignete Zahlen* definiert und repräsentiert werden. Mit diesen geeigneten Zahlen kann die Primzahlfolge auch ohne Rekursion definiert und repräsentiert werden. *geeignete Zahlen*

Zur Definition der *geeigneten Zahlen* wird die Hinzunahme der Exponentiation zur Sprache $\mathscr{L}_{\mathsf{PA}}$ wesentlich benötigt; ohne Exponentiation sind diese nicht definierbar.

Hilfssatz 5.8 (Einfache Prädikate) *Folgende Prädikate sind in $\mathscr{L}_{\mathsf{PA}}$ repräsentierbar:*

$$n < m, \quad n \leq m, \quad n|m, \quad Prim(n), \quad SuccPrim(n, m), \quad Seq(n).$$

Beweis. Betrachte folgende Formeln:

(i) $x < y :\simeq A_<(x, y) :\simeq \exists z : x + S(z) = y;$

(ii) $x \leq y :\simeq A_\leq(x, y) :\simeq x = y \vee x < y;$

(iii) $x|y :\simeq A_|(x, y) :\simeq \exists z : x \cdot S(z) = y;$

(iv) $\overline{\text{Prim}}(x) :\simeq A_{\text{Prim}}(x) :\simeq \forall z(z|y \rightarrow z = S(0) \vee z = x);$

(v) $\overline{\text{SuccPrim}}(x, y) :\simeq A_{\text{SuccPrim}}(x, y)$

$$:\simeq (\overline{\text{Prim}}(x) \wedge \overline{\text{Prim}}(y) \wedge \forall z(x < z \wedge z < y \rightarrow \neg\overline{\text{Prim}}(z)));$$

(vi) $A_{Seq}(x)$ analog; vgl. Definition von *Seq*. QED

Bemerkungen. (i) Ganz links werden abkürzende Schreibweisen definiert, die das Lesen der Formeln erleichtern sollen. Die mittlere Formel soll an den repräsentierenden Charakter der Formeln erinnern.

(ii) Die unbeschränkten Quantoren in den einzelnen repräsentierenden Formeln können (außer bei $A_<$) durch beschränkte Quantoren ersetzt werden. Dies wird benötigt, wenn man feststellt, dass die Repräsentation der primitiv-rekursiven Funktionen und Prädikate strikt Σ_1 ist.

In der Rekursionstheorie garantiert beschränkte Quantifikation, dass aus primitiv-rekursiven Funktionen erneut primitiv-rekursive Funktionen entstehen. Eine analoge Argumentation für Prädikate ist hier nicht nötig, da die repräsentierenden Formeln keiner Primitivitäts-Bedingung gehorchen müssen.

Hilfssatz 5.9 (Decodierung) *Die Decodierung $(x)_n$ ist repräsentierbar.*

Beweisskizze. Die Behauptung wird in mehreren Schritten skizziert:

Geeignete Zahlen: Eine Zahl $m \in \mathbb{N}$ heißt *geeignet*, falls es ein $n \in \mathbb{N}$ gibt mit: *geeignete Zahl*

$$m = \prod_{k=0}^{n} p_k^k = 2^0 \cdot 3^1 \cdot \ldots \cdot p_n^n$$

für Primzahlen p_k $(0 \leq k \leq n)$.

Formal kann dies wie folgt beschrieben werden:

$$\begin{aligned} Geeig(m) \; :\Longleftrightarrow \; & (2\nmid m) \wedge \forall x \forall y \forall z : (SuccPrim(x, y) \wedge x^z | m \wedge x^{S(z)} \nmid m \rightarrow \\ & (y^{S(z)} | m \wedge y^{S(S(z))} \nmid m) \vee \forall w \geq y : (Prim(w) \rightarrow w \nmid m))). \end{aligned}$$

Das Prädikat $Geeig(m)$ kann (offensichtlich) mit den (um die Exponentiation erweiterten) Mitteln von PA ausgedrückt werden. Dabei wird zwar die Primzahl-Eigenschaft einer Zahl verwendet, nicht aber die Primzahlfolge selbst.

Die repräsentierende Formel wird mit $A_{\text{geeig}}(x)$ bezeichnet.

Primzahlfolge: Die Primzahlfolge kann in PA repräsentiert werden. Betrachte dazu die folgende Formel:

$$\begin{aligned} A_{\text{prim}}(y, x) :\simeq \; & (x = \overline{0} \wedge y = \overline{2}) \vee \\ & \exists z : (x \neq 0 \wedge \overline{\text{Geeig}}(z) \wedge \overline{\text{Prim}}(y) \wedge y^x | z \wedge y^{S(x)} \nmid z). \end{aligned}$$

Decodierung: Damit stehen alle Mittel zur Verfügung, um die *Decodierung* in PA zu repräsentieren. Betrachte dazu die folgende Formel: *Decodierung*

$$\sigma(x,y) :\simeq \overline{\mathrm{Seq}(x)} \wedge \exists z(\overline{\mathrm{length}}(z,x) \wedge y < z).$$

Die Formel σ prüft, ob decodiert werden muss, oder ob der Funktionswert trivialerweise 0 ist. (Vergleiche hierzu die Fallunterscheidung bei der Definition der Decodierung. Es ist $\overline{\mathrm{length}}(z,x)$ die Länge z von x.) Damit kann eine Formel angegeben werden, welche die Decodierung repräsentiert:

$$\begin{aligned}B(z,x,y) :\simeq {} & (\neg\sigma(x,y) \wedge z = 0) \vee \\ & (\sigma(x,y) \wedge \exists v(A_{\text{prim}}(v,y) \wedge v^{S(z)} | x \wedge v^{S(S(z))} \nmid x)).\end{aligned}$$

Der Nachweis, dass die einzelnen Formeln tatsächlich die intendierten Funktionen und Relationen repräsentieren, verbleibt als Übungsaufgabe. Ebenso eine explizite Angabe der Repräsentanten von *length* und *Seq*. QED

Hilfssatz 5.10 (Rekursion) *Primitive Rekursion erhält Repräsentierbarkeit.*

Beweis. Sei g eine k-stellige Funktion, die durch A_g repräsentiert wird, und h eine $(k+2)$-stellige Funktion, die durch A_h repräsentiert wird.
Das bedeutet:

(i) $g(\vec{n}) = m \implies \begin{cases} \mathsf{PA} \vdash A_g(\overline{m}, \overline{n_1}, \ldots, \overline{n_k}), \\ \mathsf{PA} \vdash A_g(y, \overline{n_1}, \ldots, \overline{n_k}) \to y = \overline{m}. \end{cases}$

(ii) $h(\vec{n}, p, q) = m \implies \begin{cases} \mathsf{PA} \vdash A_h(\overline{m}, \overline{n_1}, \ldots, \overline{n_k}, \overline{p}, \overline{q}), \\ \mathsf{PA} \vdash A_h(y, \overline{m}, \overline{n_1}, \ldots, \overline{n_k}, \overline{p}, \overline{q}) \to y = \overline{m}. \end{cases}$

Sei nun f eine $k+1$-stellige Funktion, gegeben durch:

$$\begin{cases} f(\vec{n}, 0) = g(\vec{n}), \\ f(\vec{n}, N(p)) = h(\vec{n}, p, f(\vec{n}, p)). \end{cases}$$

Die Repräsentation der primitiven Rekursion gelingt dadurch, dass man die Existenz einer Codezahl postuliert, in der zu jedem $n \in \mathbb{N}$ alle vorhergehenden Funktionswerte der Funktion codiert sind. Dieses Vorgehen wird zunächst detailliert vorgeführt.
Es gelte also $f(\vec{n}, p)$ für ein $p \in \mathbb{N}$. Um dies zu berechnen, werden alle Funktionswerte $f(\vec{n}, p)$ der Reihe nach bis einschließlich p berechnet:

$$\begin{aligned} a_0 &:= f(\vec{n}, 0) = g(\vec{n}) \\ a_1 &:= f(\vec{n}, 1) = h(\vec{n}, 0, f(\vec{n}, 0)) = h(\vec{n}, 0, a_0) \\ a_2 &:= f(\vec{n}, 2) = h(\vec{n}, 1, f(\vec{n}, 1)) = h(\vec{n}, 1, a_1) \end{aligned}$$

$$
\begin{aligned}
&\vdots \\
&= h(\vec{n}, l, f(\vec{n}, l)) = h(\vec{n}, l, a_l) \\
&\vdots \\
&= h(\vec{n}, p-1, f(\vec{n}, p-1)) = h(\vec{n}, p-1, a_{p-1}).
\end{aligned}
$$

Die a_k $(0 \leq k \leq p)$ bilden eine endliche Folge $\vec{a} := \langle a_0, \ldots, a_p \rangle$.
Diese Folge wird durch die Zahl $w := \langle\!\langle \vec{a} \rangle\!\rangle \in \mathbb{N}$ codiert.
Damit gilt mit den einzelnen Definitionen:

$$
\mathit{Seq}(w), \quad \mathit{length}(w) = p+1 \quad \text{und} \quad f(\vec{n}, p) = (w)_p.
$$

Dieses Vorgehen zur Berechnung der Funktion f kann formal repräsentiert werden. Betrachte dazu die folgende Formel $\sigma \in \mathscr{L}_{\mathsf{PA}}$:

$$
\begin{aligned}
\sigma(y, , x_1, \ldots, x_k, x) :\simeq \exists z : (\overline{\mathrm{Seq}}(z) \wedge \overline{\mathrm{length}}(S(x), z) \wedge A_g((z)_0, \vec{x}) \wedge \\
\forall i < x : A_h((z)_{S(i)}, \vec{x}, i, (z)_i) \wedge y = (z)_x).
\end{aligned}
$$

Nun beweisen wir, dass σ tatsächlich die Funktion f repräsentiert. Dazu ist zu zeigen:

$$
f(\vec{n}, p) = m \implies \begin{cases} \mathsf{PA} \vdash \sigma(\overline{m}, \overline{n_1}, \ldots, \overline{n_k}, \overline{p}), & \text{(a)} \\ \mathsf{PA} \vdash \sigma(y, \overline{n_1}, \ldots, \overline{n_k}, \overline{p}) \rightarrow y = \overline{m}. & \text{(b)} \end{cases}
$$

Wir beginnen mit Existenz (a): Beweis per Induktion über $p \in \mathbb{N}$.
Zuvor lässt sich allgemein für alle $p \in \mathbb{N}$ und für alle wie oben konstruierte $w \in \mathbb{N}$ festhalten:

$$
\mathsf{PA} \vdash \overline{\mathrm{Seq}}(\overline{w}) \quad \text{und} \quad \mathsf{PA} \vdash \overline{\mathrm{length}}(\overline{p+1}, \overline{w}).
$$

Auch gilt für jedes $0 \leq i \leq p$: $\mathsf{PA} \vdash (\overline{w})_{\overline{i}} = \overline{a_i}$.
Insbesondere gilt also: $\mathsf{PA} \vdash A_g((\overline{w})_0, \overline{\vec{n}})$.
Daraus folgt zunächst der Induktionsanfang: $\mathsf{PA} \vdash \sigma(\overline{f(\vec{n}, 0)}, \overline{\vec{n}}, \overline{0})$.
Aus der Induktionsvoraussetzung folgt weiterhin:

$$
\begin{aligned}
&\mathsf{PA} \vdash A_g(\overline{a_0}, \overline{n_1}, \ldots, \overline{n_k}) \\
&\mathsf{PA} \vdash A_h(\overline{a_1}, \overline{n_1}, \ldots, \overline{n_k}, \overline{0}, \overline{a_0}) \\
&\mathsf{PA} \vdash A_h(\overline{a_2}, \overline{n_1}, \ldots, \overline{n_k}, \overline{1}, \overline{a_1}) \\
&\qquad \vdots \\
&\mathsf{PA} \vdash A_h(\overline{a_p}, \overline{n_1}, \ldots, \overline{n_k}, \overline{p-1}, \overline{a_{p-1}}).
\end{aligned}
$$

Für den Induktionsschritt muss weiter festgehalten werden:

$$
f(\vec{n}, p+1) = h(\vec{n}, p, f(\vec{n}, p)) = h(\vec{n}, p, a_n), \text{ also } \mathsf{PA} \vdash A_h(\overline{a_{n+1}}, \overline{\vec{n}}, \overline{p}, \overline{a_n}).
$$

Damit gilt aber für jedes $i \in \mathbb{N}$ mit $0 \leq i < p$: $\mathsf{PA} \vdash A_h((\overline{w})_{S(i)}, \overline{\vec{n}}, \overline{i}, (\overline{w})_i)$.
Insbesondere gilt ebenfalls: $\mathsf{PA} \vdash \overline{f(\vec{n}, p+1)} = (\overline{w})_{\overline{p+1}}$.
Daraus folgt aber insgesamt: $\mathsf{PA} \vdash \sigma(\overline{f(\vec{n}, p+1)}, \overline{\vec{n}}, \overline{p+1})$.
Somit wurde die Korrektheit (a) gezeigt.

Zeige noch die Rechtseindeutigkeit (b): Erneut durch Induktion über p.
Sei dazu $m = f(\vec{n}, p)$ gegeben, und sei $v :\bumpeq \overline{n_1}, \ldots, \overline{n_k}$ eine abkürzende Schreibweise für eben diese Zeichenreihe.
Ebenfalls sei $\tau(x_0, x_1, \ldots, x_k, y, z)$ der Kern von σ (also ohne $\exists z$).

$p = 0$: Es gilt: $m = f(\vec{n}, 0) = g(\vec{n})$.

Betrachte die Ableitung

$$\dfrac{\dfrac{\dfrac{\sigma(y,v,\overline{0})}{\exists z : \tau(y,v,\overline{0},z)}(\bumpeq) \qquad \dfrac{\dfrac{[\tau(y,v,\overline{0},z)]^1}{y=(z)_0} \quad \dfrac{[\tau(y,v,\overline{0},z)]^1}{A_g((z)_0,v)}}{A_g(y,v)}}{A_g(y,v)}(1) \qquad \dfrac{\mathsf{PA}}{A_g(y,v) \to y=\overline{m}}}{y=\overline{m}}$$

Damit gilt: $\mathsf{PA} \vdash \sigma(y, v, \overline{0}) \to y = \overline{m}$.

$p+1$: Es gilt $m = f(\vec{n}, p+1) = h(\vec{n}, p, f(\vec{n}, p))$.

Nach IV gilt für $n := f(\vec{n}, p)$: $\mathsf{PA} \vdash \sigma(y, v, \overline{p}) \to y = \overline{n}$.

Betrachte folgende Ableitung:

$$\dfrac{\dfrac{\dfrac{\sigma(y,v,\overline{p+1})}{\exists z : \tau(y,v,\overline{p+1},z)}(\bumpeq) \qquad \begin{array}{c}[\tau(y,v,\overline{p+1},z)]^1 \\ \mathscr{D} \\ A_h(y,v,\overline{p},\overline{n})\end{array}}{A_h(y,v,\overline{p},\overline{n})}(1) \qquad \dfrac{\mathsf{PA}}{A_h(y,v,\overline{p},\overline{n}) \to y=\overline{m}}}{y=\overline{m}}$$

wobei

$$\mathscr{D} :\bumpeq \left\{ \dfrac{\dfrac{\tau(y,v,\overline{p+1},z)}{y=(z)_{\overline{p+1}}} \qquad \dfrac{\dfrac{\mathrm{IV}}{\overline{n}=(z)_{\overline{p}}} \quad \dfrac{\tau(y,v,\overline{p+1},z)}{A_h((z)_{S(\overline{p})},v,\overline{p},(z)_{\overline{p}})}}{A_h((z)_{S(\overline{p})},v,\overline{p},\overline{n})}}{A_h(y,v,\overline{p},\overline{n})} \right.$$

QED

Theorem 5.11 (Repräsentierbarkeit primitiv-rekursiver Funktionen und Prädikate)
Die primitiv-rekursiven Funktionen und Prädikate sind repräsentierbar.

Beweis. Aus den Hilfssätzen folgt mit Induktion über der Konstruktion von primitiv-rekursiven Funktionen die Behauptung. QED

Bemerkungen. (i) Die Repräsentation der Rekursion beruht wesentlich auf Listenerstellung. Die Liste enthält alle vorherigen (also bisher berechneten) Funktionswerte.

(ii) Alle repräsentierenden Formeln sind Σ_1. Es wird sogar schon bei der Komposition beschränkt quantifiziert. Es kann aber angenommen werden, dass die verwendeten Σ_1-Formeln strikt Σ_1 sind.

(iii) Das Theorem lässt sich verallgemeinern. Auch totale μ-rekursive Funktionen sind in PA repräsentierbar. (Vergleiche van Dalen, 2013, Theorem 7.5.6.)

6 Erster Unvollständigkeitssatz

Bisher wurden die technischen Voraussetzungen geschaffen, um den ersten Unvollständigkeitssatz zu beweisen. Nun wird zunächst ein Fixpunktsatz bewiesen. Mit diesem ist es dann unter der Voraussetzung der ω-Konsistenz von PA möglich, deren Unvollständigkeit (nach Gödel) zu zeigen. Dieses Ergebnis wird dann durch die Variante nach Rosser (1936) verbessert, der lediglich die Konsistenz von PA benötigt.

6.1 Fixpunktsatz

Definition 6.1 (Hilfsfunktionen) Zum Beweis des Fixpunktsatzes werden die folgenden primitiv-rekursiven Funktionen benötigt:

(i) $\mathit{Num} : \mathbb{N} \to \mathbb{N} : n \mapsto \ulcorner \overline{n} \urcorner = \begin{cases} 11 & \text{für } n = 0, \\ 2^{13+1} \cdot 3^{31+1} \cdot 5^{\mathit{Num}(n-1)+1} \cdot 7^{37+1} & \text{sonst.} \end{cases}$

Num ordnet jeder Zahl n die Codierung ihrer Repräsentation zu.

(ii) $s : \mathbb{N} \times \mathbb{N} \to \mathbb{N} : \langle n, m \rangle \mapsto s(n,m) := \mathit{Sub}(n, \ulcorner x_0 \urcorner, \mathit{Num}(m))$.

Dies ist ein Spezialfall der Substitutionsfunktion. In Formeln mit der Gödel-Zahl n wird die Variable x_0 durch den Term mit der Gödel-Zahl m ersetzt.

Theorem 6.2 (Fixpunktsatz) *Sei $B(x) \in \mathscr{L}_{\mathsf{PA}}$ eine beliebige Formel mit der einzigen freien Variable x (d. h. $\mathrm{FV}(B) = \{x\}$). Dann gibt es eine Formel $A \in \mathscr{L}_{\mathsf{PA}}$ mit:*

$$\mathsf{PA} \vdash A \leftrightarrow B(\overline{\ulcorner A \urcorner}).$$

Bemerkung. Dies bedeutet, dass es zu jeder (einstelligen) Eigenschaft B eine Formel A gibt, die logisch äquivalent zu der Aussage ist, dass sie (codiert und repräsentiert) die Eigenschaft B hat. Damit sagt A (metasprachlich) aus: *Ich habe die Eigenschaft B.*

Beweis. Definiere zunächst weitere Formeln und Terme:

(i) $\sigma(z,x,y)$ sei die Repräsentation der Hilfsfunktion s.

(ii) $\theta(x) :\simeq \exists y(B(y) \wedge \sigma(y,x,x))$.

(iii) $m :\simeq \ulcorner \theta(x_0) \urcorner$.

(iv) $A :\simeq \theta(\overline{m}) \simeq \exists y(B(y) \wedge \sigma(y, \overline{m}, \overline{m}))$.

Die repräsentierende Formel σ gibt es, da s primitiv-rekursiv ist.
Damit gilt (Eindeutigkeit der Abbildung):

$$\mathsf{PA} \vdash \forall y(\sigma(y, \overline{m}, \overline{m}) \leftrightarrow y = \overline{s(m,m)}).$$

Ebenfalls gilt: $\overline{s(m,m)} \simeq \overline{s(\ulcorner \theta(x_0) \urcorner, m)} \simeq \overline{\mathit{Sub}(\ulcorner \theta(x_0) \urcorner, \ulcorner x_0 \urcorner, \mathit{Num}(m))} \simeq \overline{\ulcorner \theta(\overline{m}) \urcorner}$.

Damit gilt:

$$\begin{aligned}\mathsf{PA} \vdash A \leftrightarrow \exists y(B(y) \wedge y = \overline{\ulcorner \theta(\overline{m}) \urcorner}) &\implies \mathsf{PA} \vdash A \leftrightarrow B(\overline{\ulcorner \theta(\overline{m}) \urcorner}) && (\exists y \text{ beseitigt})\\ &\implies \mathsf{PA} \vdash A \leftrightarrow B(\overline{\ulcorner A \urcorner}) && (\text{Definition von } A)\end{aligned}$$

Damit wurde die Behauptung gezeigt. QED

Korollar 6.3 (zum Fixpunktsatz) *Ist $T \supseteq \mathsf{PA}$ eine Theorie, welche die Arithmetik PA erweitert, so gibt es zu jeder einstelligen Eigenschaft $B \in \mathscr{L}_{\mathsf{PA}}$ einen Fixpunkt $A \in \mathscr{L}_{\mathsf{PA}}$ mit $T \vdash A \leftrightarrow B(\overline{\ulcorner A \urcorner})$.*

Beweis. Aus $\mathsf{PA} \vdash A \leftrightarrow B(\overline{\ulcorner A \urcorner})$ folgt sofort $T \vdash A \leftrightarrow B(\overline{\ulcorner A \urcorner})$. QED

Bemerkung. Die Argumentation im Beweis des Fixpunktsatzes könnte vereinfacht werden, falls die primitiv-rekursiven Funktionen in PA durch Terme repräsentiert werden könnten. Man würde einen Term t_s finden, der s in der formalen Arithmetik repräsentiert. Um dies zu erreichen, müsste man aber alle Definitionsgleichungen von primitiv-rekursiven Funktionen zu den Axiomen von PA (wie schon bei der Exponentiation geschehen) hinzunehmen.

6.2 Erster Unvollständigkeitssatz

Im Beweis des ersten Unvollständigkeitssatzes wird das primitiv-rekursive Prädikat $Bew(n,m)$ verwendet. Dabei codiert $Bew(\ulcorner \mathscr{D} \urcorner, \ulcorner A \urcorner)$, dass $\mathscr{D}$ eine Ableitung für die Formel A in PA ist. $\overline{\mathrm{Bew}}(x,y)$ sei die Repräsentation in der formalen Arithmetik.

Theorem 6.4 (Erster Unvollständigkeitssatz, Gödel, 1931)
Wenn PA ω-konsistent ist, dann ist PA unvollständig.

Beweis. Sei PA ω-konsistent. Definiere zunächst das einstellige Prädikat

$$\overline{\mathrm{Thm}}(y) :\simeq \exists x\, Bew(x,y).$$

Nach dem Fixpunktsatz gibt es zu der Eigenschaft $B(x) :\simeq \neg\overline{\mathrm{Thm}}(x)$ eine Formel A, so dass

$$\mathsf{PA} \vdash A \leftrightarrow \neg\overline{\mathrm{Thm}}(\overline{\ulcorner A \urcorner}). \tag{$\star$}$$

A sagt (metasprachlich) aus: "*Ich bin nicht beweisbar.*"
Im Folgenden werden sowohl die Annahme $\mathsf{PA} \vdash A$ als auch die Annahme $\mathsf{PA} \vdash \neg A$ jeweils zum Widerspruch geführt.

$$\begin{aligned}\mathsf{PA} \vdash A &\implies \text{Es gibt einen Beweis } \mathscr{D} \text{ von } A \text{ mit } \ulcorner \mathscr{D} \urcorner = n \in \mathbb{N}\\ &\implies \text{Für dieses } n\text{: } Bew(n, \ulcorner A \urcorner)\\ &\implies \text{Durch Repräsentation: } \mathsf{PA} \vdash \overline{\mathrm{Bew}}(\overline{n}, \overline{\ulcorner A \urcorner})\end{aligned}$$

$$\begin{aligned}
&\implies \text{Existenzeinführung: } \mathsf{PA} \vdash \exists x\, \overline{\mathrm{Bew}}(x, \overline{\ulcorner A \urcorner}) \\
&\implies \text{Definition von } \overline{\mathrm{Thm}}\text{: } \mathsf{PA} \vdash \overline{\mathrm{Thm}}(\overline{\ulcorner A \urcorner}) && \text{Widerspruch zu } (\star).
\end{aligned}$$

$$\begin{aligned}
\mathsf{PA} \vdash \neg A &\implies \mathsf{PA} \vdash \overline{\mathrm{Thm}}(\overline{\ulcorner A \urcorner}) && (\text{mit } (\star)) \\
&\implies \mathsf{PA} \vdash \exists x\, \overline{\mathrm{Bew}}(x, \overline{\ulcorner A \urcorner}) && (\text{nach Definition}) \\
&\implies \text{Es gilt nicht für alle } n \in \mathbb{N}\text{: } \mathsf{PA} \vdash \neg\overline{\mathrm{Bew}}(\overline{n}, \overline{\ulcorner A \urcorner}) && (\star\star) \\
&\quad (\text{mit } \omega\text{-Konsistenz})
\end{aligned}$$

Andererseits gilt immer noch:

$$\begin{aligned}
\mathsf{PA} \vdash \neg A &\implies \text{Mit PA konsistent: } \mathsf{PA} \nvdash A \\
&\quad (\text{da } \omega\text{-Konsistenz} \implies \text{Konsistenz}) \\
&\implies \text{Es gibt kein } n \in \mathbb{N} \text{ mit: } \mathit{Bew}(n, \ulcorner A \urcorner) \\
&\implies \text{Für alle } n \in \mathbb{N} \text{ gilt: } \neg\, \mathit{Bew}(n, \ulcorner A \urcorner) \\
&\implies \text{Für alle } n \in \mathbb{N} \text{ gilt: } \mathsf{PA} \vdash \neg\overline{\mathrm{Bew}}(\overline{n}, \ulcorner A \urcorner). && \text{Widerspruch zu } (\star\star).
\end{aligned}$$

Insgesamt wurde die Behauptung gezeigt. QED

Satz 6.5 (Wahrheit von G) *Die im Beweis konstruierte Aussage* $G \doteq \neg\overline{\mathrm{Bew}}(\overline{n}, \ulcorner A \urcorner)$ *ist im Standardmodell* $\mathbb{N} = \{\mathbb{N}, +, \cdot, \uparrow, 0\}$ *gültig.*

Beweis. Angenommen nicht, also $\mathbb{N} \nvDash G$.
Nach Konstruktion $(\star)$ gilt dann für $m := \ulcorner G \urcorner \in \mathbb{N}$: $\mathbb{N} \nvDash \neg\overline{\mathrm{Thm}}(\overline{m})$.
Da $\mathbb{N}$ eine Struktur und $\neg\overline{\mathrm{Thm}}(\overline{m})$ eine Aussage ist, gilt: $\mathbb{N} \vDash \overline{\mathrm{Thm}}(\overline{m})$. Also nach Konstruktion von $\overline{\mathrm{Thm}}$: $\mathbb{N} \vDash \exists x\, \mathit{Bew}(x, \overline{m})$.
Damit gäbe es eine natürliche Zahl $n \in \mathbb{N}$ mit

$$\mathbb{N} \vDash \overline{\mathrm{Bew}}(\overline{n}, \overline{m}) \tag{$\dagger$}$$

Da Bew ein primitiv-rekursives Prädikat ist, würde aus $\langle n, m \rangle \notin \mathit{Bew}$ sofort aufgrund der Repräsentierbarkeit folgen, dass $\mathsf{PA} \vdash \neg\overline{\mathrm{Bew}}(\overline{n}, \overline{m})$, und aufgrund $\mathbb{N} \vDash \mathsf{PA}$ auch $\mathbb{N} \vDash \neg\overline{\mathrm{Bew}}(\overline{n}, \overline{m})$. Dies widerspricht jedoch $(\dagger)$.
Es müsste also $\langle n, m \rangle \in \mathit{Bew}$ gelten, und aufgrund Repräsentierbarkeit $\mathsf{PA} \vdash \overline{\mathrm{Bew}}(\overline{n}, \overline{m})$.
Mit den Regeln des Kalküls folgt daraus sofort $\mathsf{PA} \vdash \exists x \overline{\mathrm{Bew}}(x, \overline{m})$, also $\mathsf{PA} \vdash \overline{\mathrm{Thm}}(\overline{m})$.
Mit $(\star)$ folgt nun $\mathsf{PA} \vdash \neg G$. Widerspruch zum Unvollständigkeitssatz.
Also gilt tatsächlich: $\mathbb{N} \vDash G$. QED

6.3 Theorem von Rosser

Für den Unvollständigkeitssatz in der Variante nach Rosser werden folgende Eigenschaften von Formeln in $\mathscr{L}_{\mathsf{PA}}$ benötigt.

Lemma 6.6 (Eigenschaften von $\mathscr{L}_{\mathsf{PA}}$-Formeln)
(i) Für alle Δ_0-Formeln A und B gilt:

$$\mathsf{PA} \vdash A \vee B \implies \mathsf{PA} \vdash A \text{ oder } \mathsf{PA} \vdash B.$$

(ii) Jede Σ_1-Formel A ist logisch äquivalent zu einer Formel A', so dass A' die selben freien Variablen wie A hat und eine strikte Σ_1-Formel ist.

Beweis. (i) Ergibt sich aus Proposition 2.10 (Vollständigkeit für Δ_0-Aussagen).

(ii) Vergleiche Boolos, Burgess & Jeffrey (2007), S. 204–207. QED

Theorem 6.7 (Rosser, 1936) *Wenn* PA *konsistent ist, dann ist* PA *unvollständig.*

Beweisskizze. Zunächst benötigt man, dass die Funktion $neg(n)$, die der Gödel-Zahl einer Formel $A \in \mathscr{L}_{\mathsf{PA}}$ die Gödel-Zahl ihrer Negation zuordnet (d. h. $neg(\ulcorner A \urcorner) = \ulcorner \neg A \urcorner$), primitiv-rekursiv und damit repräsentierbar ist.
Dann definiert man folgendes Prädikat:

$$\overline{\mathrm{Ross}}(x) :\simeq \forall y(\overline{\mathrm{Bew}}(y,x) \to \exists z < y : \overline{\mathrm{Bew}}(z, \overline{\mathrm{neg}}(x))).$$

Es sagt $Ross(\ulcorner A \urcorner)$ Folgendes aus: Wenn es für die Formel A in PA einen Beweis mit der Gödel-Zahl m gibt, dann findet sich ein Beweis von $\neg A$ in der Arithmetik mit kleinerer Gödel-Zahl. Dadurch wird in diesem Prädikat die Beweisbarkeit einer Formel mit einer schwachen Version der Konsistenz verbunden.
Nach obigem Lemma 6.6(ii) kann man davon ausgehen, dass die Formel $\overline{\mathrm{Ross}}(x)$ eine strikte Σ_1-Formel ist, und somit das verwendete $\overline{\mathrm{Bew}}(x,y)$ eine Δ_0-Formel ist.
Für $\overline{\mathrm{Ross}}(x) \in \mathscr{L}_{\mathsf{PA}}$ gibt es nach dem Fixpunktsatz eine Formel A mit

$$\mathsf{PA} \vdash A \leftrightarrow \overline{\mathrm{Ross}}(\overline{\ulcorner A \urcorner}) \tag{$\dagger$}$$

Damit sagt A aus: *Wenn ich beweisbar bin, dann ist meine Negation vor mir beweisbar.*
Mit der Voraussetzung, dass PA konsistent ist, wird nun analog zu Gödel in einem Widerspruchsbeweis gezeigt, dass $\mathsf{PA} \nvdash A$ und $\mathsf{PA} \nvdash \neg A$.

$$\begin{aligned}
\mathsf{PA} \vdash A &\implies \text{Es gibt einen Beweis } \mathscr{D} \text{ für } A \text{ mit } \ulcorner \mathscr{D} \urcorner = n \in \mathbb{N}\\
&\implies \text{Für dieses } n \text{ gilt: } \overline{\mathrm{Bew}}(n, \ulcorner A \urcorner)\\
&\implies \text{Durch Repräsentation: } \mathsf{PA} \vdash \overline{\mathrm{Bew}}(\overline{n}, \overline{\ulcorner A \urcorner})\\
&\implies \text{Aus } (\dagger) \text{ und der Def. von } \overline{\mathrm{Ross}} \text{ folgt: } \mathsf{PA} \vdash \exists y < \overline{n} : \overline{\mathrm{Bew}}(y, \overline{\ulcorner \neg A \urcorner})\\
&\implies \text{Damit gilt: } \mathsf{PA} \vdash \textstyle\bigvee_{k<n} \overline{\mathrm{Bew}}(\overline{k}, \overline{\ulcorner \neg A \urcorner})\\
&\implies \mathsf{PA} \vdash \overline{\mathrm{Bew}}(\overline{0}, \overline{\ulcorner \neg A \urcorner}) \text{ oder } \dots \text{ oder } \mathsf{PA} \vdash \overline{\mathrm{Bew}}(\overline{n-1}, \overline{\ulcorner \neg A \urcorner})
\end{aligned}$$

(Dies folgt mit Lemma 6.6(i), da $\overline{\mathrm{Bew}}(x,y)$ eine Δ_0-Formel ist.)

$$\begin{aligned}
&\Longrightarrow \text{ Damit gibt es ein } k < n \text{ mit: } \mathsf{PA} \vdash \overline{\text{Bew}}(\overline{k}, \overline{\ulcorner \neg A \urcorner}) \\
&\Longrightarrow \text{ Damit gilt: } \mathsf{PA} \vdash \neg A. \quad \text{Widerspruch zur Konsistenz von } \mathsf{PA}.
\end{aligned}$$

$$\begin{aligned}
\mathsf{PA} \vdash \neg A &\Longrightarrow \text{ Es gibt einen Beweis } \mathscr{D} \text{ für } \neg A \text{ mit } \ulcorner \mathscr{D} \urcorner = n \in \mathbb{N} \\
&\Longrightarrow \mathsf{PA} \vdash \overline{\text{Bew}}(\overline{n}, \overline{\ulcorner \neg A \urcorner}) \\
&\Longrightarrow \mathsf{PA} \vdash \forall y > \overline{n} : (\exists z < y : \overline{\text{Bew}}(z, \overline{\ulcorner \neg A \urcorner})) \qquad (\ddagger)
\end{aligned}$$

Aufgrund der Konsistenz von PA gilt:

$$\begin{aligned}
\mathsf{PA} \nvdash \neg A &\Longrightarrow \text{ Für alle } k \in \mathbb{N}\text{: nicht } \mathit{Bew}(k, \ulcorner A \urcorner) \\
&\Longrightarrow \text{ Für alle } k \in \mathbb{N}\text{: } \mathsf{PA} \vdash \neg\overline{\text{Bew}}(\overline{k}, \overline{\ulcorner A \urcorner}) \\
&\Longrightarrow \text{ Für alle } k \in \mathbb{N}\text{: } \mathsf{PA} \vdash \forall y < \overline{k} \neg\overline{\text{Bew}}(y, \overline{\ulcorner A \urcorner}) \\
&\Longrightarrow \text{ Für alle } k \in \mathbb{N}\text{: } \mathsf{PA} \vdash \forall y(\overline{\text{Bew}}(y, \overline{\ulcorner A \urcorner}) \to y \geq \overline{k}) \\
&\Longrightarrow \text{ Mit } n := k+1\text{: } \mathsf{PA} \vdash \forall y(\overline{\text{Bew}}(y, \overline{\ulcorner A \urcorner}) \to y > \overline{n}) \\
&\Longrightarrow \text{ Mit } (\ddagger)\text{: } \mathsf{PA} \vdash \forall y(\overline{\text{Bew}}(y, \overline{\ulcorner A \urcorner}) \to \exists z < y : \overline{\text{Bew}}(z, \overline{\ulcorner \neg A \urcorner})).
\end{aligned}$$

Das heißt, $\mathsf{PA} \vdash \overline{\text{Ross}}(\overline{\ulcorner A \urcorner})$.
Mit $(\dagger)$ folgt $\mathsf{PA} \vdash A$. Widerspruch zur Konsistenz von PA. QED

7 Zweiter Unvollständigkeitssatz

Der zweite Unvollständigkeitssatz von Gödel zeigt, dass ein Beweis der Widerspruchsfreiheit für PA in PA nicht möglich ist. Es wird also gezeigt: "*Die Widerspruchsfreiheit von* PA *ist formal nicht beweisbar.*"
Gödel selbst hat in seiner Arbeit den zweiten Unvollständigkeitssatz lediglich skizziert. Ein detaillierter Beweis wurde von Bernays in Hilbert & Bernays (1922) durchgeführt. Hier wird ebenfalls nur eine Skizze gegeben.
Das Folgende lässt sich ohne Probleme für beliebige rekursiv axiomatisierte Erweiterungen von PA übertragen. Es ist nicht entscheidend, welche Axiome konkret in das Axiomensystem aufgenommen werden. Wir behandeln nur den speziellen Fall der eigentlichen PA.

Notation. (i) Im Folgenden wird das Falsum ($\bot$) als abkürzende Schreibweise für die $\mathscr{L}_{\mathsf{PA}}$-Formel $0 = 1$ verwendet.

(ii) Der Gödel-Satz "*Ich bin nicht beweisbar*" wird wieder durch G bezeichnet.

(iii) Bei Ableitbarkeitsaussagen für PA verzichten wir an einigen Stellen auf den Verweis auf PA. Das bedeutet: $\vdash A$ steht im Folgenden immer für $\mathsf{PA} \vdash A$.

(iv) Als ergänzende Literatur zu diesem Abschnitt wird empfohlen: Boolos, Burgess & Jeffrey (2007), § 18.

7.1 Grundidee

Bevor technische Details zum zweiten Unvollständigkeitssatz skizziert werden, soll zunächst die Grundidee angegeben werden.
Die Widerspruchsfreiheit (Konsistenz) von PA wird durch folgende Relation charakterisiert:

$$Cons(\mathsf{PA}) \;:\Longleftrightarrow\; \text{nicht } Thm(\ulcorner 0 = 1 \urcorner) \;\Longleftrightarrow\; \text{nicht } \exists x : \overline{\mathrm{Bew}}(x, \ulcorner 0 = 1 \urcorner).$$

In dieser Notation hat der erste Unvollständigkeitssatz folgende Struktur:

$$Cons(\mathsf{PA}) \;\Longrightarrow\; \begin{cases} \text{(a) nicht } Thm(\ulcorner G \urcorner), \\ \text{(b) falls PA } \omega\text{-konsistent: nicht } Thm(\ulcorner \neg G \urcorner). \end{cases}$$

Durch Internalisierung der metasprachlichen Aussage (a) kann Folgendes in PA bewiesen werden:

$$\mathsf{PA} \vdash \neg\overline{\mathrm{Thm}}(\ulcorner 0 = 1 \urcorner) \to \neg\overline{\mathrm{Thm}}(\ulcorner G \urcorner) \qquad (\star)$$

Aufgrund der Fixpunkteigenschaft von G gilt ebenfalls:

$$\mathsf{PA} \vdash G \leftrightarrow \neg\overline{\mathrm{Thm}}(\ulcorner G \urcorner).$$

Damit erhält man:

$$\mathsf{PA} \vdash \neg\overline{\mathrm{Thm}}(\ulcorner 0 = 1 \urcorner) \to G.$$

Gäbe es also einen Beweis für die Konsistenz von PA in PA (d. h. $\mathsf{PA} \vdash \overline{\mathrm{Cons}(\mathsf{PA})}$), dann hätte man einen Beweis für den Gödel-Satz G gefunden. Dies steht aber im Widerspruch zum ersten Unvollständigkeitssatz.
Problematisch ist lediglich die Internalisierung ($\star$). Dazu wird zunächst diskutiert, was ein Beweisbarkeitsprädikat ist.

7.2 Beweisbarkeitsprädikat

Definition 7.1 (Beweisbarkeitsprädikat) Eine Formel $B(x)$ mit genau einer freien Variable x heißt *Beweisbarkeitsprädikat* in einer Theorie T, falls für alle Aussagen $A \in \mathscr{L}_{\mathsf{PA}}$ gilt: *Beweisbarkeitsprädikat*

(BL1) $T \vdash A \implies T \vdash B(\overline{\ulcorner A \urcorner})$;

(BL2) $T \vdash B(\overline{\ulcorner A \to B \urcorner}) \to (B(\overline{\ulcorner A \urcorner}) \to B(\overline{\ulcorner B \urcorner}))$;

(BL3) $T \vdash B(\overline{\ulcorner A \urcorner}) \to B(\overline{\ulcorner B(\overline{\ulcorner A \urcorner}) \urcorner})$.

Bemerkungen. (i) BL steht als Abkürzung für Bernays und Löb.

(ii) BL2 ist die Internalisierung des modus ponens und BL3 die Internalisierung von BL1.

(iii) $B(x) :\simeq (x = x)$ ist trivialerweise ein Beweisbarkeitsprädikat.

Notation. Im Folgenden wird als Abkürzung für $B(\overline{\ulcorner A \urcorner})$ die modallogische Schreibweise $\Box A$ verwendet. Damit erfüllt ein Beweisbarkeitsprädikat B folgende Axiome:

(BL1) $\vdash A \implies \vdash \Box A$;

(BL2) $\vdash \Box(A \to B) \to (\Box A \to \Box B)$;

(BL3) $\vdash \Box A \to \Box\Box A$;

Lemma 7.2 *Die Formel* $\overline{\mathrm{Thm}}(x)$ *ist ein Beweisbarkeitsprädikat.*

Beweisskizze. (BL1): Sei $\vdash A$ für eine Aussage $A \in \mathscr{L}_{\mathsf{PA}}$ gegeben.
Dann gibt es einen Beweis von A.
Dann gilt (nach Konstruktion): $Thm(\ulcorner A \urcorner)$.
Durch Repräsentation erhält man: $\vdash \overline{\mathrm{Thm}}(\overline{\ulcorner A \urcorner})$.

(BL2) und (BL3): Die Beweise sind technisch aufwändiger und verbleiben hier unbewiesen. QED

7.3 Satz von Löb

Zum Beweis des zweiten Unvollständigkeitssatzes benötigt man dann noch den *Satz von Löb*.

Satz 7.3 (Löb, 1955) *Für ein beliebiges Beweisbarkeitsprädikat $\Box$ und alle Aussagen $A \in \mathscr{L}_{\mathsf{PA}}$ gilt:*

$$\vdash \Box A \to A \implies \vdash A.$$

Beweis. Sei $A \in \mathscr{L}_{\mathsf{PA}}$ beliebige Aussage, so dass $\vdash \Box A \to A$ gilt.
Zu der einstelligen Eigenschaft $(\Box(\cdot) \to A)$ gibt es nach dem Fixpunktsatz eine Formel B, so dass Folgendes gilt:

$$\vdash B \leftrightarrow (\Box B \to A) \tag{$\star\star$}$$

Daraus folgt zunächst: $\vdash B \to (\Box B \to A)$.

$$\begin{array}{lll}
\implies & \vdash \Box(B \to (\Box B \to A)) & \text{(mit BL1)} \\
\implies & \vdash \Box B \to \Box(\Box B \to A) & \text{(mit BL2)} \\
\implies & \vdash \Box B \to (\Box\Box B \to \Box A) & \text{(mit BL2)} \\
\implies & \vdash (\Box B \to \Box\Box B) \to (\Box B \to \Box A) & (\text{mit } A \to (B \to C) \dashv\vdash (A \to B) \to (A \to C)) \\
\implies & \vdash \Box B \to \Box A & \text{(mit BL3 und modus ponens)} \\
\implies & \vdash \Box B \to A \quad (\star\star\star) & (\text{nach Vor. gilt } \vdash \Box A \to A\text{; Kettenschluss}) \\
\implies & \vdash B & (\text{mit } (\star\star) \text{ von rechts und modus ponens}) \\
\implies & \vdash \Box B & \text{(mit BL1)} \\
\implies & \vdash A & (\text{mit } (\star\star\star) \text{ und modus ponens})
\end{array}$$

Damit wurde der Satz von Löb gezeigt. QED

Bemerkung. Wie die modallogische Notation schon nahelegt, kann anstatt der Theorie der Beweisbarkeitsprädikate auch Modallogik betrieben werden:

(BL1) entspricht der *Nezessitationsregel*.

(BL2) entspricht der *Distributivität*.

(BL3) entspricht dem modallogischen *Axiom 4*.

Damit entsprechen die Axiome (BL1)-(BL3) dem modallogischen System K4.
In K4 kann der Satz von Löb nicht bewiesen werden, da hier der Fixpunktsatz fehlt. Entsprechend wird der Satz von Löb in der modallogischen Behandlung als Regel hinzugenommen.
Es ist GL = (BL1)-(BL3)+(Satz von Löb) ein System, in dem die modallogische Theorie der Beweisbarkeitsprädikate behandelt wird. ("GL" steht für Gödel–Löb.) Diese Theorie zeigt, dass sich grundlegende Eigenschaften der prädikatenlogischen Beweisbarkeitslogik in einer *aussagenlogischen* modalen Theorie behandeln lassen. Eine sehr gut lesbare

Einführung findet sich bei von Bülow (2006). Eine Darstellung, die auch die allgemeinen modallogischen Hintergründe behandelt, ist Friedrichsdorf (1992).

7.4 Zweiter Unvollständigkeitssatz

Theorem 7.4 (Zweiter Unvollständigkeitssatz, Gödel, 1931) *Wenn* PA *konsistent ist (d. h.* $\mathsf{PA} \nvdash 0 = 1$*), dann kann die Konsistenz von* PA *nicht in* PA *beweisen werden. Das heißt,*

$$\mathsf{PA} \nvdash \neg\overline{\mathrm{Thm}}(\ulcorner 0 = 1 \urcorner).$$

Beweis. Sei $\mathsf{PA} \nvdash 0 = 1$.
Angenommen $\mathsf{PA} \vdash \neg\Box(0 = 1)$ für ein Beweisbarkeitsprädikat $\Box$.
Dann gilt für beliebige Aussagen $A \in \mathscr{L}_{\mathsf{PA}}$: $\mathsf{PA} \vdash \Box(0 = 1) \to A$.
Dann gilt $\mathsf{PA} \vdash \Box(0 = 1) \to (0 = 1)$.
Da $\overline{\mathrm{Thm}}$ ein Beweisbarkeitsprädikat ist, folgt mit dem Satz von Löb: $\mathsf{PA} \vdash 0 = 1$. Widerspruch zur Annahme $\mathsf{PA} \nvdash 0 = 1$. QED

Korollar 7.5 *Für jedes Beweisbarkeitsprädikat* $\Box$ *gilt, dass die* $\mathscr{L}_{\mathsf{PA}}$*-Formel* $A :\simeq$ *“*Ich bin $\Box$-beweisbar*” tatsächlich in* PA *beweisbar ist.*

Beweis. Nach dem Fixpunktsatz gilt $\mathsf{PA} \vdash A \leftrightarrow \Box A$, und mit dem Satz von Löb folgt $\mathsf{PA} \vdash A$. QED

Korollar 7.6 *Wenn* PA *konsistent ist, dann hat* PA *kein (intern definiertes) Wahrheitsprädikat.*

Beweis. Angenommen $\overline{\mathrm{True}}(x)$ wäre ein Wahrheitsprädikat.
Das bedeutet, dass für jede Aussage $A \in \mathscr{L}_{\mathsf{PA}}$ gilt:

$$\mathsf{PA} \vdash A \leftrightarrow \overline{\mathrm{True}}(\ulcorner A \urcorner).$$

Insbesondere ist $\overline{\mathrm{True}}(x)$ ein Beweisbarkeitsprädikat (leicht zu zeigen).
Damit gilt (mit dem Satz von Löb): $\mathsf{PA} \vdash A$ für jedes $A \in \mathscr{L}_{\mathsf{PA}}$.
Damit ist PA inkonsistent. Widerspruch. QED

8 Philosophische Bemerkungen

Zum Abschluss sollen noch einige philosophische Bemerkungen im Umfeld der beiden Begriffe Vollständigkeit und Unvollständigkeit gemacht werden.

8.1 Vollständigkeit

Aus der Vollständigkeit der Prädikatenlogik folgt die Vollständigkeit für jede rekursiv axiomatisierbare, erststufige Theorie T:

$$T \vDash A \iff T \vdash A.$$

Es kann T auch durch die Axiome der formalen Arithmetik gegeben sein. In diesem Fall gilt:

$$\mathsf{PA} \vDash A \iff \mathsf{PA} \vdash A.$$

Das bedeutet, dass genau diejenigen Formeln, die in allen PA-Modellen gültig sind, aus den Axiomen von PA abgeleitet werden können. Insofern ist PA vollständig.
Die Unvollständigkeit von PA verweist darauf, dass der Gödel-Satz G im Standardmodell $\mathbb{N}$ von PA zwar gültig ist, aber aus den Axiomen von PA nicht ableitbar ist. Das heißt,

$$\mathbb{N} \vDash G, \text{ aber } \mathsf{PA} \nvdash G.$$

8.2 Nichtstandardmodelle

Modelltheoretisch bezieht sich die Vollständigkeit auf Gültigkeit in allen Modellen. Die Unvollständigkeit aber bezieht sich auf Gültigkeit in *einem speziellen Modell*, nämlich dem Standardmodell $\mathbb{N}$.
Es muss also Nichtstandardmodelle $\mathfrak{M}$ von PA geben, in welchen der Gödel-Satz G nicht gültig ist (d. h. $\mathfrak{M} \nvDash G$). Ansonsten wäre G nach dem Vollständigkeitssatz aus den Axiomen ableitbar.
Auch gibt es Nichtstandardmodelle von PA, in denen der Satz "PA *ist inkonsistent*", also $\overline{\mathrm{Thm}}(\ulcorner\bot\urcorner)$, wahr ist. Ansonsten wäre der Satz "*Die* PA *ist konsistent*", also $\neg\overline{\mathrm{Thm}}(\ulcorner\bot\urcorner)$, in allen Modellen wahr und entsprechend nach dem Vollständigkeitssatz beweisbar. (Vergleiche Franzén, 2005.)
Man könnte auch modelltheoretisch den Unvollständigkeitssatz durch die explizite Konstruktion geeigneter Nichtstandardmodelle direkt beweisen. Das bedeutet, dass man zwei Modelle $\mathfrak{M}_1$, $\mathfrak{M}_2$ von PA konstruiert, so dass für ein geeignete Aussage $A \in \mathscr{L}_{\mathsf{PA}}$ gilt:

$$\mathfrak{M}_1 \vDash A \quad \text{und} \quad \mathfrak{M}_2 \vDash \neg A.$$

Damit entspricht die Unabhängigkeit einer Aussage von einem Axiomensystem der Unvollständigkeit des Axiomensystems. Eine solche Aussage wurde z. B. von Paris &

Harrington (1977) angegeben. Vergleiche auch Putnam (2000) zu Resultaten von Kripke. Auch Gentzens (1943) direkter Nachweis, dass ε_0-Induktion nicht in der Arithmetik erster Stufe ableitbar ist, kann man als Nachweis einer "wahren" mathematischen Aussage ansehen, die nicht in PA ableitbar ist.

8.3 Logik zweiter Stufe

Die in Logik zweiter Stufe axiomatisierte Arithmetik PA^2 ist *kategorisch*: Je zwei Modelle sind (bezüglich der Standard-Semantik) isomorph. Das bedeutet, dass die Axiome alle Modelle bis auf Isomorphie charakterisieren.

kategorisch

Damit gilt für jede Aussage $A \in \mathscr{L}_{\mathsf{PA}}$:

$$\text{Entweder } \mathsf{PA}^2 \models A \text{ oder } \mathsf{PA}^2 \models \neg A.$$

(Es wird $\models$ hierbei im Sinne der Standard-Semantik verstanden.)
Dies gilt, da

(i) für jedes Modell $\mathfrak{M}$ gilt:

$$\mathfrak{M} \models A \iff \mathfrak{N} \models A$$

für ein ausgezeichnetes Modell $\mathfrak{N}$;

(ii) für jede Aussage A gilt (nach Definition von Strukturen) immer:

$$\mathfrak{M} \models A \quad \text{oder} \quad \mathfrak{M} \models \neg A.$$

Die PA^2 bleibt aber syntaktisch unvollständig: Es gibt Aussagen $A \in \mathscr{L}_{\mathsf{PA}}$, so dass $\mathsf{PA}^2 \nvdash A$ und $\mathsf{PA}^2 \nvdash \neg A$.

8.4 Rolle des Induktionsschemas

Für die Unvollständigkeit ist nicht alleine das Induktionsschema verantwortlich. Dies kann anhand der *Robinson-Arithmetik* Q gezeigt werden, die kein Induktionsschema enthält und endlich axiomatisierbar ist (siehe Definition 9.1). Wegen des Fehlens des Induktionsschemas ist z. B. die Formel $x + y = y + x$ in nicht ableitbar, obwohl für alle Ziffern $\overline{n}, \overline{m}$ die Aussage $\overline{n} + \overline{m} = \overline{m} + \overline{n}$ ableitbar ist. Es ist Q also schwächer als PA. Jedoch ist auch Q unvollständig.

Robinson-Arithmetik

Als Grundlage für beweistheoretische Untersuchungen wählt man häufig die auf Skolem (1923) zurückgehende *primitiv-rekursive Arithmetik* PRA. Hierbei handelt es sich um eine quantorenfreie Theorie, die aus folgenden arithmetischen Axiomen besteht:

primitiv-rekursive Arithmetik

(i) $S(x) \neq 0$.

(ii) $S(x) = S(y) \to x = y$.

(iii) Rekursive Definitionsgleichungen für alle primitiv-rekursiven Funktionen.

(iv) Das quantorenfreie Induktionsschema:

$$\frac{A(0) \qquad \begin{array}{c}[A(x)]\\ A(S(x))\end{array}}{A(x)}$$

PRA wird dabei meist als der finite Kern der Arithmetik angesehen, in dem sich Widerspruchsfreiheitsbeweise formalisieren lassen (unter Hinzunahme geeigneter weiterer Prinzipien wie transfiniter Induktion).

8.5 Heyting-Arithmetik

Die für den deduktiven Abschluss für das Axiomensystem von PA verwendete Logik muss nicht die *klassische* Prädikatenlogik sein. Verwendet man die schwächere *intuitionistische Logik*, in der insbesondere das *tertium non datur* $A \vee \neg A$ nicht ableitbar ist, so erhält man die *Heyting-Arithmetik* HA. Im Unterschied zu PA hat HA die *Disjunktionseigenschaft*, d. h. es gilt: *Heyting-Arithmetik*

$$\mathsf{HA} \vdash A \vee B \implies \mathsf{HA} \vdash A \text{ oder } \mathsf{HA} \vdash B.$$

Da HA in PA enthalten ist, gilt insbesondere für den Gödel-Satz G:

$$\mathsf{HA} \nvdash G \quad \text{und} \quad \mathsf{HA} \nvdash \neg G.$$

Damit kann $G \vee \neg G$ nicht in HA ableitbar sein, da sonst mit der Disjunktionseigenschaft $\mathsf{HA} \vdash G$ oder $\mathsf{HA} \vdash \neg G$ gelten müsste.
Folglich kann das tertium non datur $A \vee \neg A$ auch in HA nicht abgeleitet werden. Allgemein kann gezeigt werden:

Theorem 8.1 (de Jongh, 1970) *Für jede in der intuitionistischen Logik nicht ableitbare Formel kann eine arithmetische Substitutionsinstanz gefunden werden, die in* HA *nicht ableitbar ist.*

9 Unentscheidbarkeit der Prädikatenlogik

Die Prädikatenlogik ist vollständig, aber unentscheidbar. Zum Nachweis der Unentscheidbarkeit genügt die Betrachtung eines unentscheidbaren Problems für eine Theorie, die in der Prädikatenlogik formalisierbar ist. Aus der Entscheidbarkeit der Prädikatenlogik würde dann die Entscheidbarkeit des unentscheidbaren Problems folgen. Die Prädikatenlogik kann somit nicht entscheidbar sein. Man reduziert also das *Entscheidungsproblem* für die Prädikatenlogik auf ein anderes, schon negativ gelöstes Entscheidungsproblem. Bei Turing (1936) ist dies das Halteproblem für Turingmaschinen, bei Church (1936a) die Unentscheidbarkeit der β-Gleichheit für λ-Terme. Im Folgenden verwenden wir die Reduktion auf die Unentscheidbarkeit der endlich axiomatisierbaren Robinson-Arithmetik Q.

Entscheidungsproblem

Definition 9.1 (Robinson-Arithmetik Q) Die *Robinson-Arithmetik* Q ist der deduktive Abschluss folgender Axiome:

Robinson-Arithmetik

(Q1) $S(x) \neq 0$.

(Q2) $S(x) = S(y) \to x = y$.

(Q3) $y = 0 \vee \exists x\, Sx = y$.

(Q4) $x + 0 = x$.

(Q5) $x + S(y) = S(x + y)$.

(Q6) $x \cdot 0 = 0$.

(Q7) $x \cdot S(y) = S(x \cdot y + x)$.

Theorem 9.2 (Robinson, 1952) Q *ist unentscheidbar.*

Theorem 9.3 *Die Prädikatenlogik ist unentscheidbar.*

Beweis. Angenommen, die Prädikatenlogik (d. h. die Relation $\vdash A$ für prädikatenlogische Formeln A) ist entscheidbar. Dann ist mit $\mathrm{Q1}, \ldots, \mathrm{Q7} \vdash A$ bzw. $\vdash (\mathrm{Q1} \wedge \ldots \wedge \mathrm{Q7}) \to A$ auch Q entscheidbar. Widerspruch. QED

Literaturverzeichnis

G. S. Boolos, J. P. Burgess & R. C. Jeffrey, *Computability and Logic. Fifth Edition*, Cambridge University Press, 2007.

A. Church, *A Note on the Entscheidungsproblem*, Journal of Symbolic Logic **1** (1936a), 40–41.

A. Church, *An unsolvable problem of elementary number theory*, American Journal of Mathematics **58** (1936b), 345–363.

R. Dedekind, *Was sind und was sollen die Zahlen?*, Friedrich Vieweg und Sohn, 1888.

D. H. J. de Jongh, *The maximality of the intuitionistic propositional calculus with respect to Heyting's Arithmetic* (Abstract), Journal of Symbolic Logic **36** (1970), 606.

T. Franzén, *Gödel's Theorem. An Incomplete Guide to Its Use and Abuse*, A K Peters, 2005.

G. Frege, *Begriffsschrift*, Nebert, 1879.

G. Frege, *Die Grundlagen der Arithmetik. Eine logisch mathematische Untersuchung über den Begriff der Zahl*, Wilhelm Koebner, 1884.

U. Friedrichsdorf, *Einführung in die klassische und intensionale Logik*, Vieweg, 1992.

G. Gentzen, *Untersuchungen über das logische Schließen*, Mathematische Zeitschrift **39** (1935), 176–210, 405–431.

G. Gentzen, *Die Widerspruchsfreiheit der reinen Zahlentheorie*, Mathematische Annalen **112** (1936), 493–565.

G. Gentzen, *Beweisbarkeit und Unbeweisbarkeil von Anfangsfällen der transfiniten Induktion in der reinen Zahlentheorie*, Mathematische Annalen **119** (1943), 140–161.

K. Gödel, *Über die Vollständigkeit des Logikkalküls*, Dissertation, Universität Wien, 1929. (In: S. Feferman et al. (Hrsg.), *Kurt Gödel, Collected Works, Volume I, Publications 1929-1936*, Oxford University Press, 1986.)

K. Gödel, *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I*, Monatshefte für Mathematik und Physik **38** (1931), 173–198. (In: S. Feferman et al. (Hrsg.), *Kurt Gödel, Collected Works, Volume I, Publications 1929-1936*, Oxford University Press, 1986.)

H. Hermes, *Aufzählbarkeit, Entscheidbarkeit, Berechenbarkeit*, 3. Auflage, Springer, 1978.

D. Hilbert, *Die logischen Grundlagen der Mathematik*, Mathematische Annalen **88** (1922), 151–156.

D. Hilbert & P. Bernays, *Grundlagen der Mathematik II*, Springer, 1939.

D. W. Hoffmann, *Grenzen der Mathematik. Eine Reise durch die Kerngebiete der mathematischen Logik*, 2. Auflage, Springer, 2013.

D. W. Hoffmann, *Die Gödel'schen Unvollständigkeitssätze. Eine geführte Reise durch Kurt Gödels historischen Beweis*, 2. Auflage, Springer, 2017.

S. C. Kleene, *Introduction to Metamathematics*, Elsevier North-Holland, 2000.

M. H. Löb, *Solution of a problem of Leon Henkin*, Journal of Symbolic Logic **20**, 1955, 115–118.

Y. Matiyasevich, *Enumerable sets are diophantine*, Soviet Mathematics Doklady **11** (1970), 354–357.

J. Paris & L. Harrington, *A mathematical incompleteness in Peano Arithmetic*, in: J. Barwise (Hrsg.), *Handbook for Mathematical Logic*, S. 1133–1142, North-Holland, 1977.

G. Peano, *Arithmetices principia, nova methodo exposita*, Fratelli Bocca, 1889.

D. Prawitz, *Natural Deduction. A Proof-Theoretical Study*, Almqvist & Wiksell, 1965. Neuauflage 2006, Dover Publications.

H. Putnam, *Nonstandard models and Kripke's proof of the Gödel Theorem*, Notre Dame Journal of Formal Logic **41** (2000), 53–58.

J. B. Rosser, *Extensions of some theorems of Gödel and Church*, Journal of Symbolic Logic **1** (1936), 87–91.

R. Robinson, *An Essentially Undecidable Axiom System*, in: *Proceedings of the International Congress of Mathematicians, Cambridge, Massachussetts, U.S.A., August 30–September 6, 1950*, S. 729–730, American Mathematical Society, 1952.

T. Skolem, *Einige Bemerkungen zur axiomatischen Begründung der Mengenlehre*, in: *Matematikerkongressen i Helsingfors den 4–7 Juli 1922*, S. 217–232, Akademiska Bokhandeln, 1923.

W. Stegmüller, *Unvollständigkeit und Unentscheidbarkeit. Die metamathematischen Resultate von Gödel, Church, Kleene, Rosser und ihre erkenntnistheoretische Bedeutung*, 3. Auflage, Springer, 1973.

A. M. Turing, *On Computable Numbers, with an Application to the Entscheidungsproblem*, Proceedings of the London Mathematical Society, Series 2, **42** (1936), 230–265.

D. van Dalen, *Logic and Structure*, 5th ed., Springer, 2013.

C. von Bülow, *Beweisbarkeitslogik. Gödel, Rosser, Solovay*, Logos Verlag, 2006.

Sachverzeichnis