Allgemeine Algebra für Informatiker

Prof. Dr. Thomas Streicher

WS 2001/2002

Einführung

Im Verlauf dieser Vorlesung soll geklärt werden, was eine „algebraische Struktur“ ist, wie man neue algebraische Strukturen aus vorgegebenen konstruiert, und wie man sie durch „Homomorphismen“ miteinander in Bezug setzt.

Außerdem diskutieren wir, wie man *Klassen* algebraischer Strukturen *axiomatisch* beschreiben kann und welche Eigenschaften solche axiomatisch definierten Strukturklassen besitzen. Wenn die Axiome ausreichend einfach sind, z.B. nur aus Gleichungen bestehen, dann gibt es in solchen Klassen ausgezeichnete Strukturen, so genannte *initiale Strukturen*, die „bis auf Isomorphie“ eindeutig bestimmt sind. Die meisten Datenstrukturen können als solche initiale Strukturen für eine geeignete Gleichungstheorie beschrieben werden. Diese Herangehensweise ist in der Informatik als „Theorie der abstrakten Datentypen“ bekannt.

Weiters behandeln wir die *Verifikationen* von Programmen, die auf solchen abstrakten Datentypen operieren. Programme werden als Funktionen aufgefasst, die durch definierende Gleichungen beschrieben werden. Unter Verwendung dieser wird mit Hilfe von *Induktions*prinzipien der Nachweis erbracht, dass die Programme bestimmte erwünschte Eigenschaften erfüllen, d.h. „ganz einfach das tun, was man von ihnen erwartet“. Eine solche Verifikation ist natürlich nur *relativ* zu einer *formalen Spezifikation* der Anforderungen an das Programm möglich.

Des weiteren soll auch ganz allgemein in die Grundbegriffe abstrakter Strukturmathematik eingeführt werden, sofern diese für die (theoretische) Informatik relevant sind. Dies beinhaltet insbesondere eine Einführung in die *axiomatisch-deduktive Methode* und die Technik des *mathematischen Beweisens*. Außerdem geben wir eine Einführung in die Theorie der *induktiven Definitionen*, welche in fast allen Bereichen der (theoretischen) Informatik ein unerlässliches Werkzeug darstellen.

I Beispiele und Grundbegriffe

In diesem ersten einleitenden Teil sollen beispielhaft einige mathematische Grundstrukturen wie die natürlichen Zahlen, Monoide und Gruppen, Automaten, partielle Ordnungen, Verbände etc. vorgestellt werden. Aus dem Studium der Eigenschaften dieser Strukturen sollen allgemeine Begriffe und Methoden „destilliert" werden wie z.B.

- *Äquivalenz-* und *Kongruenzrelationen*
- *Faktorisierung* bezüglich Kongruenzrelationen
- *Homomorphismen*
- *induktive Definitionen* von Teilmengen einer gegebenen Menge
- *Repräsentation* abstrakter Strukturen in konkreten Strukturen

um nur die Wichtigsten zu nennen.
Die Definition und Betrachtung *allgemeiner* algebraischer Strukturen und verwandter Begriffe erfolgt in Teil 2.

I.1 Der „Datentyp" der natürlichen Zahlen

Wir werden nun einige Axiome angeben, die die natürlichen Zahlen eindeutig charakterisieren. Diese Axiome gehen auf den Mathematiker Giuseppe PEANO zurück und werden deshalb *Peano-Axiome* genannt. Später werden wir Algorithmen bzw. algorithmische Funktionen auf den natürlichen Zahlen betrachten und mit Hilfe der Axiome Eigenschaften von ihnen nachweisen.

Definition 1.1 (Peano-Axiome)
Sei N eine Menge, $\mathrm{zero} \in N$ *ein festes Element daraus und* $\mathrm{succ} : N \to N$ *eine Funktion. Das Tripel* $(N, \mathrm{zero}, \mathrm{succ})$ *heißt* Peano-Struktur *genau dann, wenn die folgenden drei Bedingungen gelten:*

(1) $\mathrm{zero} \neq \mathrm{succ}(x)$ *für alle* $x \in N$*;*

(2) $\mathrm{succ}(x) = \mathrm{succ}(y)$ *impliziert* $x = y$ *für alle* $x, y \in N$*;*

(3) *für jede Teilmenge* $P \subseteq N$ *mit*

  a) $\mathrm{zero} \in P$ *und*
  b) $\mathrm{succ}(P) \subseteq P$*, d.h.* $\forall x \in N.\ x \in P \Rightarrow \mathrm{succ}(x) \in P$

  gilt $P = N$*, d.h.* $\forall x \in N.\ x \in P$. ◊

Eine Menge X zusammen mit $z \in X$ und $s : X \to X$ kann man als *Zählstruktur* betrachten. Mit $z \in X$ fängt man an zu zählen, und mit s zählt man weiter. Man kann dann folgende Auflistung betrachten

$$z,\ s(z),\ s(s(z)),\ s(s(s(z))),\ \ldots,\ s^n(z), \ldots\ .$$

Intuitiv sind für den Fall $X = \mathbb{N}$ (Menge der natürlichen Zahlen), $z = 0$ und $s(n) = n + 1$ die Peano-Axiome aus Definition 1.1 erfüllt, denn

(1) durch Weiterzählen kann man niemals 0 erreichen;

(2) aus $n + 1 = m + 1$ folgt $n = m$;

(3) jede natürliche Zahl lässt sich von 0 aus erreichen, indem man endlich oft um 1 weiterzählt; wenn nun $P \subseteq \mathbb{N}$ und

  (a) $0 \in P$ und

  (b) $\forall n \in \mathbb{N}.\, n \in P \Rightarrow n + 1 \in P$,

  dann gilt für jedes $n \in \mathbb{N}$, dass $n \in P$:

$$\begin{array}{ccccccccc} P(0) & \Rightarrow & P(1) & \Rightarrow & \ldots & \Rightarrow & P(n-1) & \Rightarrow & P(n)\ . \\ \text{(a)} & \text{(b)} & & \text{(b)} & & \text{(b)} & & \text{(b)} & \end{array}$$

Später werden wir sehen, dass es „bis auf Isomorphie" nur eine Peano-Struktur gibt. Sei also $(N, \mathrm{zero}, \mathrm{succ})$ im Weiteren eine fixe, aber beliebige Peano-Struktur, deren Elemente wir im Lichte obiger Betrachtung als „natürliche Zahlen" bezeichnen werden. Wir werden nun — unter ausschließlicher Verwendung der Peano-Axiome — einige Eigenschaften von natürlichen Zahlen, d.h. Peano-Strukturen, nachweisen.

Lemma 1.2 *Sei* $(N, \mathrm{zero}, \mathrm{succ})$ *eine Peano-Struktur. Dann gilt*

(1) $\forall x \in N. x = \mathrm{zero} \vee \exists y \in N.\ x = \mathrm{succ}(y)$

(2) $\forall x \in N.\ x \neq \mathrm{succ}(x)$.

Das heißt, jedes Element einer Peano-Struktur ist entweder zero oder Nachfolger eines anderen Elementes, und jedes Element ist von seinem Nachfolger verschieden.

Beweis: (1) Sei $P = \{\mathrm{zero}\} \cup \overbrace{\{\mathrm{succ}(y) \mid y \in N\}}^{=\mathrm{succ}(N)}$. D.h., P ist die Menge aller x, für die gilt: $x = \mathrm{zero} \ \vee\ \exists\, y \in N. x = \mathrm{succ}(y)$. Wir zeigen, dass $P = N$. Wegen des dritten Peano-Axioms genügt es zu zeigen, dass (a) zero $\in P$ und (b) $\mathrm{succ}(P) \subseteq P$.

(a) gilt aufgrund der Definition von P. Nach Definition von P gilt auch $\mathrm{succ}(N) \subseteq P$ und somit $\mathrm{succ}(P) \subseteq \mathrm{succ}(N) \subseteq P$, also (b).

(2) Überlassen wir dem geneigten Leser als Übung. □

Im weiteren identifizieren wir *Eigenschaften* natürlicher Zahlen mit Teilmengen von N. Wenn P also eine Eigenschaft natürlicher Zahlen ist, so *identifizieren* wir P mit der Menge aller natürlichen Zahlen, die diese Eigenschaft haben.

$$P = \{\, n \in N \mid P(n) \,\} \subseteq N$$

Das dritte Peano-Axiom versteht sich dann als *Induktionsprinzip* für die natürlichen Zahlen, nämlich

$$P(\mathrm{zero}) \wedge (\forall x \in N.\ P(x) \Rightarrow P(\mathrm{succ}(x))) \Rightarrow \forall x \in N.\ P(x)$$

Wir betrachten nun einige Algorithmen auf natürlichen Zahlen und weisen einige ihrer wohlvertrauten Eigenschaften *rein formal* nach.

Definition 1.3 *Seien* $\mathrm{pred} : N \to N$ *und* $\mathrm{add}, \mathrm{mult} : N \times N \to N$ *Funktionen, die folgenden Gleichungen genügen*

(pred1)	$\mathrm{pred}(\mathrm{zero}) = \mathrm{zero}$
(pred2)	$\mathrm{pred}(\mathrm{succ}(n)) = n$
(add1)	$\mathrm{add}(n, \mathrm{zero}) = n$
(add2)	$\mathrm{add}(n, \mathrm{succ}(m)) = \mathrm{succ}(\mathrm{add}(n, m))$
(mult1)	$\mathrm{mult}(n, \mathrm{zero}) = \mathrm{zero}$
(mult2)	$\mathrm{mult}(n, \mathrm{succ}(m)) = \mathrm{add}(\mathrm{mult}(n, m), n)$ ◊

Diese Gleichungen lassen sich *operational* auch als Auswertungsregeln verstehen (und in funktionalen Programmiersprachen wie LISP, ML, etc. auch in dieser Form hinschreiben).

Bevor wir Eigenschaften obiger Funktionen nachweisen, betrachten wir folgendes Definitionsprinzip.

Definition 1.4 (Primitive Rekursion)
Für Funktionen $g : N^k \to N$ *und* $h : N^{k+2} \to N$ *sei* $f = R[g, h] : N^{k+1} \to N$ *folgendermaßen definiert:*

(PR1)	$f(\vec{x}, \mathrm{zero}) = g(\vec{x})$
(PR2)	$f(\vec{x}, \mathrm{succ}(n)) = h(\vec{x}, f(\vec{x}, n), n)$ ◊

Bemerkung: Mit Theorem 1.10 kann bewiesen werden, dass ein solches f tatsächlich existiert und auch eindeutig ist.

Übung. Wie kann man pred, add und mult mithilfe primitiver Rekursion definieren?

Jetzt zu den Eigenschaften:

Lemma 1.5 *Es gilt*

$$\text{add}(\text{add}(x, y), z) = \text{add}(x, \text{add}(y, z))$$

für alle $x, y, z \in N$.

Beweis: Seien $x, y \in N$ fest, aber beliebig. Wir zeigen durch Induktion über $z \in N$, dass die Eigenschaft

$$P(z) \;\equiv\; \text{add}(\text{add}(x, y), z) = \text{add}(x, \text{add}(y, z))$$

für alle $z \in N$ gilt. Wir zeigen zuerst, dass $P(\text{zero})$

$$\begin{array}{lll} & \text{add}(\text{add}(x, y), \text{zero}) & \text{(add1)} \\ = & \text{add}(x, y) & \text{(add1)} \\ = & \text{add}(x, \text{add}(y, \text{zero})) & \end{array}$$

Als nächstes zeigen wir für alle $z \in N$, dass $P(\text{succ}(z))$ aus $P(z)$ folgt. Sei $z \in N$ beliebig, aber fest. Wir nehmen an, dass $P(z)$ gilt (als so genannte Induktionshypothese), und folgern daraus, dass $P(\text{succ}(z))$ ebenfalls gilt.

$$\begin{array}{lll} & \text{add}(\text{add}(x, y), \text{succ}(z)) & \text{(add2)} \\ = & \text{succ}(\text{add}(\text{add}(x, y), z)) & \text{(I.H. } P(z)) \\ = & \text{succ}(\text{add}(x, \text{add}(y, z))) & \text{(add2)} \\ = & \text{add}(x, \text{succ}(\text{add}(y, z))) & \text{(add2)} \\ = & \text{add}(x, \text{add}(y, \text{succ}(z))) & \end{array}$$

Also folgt nach dem dritten Peano-Axiom, dass $P(z)$ für alle $z \in N$. Weil dies für beliebige $x, y \in N$ der Fall ist, folgt die Aussage des Lemmas. □

Als nächstes zeigen wir die Kommutativität von add, d.h. $\text{add}(x, y) = \text{add}(y, x)$. Dies geht jedoch nicht direkt, sondern erfordert folgenden Hilfssatz

Lemma 1.6 *Es gilt für alle* $x, y \in N$, *dass*

(1) $\text{add}(\text{zero}, x) = x$

(2) $\text{add}(\text{succ}(x), y) = \text{add}(x, \text{succ}(y))$.

Beweis: (1) Beweis durch Induktion über x. Wegen (add1) ist $\text{add}(\text{zero}, \text{zero}) = \text{zero}$. Sei nun $x \in N$ fest aber beliebig, und es gelte $\text{add}(\text{zero}, x) = x$. Dann

$$\begin{array}{lll} & \text{add}(\text{zero}, \text{succ}(x)) & \text{(add2)} \\ = & \text{succ}(\text{add}(\text{zero}, x)) & (\text{I.H. add}(\text{zero}, x) = x) \\ = & \text{succ}(x). & \end{array}$$

(2) Sei $x \in N$ fix, aber beliebig. Wir definieren die Eigenschaft $P(y)$ durch

$$P(y) \;\equiv\; \mathrm{add}(\mathrm{succ}(x), y) = \mathrm{add}(x, \mathrm{succ}(y))$$

und zeigen $\forall y \in N.\ P(y)$ mit Induktion. Es gilt $P(\mathrm{zero})$ wegen

$$\begin{array}{llr} & \mathrm{add}(\mathrm{succ}(x), \mathrm{zero}) & (\mathrm{add1}) \\ = & \mathrm{succ}(x) & (\mathrm{add1}) \\ = & \mathrm{succ}(\mathrm{add}(x, \mathrm{zero})) & (\mathrm{add2}) \\ = & \mathrm{add}(x, \mathrm{succ}(\mathrm{zero})) & \end{array}$$

Wir zeigen nun $\forall y \in N.\ P(y) \Rightarrow P(\mathrm{succ}(y))$. Sei $y \in N$ fix, aber beliebig, und es gelte die Induktionshypothese

$$(\mathrm{I.H.}) \quad P(y) \equiv \mathrm{add}(\mathrm{succ}(x), y) = \mathrm{add}(x, \mathrm{succ}(y))$$

Dann gilt auch $P(\mathrm{succ}(y))$ wegen

$$\begin{array}{llr} & \mathrm{add}(\mathrm{succ}(x), \mathrm{succ}(y)) & (\mathrm{add2}) \\ = & \mathrm{succ}(\mathrm{add}(\mathrm{succ}(x), y)) & (\mathrm{I.H.}) \\ = & \mathrm{succ}(\mathrm{add}(x, \mathrm{succ}(y))) & (\mathrm{add2}) \\ = & \mathrm{add}(x, \mathrm{succ}(\mathrm{succ}(y))) & \square \end{array}$$

Nun können wir die Kommutativität von add zeigen.

Lemma 1.7 *Für alle $x, y \in N$ gilt* $\mathrm{add}(x, y) = \mathrm{add}(y, x)$.

Beweis: Sei $x \in N$ beliebig, aber fix. Sei $P(y) \equiv \mathrm{add}(x, y) = \mathrm{add}(y, x)$. Wir zeigen $\forall y \in N.\ P(y)$ mit Induktion. Die Aussage $P(\mathrm{zero})$ gilt wegen

$$\begin{array}{llr} & \mathrm{add}(x, \mathrm{zero}) & (\mathrm{add1}) \\ = & x & (\text{Lemma } 1.6(1)) \\ = & \mathrm{add}(\mathrm{zero}, x) & \end{array}$$

Wir zeigen nun $\forall y \in N.\ P(y) \Rightarrow P(\mathrm{succ}(y))$. Sei $y \in N$ fix aber beliebig, und es gelte die I.H. $P(y)$. Wir zeigen unter dieser Annahme, dass $P(\mathrm{succ}(y))$

$$\begin{array}{llr} & \mathrm{add}(x, \mathrm{succ}(y)) & (\mathrm{add2}) \\ = & \mathrm{succ}(\mathrm{add}(x, y)) & (\mathrm{I.H.}) \\ = & \mathrm{succ}(\mathrm{add}(y, x)) & (\mathrm{add2}) \\ = & \mathrm{add}(y, \mathrm{succ}(x)) & (\text{Lemma } 1.6(2)) \\ = & \mathrm{add}(\mathrm{succ}(y), x) & \square \end{array}$$

Die folgenden vertrauten Eigenschaften der Multiplikation lassen sich ähnlich beweisen.

Lemma 1.8 *Für alle $x, y, z \in N$ gilt:*

(1) $\mathrm{mult}(x, \mathrm{add}(y, z)) = \mathrm{add}(\mathrm{mult}(x, y), \mathrm{mult}(x, z))$ *(Distributivität)*

(2) $\mathrm{mult}(\mathrm{add}(x, y), z) = \mathrm{add}(\mathrm{mult}(x, z), \mathrm{mult}(y, z))$ *(Distributivität)*

(3) $\mathrm{mult}(\mathrm{mult}(x, y), z) = \mathrm{mult}(x, \mathrm{mult}(y, z))$ *(Assoziativität)*

(4) $\mathrm{mult}(x, y) = \mathrm{mult}(y, x)$ *(Kommutativität)*

Beweis: Übung! (mit Induktion). □

Wir haben gesehen, dass der Nachweis der üblichen Eigenschaften der Addition und Multiplikation aus den Peano-Axiomen und den definierenden Gleichungen für add und mult einer gewissen Organisation bedarf. Die Einzelschritte jedoch sind alle eher trivial. Ein Vorteil dieser strengen Vorgangsweise, wo nur logische Schlussfolgerungen aus den Axiomen zugelassen sind, ist, dass man sicher sein kann, dass *alle Annahmen in die Axiome konzentriert sind.*

Als nächstes weisen wir nach, dass alle Peano-Strukturen *bis auf Isomorphie* gleich sind.

Definition 1.9 *Ein* Orbit *ist ein Tripel (A, a, t), das aus einer Menge A, einem ausgezeichneten Element $a \in A$ und einer Funktion $t : A \to A$ besteht.*

Seien $(A_1, a_1, t_1), (A_2, a_2, t_2)$ Orbits. Ein Orbit- Homomorphismus *von (A_1, a_1, t_1) nach (A_2, a_2, t_2) ist eine Funktion $h : A_1 \to A_2$ so dass*

(i) $h(a_1) = a_2$

(ii) $h(t_1(x)) = t_2(h(x))$ *für alle $x \in A_1$.*

wie in folgendem Diagramm veranschaulicht:

$$
\begin{tikzcd}
 & A_1 \arrow[r, "t_1"] \arrow[dd, "h"'] & A_1 \arrow[dd, "h"] \\
1 \arrow[ur, "a_1"] \arrow[dr, "a_2"'] & & \\
 & A_2 \arrow[r, "t_2"'] & A_2
\end{tikzcd}
$$

Wir schreiben $h : (A_1, a_1, t_1) \to (A_2, a_2, t_2)$ um auszudrücken, dass h ein Orbit-Homomorphismus von (A_1, a_1, t_1) nach (A_2, a_2, t_2) ist. ◊

Übung: Zeige die folgenden, später immer wieder verwendete Aussagen:

(1) Für jeden Orbit (A, a, t) ist die Identitätsfunktion $id_A(x) = x$ auch ein Orbit-Homomorphismus $id_A : (A, a, t) \to (A, a, t)$.

(2) Wenn $h_1 : (A_1, a_1, t_1) \to (A_2, a_2, t_2)$ und $h_2 : (A_2, a_2, t_2) \to (A_3, a_3, t_3)$ zwei Orbit-Homomorphismen sind, dann ist auch deren Hintereinanderausführung, die durch $(h_2 \circ h_1)(x) = h_2(h_1(x))$ definiert ist, ein Orbit-Homomorphismus $h_2 \circ h_1 : (A_1, a_1, t_1) \to (A_3, a_3, t_3)$.

Theorem 1.10 *Sei $(N, \text{zero}, \text{succ})$ eine Peano-Struktur und (A, a, t) ein Orbit. Dann gibt es genau einen Orbit-Homomorphismus $h : (N, \text{zero}, \text{succ}) \to (A, a, t)$.*

Beweis:

- Wir beweisen zuerst die *Eindeutigkeit.*

  Seien $h, k : (N, \text{zero}, \text{succ}) \to (A, a, t)$ Orbit-Homomorphismen. Wir definieren die Eigenschaft $P = \{x \in N \mid \; h(x) = k(x)\}$ und zeigen wie üblich $P = N$ unter Verwendung des 3. Peano-Axioms, d.h. mit Induktion:

  Einerseits ist zero $\in P$, weil $h(\text{zero}) = a = k(zero)$. Andererseits gilt auch $\text{succ}(P) \subseteq P$, also $P(x) \Rightarrow P(\text{succ}(x))$: Sei $x \in N$ fix, aber beliebig, und es gelte $h(x) = k(x)$. Dann folgt $h(\text{succ}(x)) = t(h(x)) = t(k(x)) = k(\text{succ}(x))$.

- Wir wenden uns nun der schwierigeren Aufgabe zu, einen solchen Orbit-Homomorphismus zu konstruieren.

  Dazu definieren wir die Menge $\mathcal{S}$ aller Relationen R von N nach A, die der Homomorphie-Eigenschaft genügen, also derjenigen $R \in \mathcal{P}(N \times A)$, so dass

  (a) $(\text{zero}, a) \in R$ und

  (b) $(\text{succ}(x), t(y)) \in R$, falls $(x, y) \in R$.

  Wir setzen

  $$H := \bigcap \mathcal{S} = \{(x, y) \in N \times A \mid \forall R \in \mathcal{S}.\ (x, y) \in R\}$$

  und zeigen im folgenden, dass H in der Tat der Graph des gesuchten Orbit-Homomorphismus ist.

  Man zeigt leicht (Übung!), dass

  (1) $H \in \mathcal{S}$

  Wir definieren weiters

  $$H' := \{(\text{zero}, a)\} \cup \{(\text{succ}(x), t(y)) \mid (x, y) \in H\}$$

  für welches gilt (Übung!), dass

(2) $H' \subseteq H$ (da ja $H \in \mathcal{S}$) und

(3) $H' \in \mathcal{S}$, also $H \subseteq H'$.

Aus (2) und (3) folgt somit $H = H'$. Das bedeutet explizit

(4) $(x, y) \in H$ genau dann, wenn
$(x = \text{zero} \wedge y = a) \vee (\exists (x', y') \in H.\ x = \text{succ}(x') \wedge y = t(y'))$.

Wir beweisen nun durch Induktion über N, dass H auch wirklich Graph einer Funktion ist, d.h., dass für alle $x \in N$ genau ein $y \in A$ existiert mit $(x, y) \in H$, oder formal

(5) $\forall x \in N.\ \exists!\, y \in A.\ (x, y) \in H$.

Für $x = \text{zero}$ gilt auf Grund von (4), dass $(\text{zero}, a) \in H$. Außerdem ist a auch das einzig mögliche y mit $(\text{zero}, y) \in H$, denn wegen des ersten Peano-Axioms kann zero nicht von der Form $\text{succ}(x')$ sein.

Für den Induktionschritt nehmen wir an, dass für ein beliebiges, aber fixes $x \in N$ als Induktionshypothese gelte, dass es genau ein $y \in A$ gibt mit $(x, y) \in H$. Dann gilt aber auch $(\text{succ}(x), t(y)) \in H$, da $H \in \mathcal{S}$ nach (1). Sei nun $(\text{succ}(x), z) \in H$ für ein weiteres z. Wenn wir zeigen können $t(y) = z$, sind wir fertig. Wegen (4) gibt es aber ein $(x', z') \in H$ mit $\text{succ}(x) = \text{succ}(x')$ und $z = t(z')$, da der Fall $\text{zero} = \text{succ}(x)$ ja durch das erste Peano-Axiom ausgeschlossen ist. Mit dem zweiten Peano-Axioms folgt $x = x'$, also $(x, z') \in H$, woraus auf Grund der Induktionshypothese folgt, dass $y = z'$ und somit auch $t(y) = t(z') = z$. Damit ist die Eindeutigkeit von $t(y)$ gezeigt.

Damit haben wir bewiesen, dass H der Graph einer Funktion $h : N \to A$ ist, wobei

(6) $h(x) = y$ genau dann, wenn $(x, y) \in H$.

Da $H \in \mathcal{S}$ ist, gelten definitionsgemäß die Homomorphie-Bedingungen $h(\text{zero}) = a$ und $h(\text{succ}(x)) = t(h(x))$. Wir haben also tatsächlich den gesuchten Homomorphismus $h : (N, \text{zero}, \text{succ}) \to (A, a, t)$ konstruiert. $\square$

Definition 1.11 *Ein Orbit* $(N, \text{zero}, \text{succ})$ *heißt* initial *genau dann, wenn für jeden Orbit* (A, a, t) *genau ein Orbit-Homomorphismus*

$$h : (N, \text{zero}, \text{succ}) \to (A, a, t)$$

existiert. $\diamondsuit$

Wir können nun Peano-Strukturen als die initialen Orbits charakterisieren.

Theorem 1.12 *Jeder initiale Orbit ist eine Peano-Struktur.*

Beweis: Sei (A, a, t) ein initialer Orbit. Wir müssen nachweisen, dass dieser Orbit die drei Peano-Axiome erfüllt. Dazu konstruieren wir einen neuen Orbit, indem wir „ganz vorne" ein neues Element anfügen.
Sei $*$ ein neues Element mit $* \notin A$. Dann setze $A' := \{*\} \cup A$ und $a' := *$, und definiere die Funktion $k : A' \to A$ wie folgt

$$k(*) = a \qquad \text{und} \qquad k(x) = t(x) \ \text{ für } x \in A\ .$$

Wir können die alte Menge in die neue einbetten[1] durch $i : A \hookrightarrow A' : x \mapsto x$. Offensichtlich gilt $k \circ i = t$. Die Funktion $t' : A' \to A'$ sei als $i \circ k$ definiert, wie in folgendem Diagramm veranschaulicht wird:

$$\begin{tikzcd}
A' \arrow[r, "k"] \arrow[dr, "t'"'] & A \arrow[d, hook, "i"] \\
& A'
\end{tikzcd}$$

Offenbar ist (A', a', t') ein Orbit. Sei $h : A \to A'$ der eindeutig bestimmte Orbit-Homomorphismus von (A, a, t) nach (A', a', t').
Die Abbildung k ist ein Orbit-Homomorphismus von (A', a', t') nach $(A, a, t,)$, da $k(a') = k(*) = a$ und $k \circ t' = k \circ i \circ k = t \circ k$.
Damit ist auch die Hintereinanderausführung $k \circ h : (A, a, t) \to (A, a, t)$ ein Orbit-Homorphismus. Wegen der Initialität von (A, a, t) ist aber $\mathrm{id}_A : (A, a, t) \to (A, a, t)$ der einzige Orbit-Homorphismus, der dafür in Frage kommt. Also gilt $k \circ h = id_A$.

Um die ersten beiden Peano-Axiome zu beweisen, beobachten wir, dass

(1) $h(a) = a' = * \notin A$ und

(2) $h(t(x)) = t'(h(x))) = i(k(h(x))) = i(x) = x \in A$ für alle $x \in A$.

Das erste Peano-Axiom, d.h. $a \neq t(x)$, gilt, weil $h(a) \notin A$, während $h(t(x)) \in A$.

Wir zeigen nun das zweite Peano-Axiom. Sei $t(x) = t(y)$. Dann gilt $x = h(t(x)) = h(t(y)) = y$ wegen (2) und somit $x = y$.

Für das dritte Peano-Axiom betrachte eine Teilmenge $P \subseteq A$, die a enthält ($a \in P$) und unter t abheschlossen ist ($t(P) \subseteq P$). Offenbar ist $(P, a, t_{|P})$ ebenfalls ein Orbit, wobei $t_{|P}$ die Einschränkung von t auf P ist, und die Inklusion $i : P \hookrightarrow A : x \mapsto x$ ein Orbit-Homomorphismus von $(P, a, t_{|P})$ nach (A, a, t).

[1]Die Notation mit $\hookrightarrow$ soll andeuten, dass es sich um eine Einbettung handelt, d.h. eine Abbildung i mit $i(x) = x$ für alle Argumente x.

Sei nun $h : A \to P$ der eindeutig bestimmte Orbit–Homomorphismus von (A, a, t) nach $(P, a, t_{|P})$. Dann ist $i \circ h$ ein Homomorphismus von (A, a, t) nach (A, a, t) und somit ist wieder $i \circ h = id_A$ wegen der Initialität von (A, a, t). Dann gilt aber $A \subseteq P$, da für jedes $x \in A$ gilt, dass $x = i(h(x)) = h(x) \in P$.
Da ja sowieso $P \subseteq A$, folgt $A = P$, wie verlangt. □

Somit erhalten wir folgende Äquivalenz:

Theorem 1.13 *Ein Orbit (A, a, t) ist genau dann initial, wenn (A, a, t) eine Peano-Struktur ist.*

Beweis: Mit Theorem 1.10 und Theorem 1.12. □

Es lässt sich nun leicht zeigen, dass Peano-Strukturen „bis auf Isomorphie“, also bis auf eine Umbenennung der Elemente, eindeutig bestimmt sind. Formal drückt man das folgendermaßen aus.

Definition 1.14 *Zwei Orbits (A, a, t) und (A', a', t') heißen* isomorph*, falls es zwei Orbit-Homomorphismen*

$$h_1 : (A, a, t) \to (A', a', t') \quad \text{und} \quad h_2 : (A', a', t') \to (A, a, t)$$

gibt, deren Verkettung die Identität ist, also $h_2 \circ h_1 = id_A$ und $h_1 \circ h_2 = id_{A'}$. ◊

Als Funktionen sind h_1 und h_2 dann notwendigerweise bijektiv.

Theorem 1.15 *Wenn (A_1, a_1, t_1) und (A_2, a_2, t_2) initiale Orbits sind, so sind sie auch isomorph.*

Beweis: Als Übung. □

Bemerkung. In der Theorie der „abstrakten Datentypen“ werden Datenstrukturen als „initiale Algebren“ definiert. Dies funktioniert analog dazu, wie wir in diesem Abschnitt die natürlichen Zahlen als „initiale Orbits“ charakterisiert haben.

I.2 Äquivalenzrelationen und Faktorisierung

Oft erweist es sich als nützlich, eine Menge A in nichtleere Klassen, sogenannte Äquivalenzklassen, aufzuteilen. Dabei soll jedes Element von A in genau einer Klasse liegen. Eine solche Klassenaufteilung heißt *Partition* und wird formal folgendermaßen definiert.

Definition 2.1 *Eine* Partition *einer Menge A ist eine Teilmenge $\mathcal{P} \subseteq \mathcal{P}(A)$, so dass*

(P1) $A = \bigcup \mathcal{P} := \{x \in A \mid \exists P \in \mathcal{P}.\, x \in P\}$

(P2) $\forall P \in \mathcal{P}.\, P \neq \emptyset$

(P3) $\forall P, Q \in \mathcal{P}.\, \forall x \in A.\, x \in P \wedge x \in Q \Rightarrow P = Q.$ $\diamondsuit$

Dabei bedeutet (P1), dass jedes Element mindestens in einer Klasse liegt, wegen (P2) kann keine Klasse leer sein, und nach (P3) können sich verschiedene Klassen nicht überschneiden (es kann also kein Element in verschiedenen Klassen liegen).

Ein andere Sichtweise auf eine solche Klasseneinteilung bekommt man durch eine spezielle Art von Relationen, den sogenannten *Äquivalenzrelationen.*

Definition 2.2 *Eine zweistellige Relation $R \subseteq A \times A$ heißt* Äquivalenzrelation *(oder* equivalence relation *auf englisch), wenn folgende Bedingungen erfüllt sind:*

(ER1) *R ist* reflexiv*, d.h. $\forall x \in A.\, x\ R\ x$*

(ER2) *R ist* symmetrisch*, d.h. $\forall x, y \in A.\, x\ R\ y \Rightarrow y\ R\ x$*

(ER3) *R ist* transitiv*, d.h. $\forall x, y, z \in A.\, x\ R\ y \wedge y\ R\ z \Rightarrow x\ R\ z.$* $\diamondsuit$

Im folgenden Theorem geben wir explizit eine Bijektion zwischen

- $\mathsf{Part}(A)$ = Menge der Partitionen auf A und
- $\mathsf{EqRel}(A)$ = Menge der Äquivalenzrelationen auf A

an.

Theorem 2.3 *Sei A eine beliebige Menge.*

(1) *Sei $\mathcal{P}$ eine Partition von A. Dann gibt es eine eindeutig bestimmte, surjektive Abbildung*

$$\mathrm{quot}_{\mathcal{P}} : A \twoheadrightarrow \mathcal{P}$$

sodass für alle $x \in A$ gilt $x \in \mathrm{quot}_{\mathcal{P}}(x)$.

Die Relation $\ker(\mathrm{quot}_{\mathcal{P}}) \subseteq A \times A$ *mit*

$$x \ker(\mathrm{quot}_{\mathcal{P}})\ y \quad :\Leftrightarrow \quad \mathrm{quot}_{\mathcal{P}}(x) = \mathrm{quot}_{\mathcal{P}}(y)$$

ist eine Äquivalenzrelation auf A, für die wir $\mathrm{er}(\mathcal{P})$ *schreiben* ("equivalence relation induced by $\mathcal{P}$").

(2) *Sei R eine Äquivalenzrelation auf A. Wir definieren die Äquivalenzklasse $[x]_R$ eines Elementes $x \in A$ bezüglich R als*

$$[x]_R := \{y \in A \mid x\ R\ y\}$$

wodurch eine Funktion $[\_]_R : A \to \mathcal{P}(A)$ induziert wird. Die Menge

$$A/R := \{[x]_R \mid x \in A\} \subseteq \mathcal{P}(A)$$

ist dann eine Partition von A.

(3) *Wenn $\mathcal{P}$ eine Partition von A ist, dann gilt $\mathcal{P} = A/\mathrm{er}(\mathcal{P})$.*

(4) *Wenn R eine Äquivalenzrelation auf A ist, dann gilt $R = \mathrm{er}(A/R)$.*

Beweis:

(1) Anstatt zu zeigen, dass $\mathrm{er}(\mathcal{P})$ eine Äquivalenzrelation ist, zeigen wir allgemeiner, das für eine beliebige Funktion $f : A \to B$ ist der *Kern* dieser Funktion, also die Relation $\ker(f) \subseteq A \times A$ mit

$$x \ker(f)\, y \quad :\Leftrightarrow \quad f(x) = f(y)$$

eine Äquivalenzrelation auf A.

(ER1) $x \ker(f)\, x$, weil $f(x) = f(x)$

(ER2) $x \ker(f)\, y \Leftrightarrow f(x) = f(y) \Rightarrow f(y) = f(x) \Leftrightarrow y \ker(f)\, x$

(ER3) $f(x) = f(y) \wedge f(y) = f(z) \Rightarrow f(x) = f(z)$

(2) Sei R eine Äquivalenzrelation auf R. Wir müssen für A/R die Eigenschaften (P1), (P2), (P3) nachweisen.

(P1) für jedes $x \in A$ gilt $x \in [x]_R \in A/R$

(P2) jedes $[x]_R \in A/R$ enthält zumindest x, da $x \in [x]_R$

(P3) wenn $z \in [x]_R$ und $z \in [y]_R$, dann $x\ R\ z$ und $y\ R\ z$, und somit auch $x\ R\ y$, da R symmetrisch und transitiv ist.

(3) Sei $\mathcal{P}$ eine Partition von A. Vorab bemerken wir, dass

$$(\dagger) \quad x \,\mathrm{er}(\mathcal{P})\, y \;\Leftrightarrow\; \mathrm{quot}_{\mathcal{P}}(x) = \mathrm{quot}_{\mathcal{P}}(y) \;\Leftrightarrow\; \exists P{\in}\mathcal{P}.\, x, y \in P$$

wobei die Rückrichtung der zweiten Äquivalenz daraus folgt, dass es höchstens ein $P \in \mathcal{P}$ gibt, in dem sowohl x als auch y als Elemente enthalten sind.

Des weiteren gilt für $x \in P \in \mathcal{P}$, dass $P = [x]_{\mathrm{er}(\mathcal{P})}$, was man leicht wie folgt einsieht. Wenn $y \in P$, dann gilt wegen $(\dagger)$, dass $x \,\mathrm{er}(\mathcal{P})\, y$ und somit

$y \in [x]_{\mathrm{er}(\mathcal{P})}$. Wenn andererseits $y \in [x]_{\mathrm{er}(\mathcal{P})}$, dann $x \,\mathrm{er}(\mathcal{P})\, y$ und somit $y \in \mathrm{quot}_{\mathcal{P}}(y) = \mathrm{quot}_{\mathcal{P}}(x) = P$ wegen (†).

Wir zeigen nun, dass $\mathcal{P} = A/\mathrm{er}(\mathcal{P})$. Angenommen $P \in \mathcal{P}$. Dann gibt es ein $x \in P$ und es gilt $P = [x]_{\mathrm{er}(\mathcal{P})} \in A/\mathrm{er}(\mathcal{P})$. Angenommen $[x]_{\mathrm{er}(\mathcal{P})} \in A/\mathrm{er}(\mathcal{P})$. Dann gilt aber $[x]_{\mathrm{er}(\mathcal{P})} = \mathrm{quot}_{\mathcal{P}}(x) \in \mathcal{P}$.

(4) Sei R eine Äquivalenzrelation auf A. Wir zeigen nun $R = \mathrm{er}(A/R)$.

Angenommen $x \, R \, y$. Dann gilt $[x]_R = [y]_R$ und somit $x \,\mathrm{er}(A/R)\, y$. Also ist R in $\mathrm{er}(A/R)$ als Teilmenge enthalten.

Angenommen $x \,\mathrm{er}(A/R)\, y$. Dann gibt es ein $z \in A$ mit $x, y \in [z]_R$ und somit $z \, R \, x$ und $z \, R \, y$, woraus wegen der Symmetrie und Transitivität von R folgt, dass auch $x \, R \, y$ gilt. Also $\mathrm{er}(A/R)$ in R als Teilmenge enthalten. □

Obwohl Partitionen und Äquivalenzrelationen im wesentlichen auf dasselbe hinauslaufen, werden in der mathematischen Praxis Äquivalenzrelationen bevorzugt, da die Relation $R \subseteq A \times A$ oft leichter zu handhaben ist als die Partition $\mathcal{P} \subseteq \mathcal{P}(A)$.

Dessen ungeachtet möchte man oft auch den Quotienten A/R betrachten. Dies ist zunächst etwas verwirrend, da die Elemente von A/R selbst Teilmengen von A sind. Daher veranschaulichen wir die Beziehungen noch einmal in folgendem Diagramm

$$\begin{tikzcd}
A \arrow[d, "\mathrm{quot}_R"', "surj."] \arrow[dr, "{[\cdot]}"] & \\
A/R \arrow[r, hookrightarrow] & \mathcal{P}(A)
\end{tikzcd}$$

wobei wir quot_R als Abkürzung für $\mathrm{quot}_{A/R}$ schreiben.
Wir wenden uns nun dem Problem zu, wie man Funktionen

$$\bar{f} : A/R \to B$$

durch Angabe einer geeigneten Funktion $f : A \to B$ definiert, für die gilt

$$\begin{tikzcd}
A \arrow[d, "\mathrm{quot}_R"'] \arrow[dr, "f"] & \\
A/R \arrow[r, dashed, "\bar{f}"'] & B
\end{tikzcd}$$

Weil quot_R surjektiv ist, ist $\bar{f}$ durch f eindeutig bestimmt. Die Frage ist nun vielmehr, für welche $f : A \to B$ ein $\bar{f} : A/R \to B$ existiert mit $\bar{f} \circ \mathrm{quot}_R = f$.

Definition 2.4 *Sei R eine Äquivalenzrelation auf A und $f : A \to B$. Wir sagen, dass f* die Relation R respektiert *genau dann, $R \subseteq \ker(f)$, d.h.*

$$x \; R \; y \quad \textit{impliziert} \quad f(x) = f(y)$$

für alle $x, y \in A$. ◊

Dass dies die *eindeutige Fortsetzbarkeit von f längs* quot_R *zu einem $\bar{f} : A/R \to B$ mit $f = \bar{f} \circ$* quot_R charakterisiert, wird in folgendem Satz bewiesen.

Theorem 2.5 *Sei R eine Äquivalenzrelation auf A. Eine Funktion $f : A \to B$ respektiert R genau dann, wenn ein $\bar{f} : A/R \to B$ existiert mit $f = \bar{f} \circ \mathrm{quot}_R$.*

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "\mathrm{quot}_R"'] & B \\
A/R \arrow[ur, dashed, "\bar{f}"'] &
\end{tikzcd}$$

Beweis:

„⇒“ Sei $f : A \to B$ eine Funktion, die R respektiert. Falls ein $\bar{f} : A/R \to B$ existiert mit $f = \bar{f} \circ \mathrm{quot}_R$, so ist $\bar{f}$ eindeutig bestimmt (mit dieser Eigenschaft), weil quot_R nach Konstruktion surjektiv ist. Es verbleibt also die Existenz von $\bar{f}$ zu zeigen. Wenn $\bar{f}$ existiert, so muss gelten:

$$(*) \quad \bar{f}([x]) = f(x)$$

d.h. $\mathrm{Graph}(\bar{f}) = \{\langle [x]_R \, , \, f(x) \rangle \mid x \in A\}$. Diese Relation ist total und sie ist rechtseindeutig, weil

$$[x]_R = [y]_R \Rightarrow xRy \Rightarrow f(x) = f(y)$$

wobei die letzte Implikation aus der Annahme folgt, dass f die Äquivalenzrelation R respektiert.

„⇐“ Falls $f = \bar{f} \circ \mathrm{quot}_R$, dann respektiert f die Äquivalenzrelation R, weil

$$\begin{aligned}
x \; R \; y \quad &\Leftrightarrow \quad \mathrm{quot}_R(x) = \mathrm{quot}_R(y) \Rightarrow \bar{f}(\mathrm{quot}_R(x)) = \bar{f}(\mathrm{quot}_R(y)) \\
&\Leftrightarrow \quad f(x) = f(y)
\end{aligned}$$

wobei die letzte Implikation aus der Annahme $f = \bar{f} \circ \mathrm{quot}_R$ folgt. □

Bemerkung. Statt „f respektiert R“ sagt man oft auch „$f([x]_R) = f(x)$ ist *wohldefiniert*“.

Abschließend diskutieren wir, wie man eine Funktion $f : A \to B$ zerlegen kann als $f = m \circ e$

$$\begin{tikzcd}
A \arrow[dd, "f"'] \arrow[dr, "e"] & \\
& f[A] \arrow[dl, hook', "m"] \\
B &
\end{tikzcd}$$

wobei $f[A] = \{f(x) \mid x \in A\}$, $m(y) = y$ und $e(x) = f(x)$, und wie dies mit Faktorisierung von A modulo einer Äquivalenzrelation $\ker(f)$ zusammenhängt.

Theorem 2.6 *Sei $f : A \to B$ eine Funktion. Dann ist*

$$\ker(f) = \{\, \langle x, y\rangle \in A \times A \mid f(x) = f(y) \,\}$$

eine Äquivalenzrelation auf A.

Die Funktion f respektiert die Äquivalenzrelation $\ker(f)$*, und somit existiert eine eindeutige Funktion $\bar{f} : A/\ker(f) \to B$ mit $f(x) = \bar{f}([x]_{\ker(f)})$ für alle $x \in A$.*

Sei ferner $I := f[A] = \{f(x) \mid x \in A\}$ und $e : A \to I : x \mapsto f(x)$. Die Funktion e respektiert ebenfalls $\ker(f)$*. und somit gibt es eine eindeutig bestimmte Funktion $i : A/\ker(f) \to f[A]$ mit $i([x]_{\ker(f)}) = e(x)$. Dieses i ist sogar eine Bijektion, und es gilt*

$$\begin{tikzcd}
& f[A] \arrow[dr, hook, "m"] & \\
A \arrow[ur, "e"] \arrow[dr, "\mathrm{quot}_{\ker(f)}"'] & & B \\
& A/\ker(f) \arrow[uu, "i"', "\cong"] \arrow[ur, "inj.", "\bar{f}"'] &
\end{tikzcd}$$

Beweis: Es gilt aufgrund der Konstruktion der beteiligten Funktionen offensichtlich auch, dass $m \circ i = \bar{f}$. Es bleibt also zu zeigen, dass i bijektiv ist.

Wenn $i([x]_{\ker(f)}) = i([y]_{\ker(f)})$, dann gilt $e(x) = e(y)$ und somit auch $f(x) = f(y)$. Das bedeutet $x \ker(f) y$ und damit auch $[x]_{\ker(f)} = [y]_{\ker(f)}$. Also ist i injektiv.

Wenn $y \in f[A]$, dann gibt es ein $x \in A$ mit $y = f(x)$ und somit

$$y = f(x) = (i \circ \mathrm{quot}_{\ker(f)})(x) = i([x]_{\ker(f)})$$

Also ist i auch surjektiv.

Weil i injektiv und surjektiv ist, ist i auch bijektiv. □

Man kann also $f[A]$, das Bild von A unter f, als Quotienten von A auffassen, nämlich als $A/\ker(f)$.

I.3 Monoide, Automaten und Gruppen

Wir wenden uns in diesem Abschnitt bekannteren algebraischen Strukturen wie Monoiden und Gruppen zu. Dabei werden wir sehen, dass Automaten als (spezielle) Monoid-Homomorphismen aufgefasst werden können.

Definition 3.1 *Ein Monoid ist ein Tripel (M, m, e), so dass M eine Menge ist und $m : M \times M \to M$ sowie $e \in M$ die folgenden Bedingungen erfüllen:*

(Ass) $\quad m(m(x, y), z) = m(x, m(y, z))$ *für alle* $x, y, z \in M$

(Neutr) $\quad m(e, x) = x = m(x, e)$ *für alle* $x \in M$. $\Diamond$

Notation: Wir schreiben oft statt $m(x, y)$ multiplikativ $x \cdot y$ oder sogar xy, wenn m aus dem Kontext klar ist.

Bemerkung: Man beachte, dass e durch M und m bereits eindeutig bestimmt ist: wenn (M, m, e) und (M, m, e') Monoide sind, so gilt $e = e'$.

Beispiel:

(1) $(\{e\}, m, e)$ ist das triviales Monoid.

(2) Sei A eine Menge (Alphabet), A^* die Menge aller endlichen Sequenzen von Elementen aus A, und ε die leere Sequenz. Die Verknüpfung

$$m(\overbrace{s_1 \dots s_n}^{s=}, \overbrace{s'_1 \dots s'_m}^{s'=}) = s_1 \dots s_n\, s'_1 \dots s'_m = ss'$$

ist die Juxtaposition von Sequenzen (also das „Nebeneinanderschreiben“). Dadurch wird (A^*, m, ε) zu einem Monoid.

(3) $(\mathbb{N}, +, 0)$, $(\mathbb{R}, +, 0)$ etc. sind kommutative Monoide (d.h. es gilt zusätzlich das Gesetz $x + y = y + x$).

Wir werden als nächstes zeigen, dass jedes Monoid sich als Untermonoid von Monoiden bestimmter Form identifizieren lässt.

Dazu bedarf es einiger vorbereitender Bemerkungen bezüglich des Funktionsbegriffes.

Definition 3.2 *Eine Funktion ist ein Tripel $f = (A, B, F)$, wobei A, B Mengen sind und $F \subseteq A \times B$ eine Relation ist, die folgende Bedingungen erfüllt:*

$(x\ F\ y \wedge x\ F\ y') \Rightarrow y = y'$ *(Eindeutigkeit von F)*

$\forall x \in A.\, \exists y \in B.\, x\ F\ y$ *(Totalität von F)*

Dabei heißt A die Definitionsmenge *von f (domain of f), B heißt die* Wertemenge *von f (codmain of f), und F heißt der* Graph *von f.*

Wir schreiben $f : A \to B$ für die Aussage, dass ein F existiert, so dass $f = (A, B, F)$ eine Funktion ist. Wir definieren

$$B^A = \{f \mid f : A \to B\}$$

Sind $f = (A, B, F)$ und $g = (B, C, G)$ Funktionen, dann sei die Komposition *von f und g, die wir mit $g \circ f$ bezeichnen, gegeben durch (A, C, R), wobei*

$$x \; R \; z :\Leftrightarrow \exists y \in B.\, x \; F \; y \wedge y \; G \; z$$

Die Funktion $\mathrm{id}_A := (A, A, \{(x, x) \mid x \in A\})$ wird als identische Funktion auf A bezeichnet. ◊

Lemma 3.3 *Für jede Menge A ist $(A^A, \circ, \mathrm{id}_A)$ ein Monoid.*

Beweis: $\circ$ ist assoziativ, wie man sich durch Nachrechnen überzeugen kann. □

Wir werden zeigen, dass sich jedes Monoid (M, m, e) in das Monoid $(M^M, \circ, \mathrm{id}_M)$ „einbetten" lässt, was in der Literatur als *Cayley'scher Repräsentationssatz* bekannt ist. Dazu bedarf es aber einiger vorbereitender Begriffsbildungen.

Definition 3.4 *Seien (M_1, m_1, e_1), (M_2, m_2, e_2) Monoide. Ein Monoid-Homomorphismus von (M_1, m_1, e_1) nach (M_2, m_2, e_2) ist eine Abbildung $h : M_1 \to M_2$, so dass*

(i) $h(e_1) = e_2$ *und*

(ii) $h(m_1(x, y)) = m_2(h(x), h(y))$ *für alle $x, y, \in M_1$*

Wir schreiben dafür abkürzend $h : (M_1, m_1, e_1) \to (M_2, m_2, e_2)$.

Wir nennen einen solchen Homomorphismus h eine Einbettung *genau dann, wenn h injektiv ist.* ◊

Wenn $h(x) = x$ für $x \in M_1$ so ist die Monoidstruktur auf M_1 bereits durch die Bedingungen (i) und (ii) eindeutig bestimmt. Dies motiviert folgende Definition des Begriffes *Untermonoid*.

Definition 3.5 *Ein* Untermonoid *eines Monoids (M, m, e) ist gegeben durch eine Teilmenge P von M, so dass*

(i) $e \in P$ *und*

(ii) *aus $x, y \in P$ immer $m(x, y) \in P$ folgt.* ◊

Nun können wir den Cayley'schen Repräsentationssatz für Monoide beweisen.

Theorem 3.6 *Sei (M, m, e) ein Monoid. Wir definieren $h : M \to M^M$ als $h(x)(y) := m(x, y)$, d.h. $h(x)$ ist die „Linksmultiplikation" mit x. Dann ist der Homomorphismus $h : (M, m, e) \to (M^M, \circ, \mathrm{id}_M)$ eine Einbettung.*

Beweis: Die Abbildung h ist injektiv, weil aus $h(x) = h(x')$ folgt, dass

$$x = m(x, e) = h(x)(e) = h(x')(e) = m(x,' e) = x'$$

Die Abbildung h ist ein Homomorphismus, weil

- $h(e) = \mathrm{id}_M$, da $h(e)(x) = m(e, x) = x = \mathrm{id}_M(x)$
- $h(m(x, y)) = h(x) \circ h(y)$, da

$$\begin{aligned} & h(m(x,y))(z) = m(m(x,y),z) = m(x,m(y,z)) = m(x,h(y)(z)) \\ = \quad & h(x)(h(y)(z)) = (h(x) \circ h(y))(z) \end{aligned}$$

für all $x, y, z \in M$. $\square$

Die intuitive Bedeutung des obigen Repräsentationssatzes ist, dass man ein $x \in M$ mit der von ihm induzierten Funktion $h(x) : y \mapsto x \cdot y$ identifizieren kann. Durch Anwendung dieser Funktion $h(x)$ auf das neutrale Element e kann man das x wieder „zurückrechnen", da $h(x)(e) = x \cdot e = x$. Eine konkrete Manifestation dieses Sachverhaltes ist, dass man einen Vektor — etwa im $\mathbb{R}^3$ — mit der Translation längs dieses Vektors identifizieren kann.

Als nächstes wenden wir uns der Frage zu, was „vernünftige" Äquivalenzrelationen auf der unterliegenden Menge M eines Monoids (M, m, e) sind. „Vernünftig" soll heißen, dass die Quotientenabbildung $\mathrm{quot}_R : M \to M/R$ auch ein Monoid-Homomorphismus sein soll und zwar bezüglich einer geeigneten Monoidstruktur auf der Menge M/R, die, wie wir gleich sehen werden, durch diese Anforderung eindeutig bestimmt ist.
Sei (M, m, e) ein Monoid und R eine Äquivalenzrelation auf M. Wir schreiben h_R für quot_R. Wenn nun $(M/R, m_R, e_R)$ ein Monoid ist mit

$$h_R : (M, m, e) \to (M/R, m_R, e_R)$$

dann gilt

(1) $e_R = h_R(e) = [e]_R$ und

(2) $m_R([x]_R, [y]_R) = m_R(h_R(x), h_R(y)) = h_R(m(x, y)) = [m(x, y)]_R$

Damit sind e_R und m_R eindeutig festgelegt; letzteres, weil jedes Element aus M/R von der Form $[x]_R$ für ein geeignetes $x \in M$ ist. Im allgemeinen wird allerdings m_R durch (2) nicht wohldefiniert sein. Dies ist aber genau dann der Fall, wenn für alle $x_1, x_2, y_1, y_2 \in m$ gilt, dass

$$[x_1]_R = [x_2]_R \wedge [y_1]_R = [y_2]_R \Rightarrow [m(x_1, y_1)]_R = [m(x_2, y_2)]_R$$

d.h. $x_1 \, R \, x_2 \wedge y_1 \, R \, y_2 \Rightarrow m(x_1, y_1) \, R \, m(x_2, y_2)$ für alle $x_1, x_2, y_1, y_2 \in M$.

Wir zeigen nun, dass diese notwendige Bedingung tatsächlich auch hinreichend ist.

Definition 3.7 *Eine* Kongruenzrelation *auf einem Monoid* (M, m, e) *ist eine Äquivalenzrelation* R *auf* M*, so dass* R *mit* m *„verträglich“ ist, d.h. für alle* $x_1, x_2, y_1, y_2 \in M$*:*

$$x_1 \, R \, x_2 \wedge y_1 \, R \, y_2 \Rightarrow m(x_1, y_1) \, R \, m(x_2, y_2) \qquad \diamond$$

Man wäre vielleicht auch versucht zu fordern, dass R mit der nullstelligen Operation e verträglich sein soll, also $e \, R \, e$. Dies ist aber nicht nötig, weil R ohnehin reflexiv ist.

Theorem 3.8 *Seien* $\mathcal{M} = (M, m, e)$ *ein Monoid und* R *eine Kongruenzrelation auf* $\mathcal{M}$*. Dann ist* $\mathcal{M}/R = (M/R, m_R, e_R)$ *ein Monoid, wobei*

$$e_R = [e]_R \qquad \textit{und} \qquad m_R([x]_R, [y]_R) = [m(x, y)]_R$$

und $h_R := \mathrm{quot}_R : \mathcal{M} \to \mathcal{M}/R$ *ein surjektiver Monoid–Homomorphismus.*

Beweis: Wir zeigen zuerst, dass m_R wohldefiniert ist: Sei $[x_1]_R = [x_2]_R$ und $[y_1]_R = [y_2]_R$, dann gilt $x_1 \; R \; x_2$ und $y_1 \; R \; y_2$; weil R eine Kongruenzrelation ist, gilt dann auch $m(x_1, y_1) \; R \; m(x_2, y_2)$, d.h. $[m(x_1, y_1)]_R = [m(x_2, y_2)]_R$.

Den Beweis, dass $\mathcal{M}/R$ auch tatsächlich ein Monoid ist (nämlich das Faktormonoid von M bezüglich R), und dassquot_R ein Monoid–Homomorphismus ist, überlassen wir dem Leser als Übung(!). $\square$

Umgekehrt, wenn $h : \mathcal{M} \to \mathcal{M}'$ ein surjektiver Monoid–Homomorphismus ist, so ist $R := \ker(h)$ eine Kongruenzrelation auf $\mathcal{M}$. Die aufgrund von Theorem 2.5 eindeutig bestimmte Abbildung $i : M/R \to M'$ mit $i \circ h_R = h$ ist ein bijektiver Monoid-Homomorphismus von $\mathcal{M}/R$ nach $\mathcal{M}'$ wie in folgendem Diagramm veranschaulicht.

$$\begin{tikzcd}
 & \mathcal{M}' \\
\mathcal{M} \arrow[ur, "h"] \arrow[dr, "h_R"'] & \\
 & \mathcal{M}/R \arrow[uu, dashed, "i", "bij."']
\end{tikzcd}$$

Dass i ein Monoid–Homomorphismus ist, sieht man folgendermassen: es gilt $i([e]) = e'$ und $i(m_{/R}([x],[y]) = i([m(x,y)]) = h(m(x,y)) = m'(h(x),h(y)) = m'(i([x]),i([y]))$. Dass i bijektiv sieht man folgendermassen: i ist surjektiv, weil h nach Annahme surjektiv ist und $i \circ \mathrm{quot}_R = h$, und i ist injektiv, weil aus $i([x]) = i([y])$ folgt, dass $h(x) = h(y)$ und somit $x \ker(h) y$, also $[x] = [y]$.

Wenn aber i ein bijektiver Monoid-Homomorphismus ist, so ist auch i^{-1} ein Monoid-Homomorphismus, wie wir sogleich sehen werden, d.h. i ist ein Monoid-homomorphismus im Sinne folgender Definition.

Definition 3.9 *Ein* Monoid-Isomorphismus *ist ein Monoid-Homomorphismus $h : \mathcal{M}_1 \to \mathcal{M}_2$, so dass ein Monoid–Homomorphismus $k : \mathcal{M}_2 \to \mathcal{M}_1$ existiert mit $k \circ h = \mathrm{id}_{M_1}$ und $h \circ k = \mathrm{id}_{M_2}$.* $\diamondsuit$

Offensichtlich ist ein Monoid-Isomorphismus bijektiv. Es gilt aber auch die Umkehrung.

Lemma 3.10 *Ein Monoid-Homomorphismus $h : \mathcal{M}_1 \to \mathcal{M}_2$ ist ein Monoid-Isomorphismus genau dann, wenn h bijektiv ist.*

Beweis: Die schwierige Richtung ist von rechts nach links. Seien

$$\mathcal{M}_1 = (M_1, m_1, e_1) \qquad \text{und} \qquad \mathcal{M}_2 = (M_2, m_2, e_2)$$

zwei Monoide, und sei $h : \mathcal{M}_1 \to \mathcal{M}_2$ ein bijektiver Homomorphismus. Dann hat h eine Umkehrabbildung $k := h^{-1}$. Nun gilt $k(e_2) = e_1$, weil $h(e_1) = e_2$. Sind ferner $u, v \in M_2$, dann gibt es $x, y \in M_1$ mit $u = h(x)$ und $v = h(y)$, und somit gilt

$$k(m_2(u,v)) = k(m_2(h(x),h(y)) = k(h(m_1(x,y))) = m_1(x,y) = m_1(k(u),k(v))$$

weil $x = k(u)$ und $y = k(v)$. Also sind die Homomorphiebedingungen erfüllt. $\square$

Wir betrachten nun eine bemerkenswerte Eigenschaft des Wortmonoids A^* aus Beispiel (2).

Theorem 3.11 *Sei A eine beliebige Menge, $i_A : A \hookrightarrow A^*$ diejenige Funktion, die Elemente $a \in A$ auf die einelementige Sequenz $a \in A^*$ abbildet, und es sei $\mathcal{M}$ ein beliebiges Monoid. Dann gibt es für jede Funktion $f : A \to M$ genau einen Monoid-Homomorphismus $h : A^* \to \mathcal{M}$ mit $h \circ i_A = f$.*

$$\begin{tikzcd}
A \arrow[r, hook, "i_A"] \arrow[dr, "f"'] & A^* \arrow[d, "h"] \\
& M
\end{tikzcd}$$

Diese eindeutig bestimmte Abbildung h nennt man auch die Fortsetzung *von f zu einem Monoid–Homomorphismus auf A^*.*

Beweis:

Eindeutigkeit: Seien $h_1, h_2 : A^* \to \mathcal{M}$ Monoid-Homomorphismen mit $h_1 \circ i_A = h_2 \circ i_A$. Zu zeigen ist $h_1 = h_2$. Wir beweisen $h_1(s) = h_2(s)$ durch Induktion über die Länge der Sequenz $n = |s|$ in A^*:

- $n = 0$: Dann $s = \varepsilon$ und es gilt $h_1(\varepsilon) = e = h_2(\varepsilon)$.
- $n \to n+1$: Sei s eine Sequenz der Länge $n+1$, dann $s = a \cdot s'$ für ein $a \in A$ und ein $s' \in A^*$ mit Länge n; dann gilt

  $$h_1(s) = h_1(a \cdot s') = h_1(a) \circ h_1(s') \overset{(\dagger)}{=} h_2(a) \circ h_2(s') = h_2(a \cdot s') = h_2(s)$$

  Wobei der Schritt ($\dagger$) wegen

  $$h_1(a) = h_1(i_A(a)) = h_2(i_A(a)) = h_2(a)$$

  und aufgrund der Induktionshypothese $h_1(s') = h_2(s')$ gilt.

Existenz: Die Existenz eines solchen Monoid-Homomorphismus ergibt sich durch die Setzung

$$h(a_1 \dots a_n) := h(a_1) \cdot \ldots \cdot h(a_n)$$

wobei $\cdot$ die 2–stellige Operation des Monids $\mathcal{M}$ (in Infixnotation) bezeichnet. Da diese assoziativ ist, macht die rechte Seite der Gleichung auch ungeklammert Sinn. Für $n = 0$ setzen wir für die rechte Seite $h(a_1)\cdot \ldots \cdot h(a_n) = e$.

Der Nachweis, dass h ein Homorphismus ist, verbleibt als Übung. $\square$

Für den Fall, dass $\mathcal{M} = (S^S, \circ, \mathrm{id}_s)$ erhalten wir eine 1–1–Korrespondenz zwischen Familien $f : A \to S^S$ und Homomorphismen von A^* nach $(S^S, \circ, \mathrm{id}_s)$. Eine Funktion $f : A \to S^S$ lässt sich in folgendem Sinn als Automat verstehen:

- S ist die Menge der internen Zustände des Automaten
- A ist das „Eingabealphabet“ des Automaten (welche man sich intuitiv als „Knöpfe“ oder „Tasten“ außen auf dem Automaten vorstellen kann)
- f gibt für jede „Taste“ $a \in A$ an, welche Zustandsüberführung $f(a) : S \to S$ das Drücken von a bewirkt.

Die Erweiterung von $f : A \to S^S$ zu einem Homomorphismus $h : A^* \to S^S$ beschreibt dann, welche Zustandsüberführung durch eine Sequenz von Tastendrücken bewirkt wird.
Unter diesem Gesichtspunkt liegt es nahe, den Automatenbegriff dahingehend zu erweitern, dass man statt A^* ein beliebiges Monoid $\mathcal{M}$ zugrundelegt.

Definition 3.12 *Sei $\mathcal{M}$ ein Monoid. Ein* $\mathcal{M}$-Automat ist dann ein Monoid-Homomorphismus

$$h : \mathcal{M} \to (S^S, \circ, \mathrm{id}_s)$$

für eine Zustandsmenge S. $\Diamond$

Bemerkung. Wenn man für $\mathcal{M}$ setzt $(\mathbb{N}, +, 0)$, dann ist $h : \mathcal{M} \to S^S$ ein *diskretes System*; wenn man für $\mathcal{M}$ setzt $(\mathbb{R}_0^+, +, 0)$, dann ist $h : \mathcal{M} \to S^S$ ein *kontinuierliches System*.

Wir schließen unsere Betrachtungen über allgemeine Monoide mit folgender Beobachtung ab.

Theorem 3.13 *Sei $\mathcal{M} = (M, m, e)$ ein Monoid. Dann gibt es einen eindeutig bestimmten Monoid-Homomorphismus $h : M^* \to (M, m, e)$ mit $h(x) = x$ für alle $x \in M$. Außerdem ist h surjektiv, und es gilt $\mathcal{M} \cong M^*/\ker(h)$.*

Beweis: Der gesuchte Homomorphismus existiert auf Grund von Theorem 3.11. Der gewünschte Isomorphismus ergibt sich folgendermaßen

$$\begin{tikzcd} M^* \arrow[r, "h"] \arrow[dr, "\mathrm{quot}_{\ker(h)}"'] & \mathcal{M} \\ & M^*/\ker(h) \arrow[u, "\cong", "i"'] \end{tikzcd}$$

Die Surjektivität ist wegen $h(x) = x$ offensichtlich, und aus ihr folgt die Isomorphie zum Faktormonoid. $\square$

Somit ist jedes Monoid isomorph zu einem Quotienten eines *freien Monoids*, d.h. eines Monoids der Gestalt A^* für eine geeignete Menge A.

Wir studieren nun eine spezielle, besonders wichtige Klasse von Monoiden, die so genannten Gruppen.

Definition 3.14 *Eine Gruppe ist ein Monoid $(G, \cdot, e)$, in der für jedes $x \in G$ ein Element $y \in G$ mit $x \cdot y = e$ existiert.* $\Diamond$

Lemma 3.15 *Für eine Gruppe $G = (G, \cdot, e)$ gelten folgende Aussagen.*

(1) *Aus $x \cdot y = e$ folgt $y \cdot x = e$.*

(2) *Aus $x \cdot y = x \cdot z$ folgt $y = z$.*

Also gibt es für jedes $x \in G$ ein eindeutig bestimmtes Element $x^{-1} \in G$ mit

$$x \cdot x^{-1} = e = x^{-1} \cdot x$$

welches wir das inverse Element *von x nennen.*

Beweis: Wir zeigen zuerst (1). Sei $x \cdot y = e$. Dann gibt es ein $z \in G$ mit $y \cdot z = e$. Also gilt

$$x = x \cdot e = x \cdot y \cdot z = e \cdot z = z$$

und somit $y \cdot x = y \cdot z = e$.

Um (2) zu zeigen, nehmen wir an, dass $x \cdot y = x \cdot z$. Wegen (1) gibt es ein $u \in G$ mit $u \cdot x = e$, woraus

$$y = e \cdot y = u \cdot x \cdot y = u \cdot x \cdot z = e \cdot z = z$$

folgt aufgrund der Annahme $x \cdot y = x \cdot z$. □

Für jedes Monoid $M = (M, \cdot, e)$ bilden die invertierbaren Elemente von M eine Gruppe. (Beweis als Übung!)

Sei $\mathrm{Bij}(S)$ die Gruppe der invertierbaren Elemente des Monoids $(S^S, \circ, \mathrm{id}_s)$, die gerade aus den bijektiven Funktionen von S nach S besteht. Dann gibt es für jede Gruppe $G = (G, \cdot, e)$ einen injektiven Homomorphismus

$$h_G : G \to \mathrm{Bij}(G) : x \mapsto (y \mapsto x \cdot y)$$

also

$$\begin{tikzcd} G \arrow[r, "h_G"] \arrow[dr, "h"'] & \mathrm{Bij}(G) \arrow[d, hook] \\ & G^G \end{tikzcd}$$

wobei h der Monoid-Homomorphismus aus Theorem 3.6 ist. Dieser Repräsentationssatz für Gruppen ist in der Literatur als Cayley'scher Repräsentationssatz bekannt.

Lemma 3.16 *Seien G,H Gruppen und $h : G \to H$ ein Monoid-Homomorphismus. Dann gilt $h(x^{-1}) = h(x)^{-1}$ für alle $x \in G$. Somit ist das Bild* $\mathrm{im}(h) := h[G]$ *eine Untergruppe von H. Weiters ist auch*

$$\mathrm{Ker}(h) := h^{-1}(\{e\}) = \{x \in G \mid h(x) = e\}$$

eine Untergruppe von G.

Beweis: Es gilt $e = h(e) = h(x \cdot x^{-1}) = h(x) \cdot h(x^{-1})$ und somit $h(x)^{-1} = h(x^{-1})$. Also ist $\mathrm{im}(h)$ nicht nur unter der zweistelligen Operation, sondern auch unter Inversenbildung abgeschlossen. Weil $\mathrm{im}(h)$ auch e enthält, ist $\mathrm{im}(h)$ also eine Untergruppe von H.

$\mathrm{Ker}(h)$ ist eine Untergruppe von G, weil

- das neutrale Element $e \in \mathrm{Ker}(h)$ ist, da $h(e) = e$;
- sie unter der Verknüpfung abgeschlossen ist: Sind $x, y \in \mathrm{Ker}(h)$, dann gilt $h(x) = e = h(y)$ und deshalb $e = e \cdot e = h(x) \cdot h(y) = h(x \cdot y)$ woraus $x \cdot y \in \mathrm{Ker}(h)$ folgt;
- sie unter Inversenbildung abgeschlossen ist: Ist $x \in \mathrm{Ker}(h)$ dann $h(x) = e$ und deshalb $h(x^{-1}) = h(x)^{-1} = e^{-1} = e$, weswegen $x^{-1} \in \mathrm{Ker}(h)$. □

Da Monoid-Homomorphismen zwischen Gruppen auch Inversenbildung erhalten, heißen sie auch Gruppen-Homomorphismen.

Wir studieren nun Kongruenzrelationen auf Gruppen und zeigen, dass sie durch die Äquivalenzklassen von e eindeutig bestimmt sind.

Lemma 3.17 *Sei R eine Monoid-Kongruenz auf einer Gruppe G. Dann gilt*

(1) $x \, R \, y \;\Leftrightarrow\; xy^{-1} \, R \, e$

(2) $x \, R \, y \;\Rightarrow\; x^{-1} \, R \, y^{-1}$

(3) $[e]_R$ *ist eine Untergruppe von* G

(4) $e \, R \, x \;\Leftrightarrow\; e \, R \, yxy^{-1}$.

Beweis:

(1) $x \, R \, y \Rightarrow xy^{-1} \, R \, yy^{-1} \Leftrightarrow xy^{-1} \, R \, e$ und $xy^{-1} Re \Rightarrow xy^{-1}y \, R \, ey \Leftrightarrow xRy$

(2) $x \, R \, y \Rightarrow xy^{-1} \, R \, e \Rightarrow x^{-1}xy^{-1} \, R \, x^{-1} \Leftrightarrow y^{-1} \, R \, x^{-1} \Leftrightarrow x^{-1} \, R \, y^{-1}$

(3)
- Wegen $e \, R \, e$, gilt $e \in [e]_R$;
- Sind $x, y \in [e]_R$, dann $e \, R \, x$ und $e \, R \, y$. Daraus folgt $e = ee \, R \, xy$, also $xy \in [e]_R$;
- Ist $x \in [e]_R$, dann $e \, R \, x$ und somit $e = e^{-1} \, R \, x^{-1}$, weshalb $x^{-1} \in [e]_R$.

(4) $e \, R \, x \Rightarrow yey^{-1} \, R \, yxy^{-1} \Rightarrow e \, R \, yxy^{-1} \Rightarrow y^{-1}ey \, R \, y^{-1}yxy^{-1}y \Rightarrow e \, R \, x$ □

Wir zeigen nun, dass es eine 1–1–Korrespondenz zwischen Kongruenzen auf G und bestimmten Untergruppen von G gibt.

Definition 3.18 *Eine Untergruppe H einer Gruppe G heißt* Normalteiler (von G) *genau dann, wenn $yHy^{-1} = H$ für alle $y \in G$.* ◊

Es sei bemerkt, dass eine Untergruppe H von G bereits dann ein Normalteiler von G ist, wenn $yHy^{-1} \subseteq H$ für alle $y \in G$, da daraus für beliebige $y \in G$ folgt, dass $H = yy^{-1}Hyy^{-1} \subseteq yHy^{-1}$, weil ja $y^{-1}Hy \subseteq H$. Da die Bedingung $yHy^{-1} \subseteq H$ logisch äquivalent ist zu $\forall x{\in}H.\, yxy^{-1} \in H$, ist aufgrund von Lemma 3.17 für jede Kongruenz R auf G die Menge $[e]_R$ ein Normalteiler von G.

Lemma 3.19 *Sei H ein Normalteiler von G. Dann ist die Relation $R_H \subseteq G \times G$ mit*

$$x \; R_H \; y \; :\Leftrightarrow \; xy^{-1} \in H$$

eine Kongruenz auf G.

Beweis: Angenommen $x_1 \; R_H \; x_2$ und $y_1 \; R_H \; y_2$, d.h. $x_1x_2^{-1}$, $y_1y_2^{-1} \in H$. Dann gilt

$$x_1y_1(x_2y_2)^{-1} = x_1y_1y_2^{-1}x_2^{-1} = x_1y_1y_2^{-1}x_1^{-1}x_1x_2^{-1} \in H$$

weil $x_1y_1y_2^{-1}x_1^{-1} \in H$, da $y_1y_2^{-1} \in H$ und H ein Normalteiler ist, und $x_1x_2^{-1} \in H$. Also gilt $x_1y_1 \; R_H \; x_2y_2$. □

Wegen Lemma 3.17 (4) ist für jede Kongruenz R auf G die Äquivalenzklasse $[e]_R$ ein Normalteiler von G. Damit können wir die oben angekündigte Korrespondenz zeigen.

Theorem 3.20 *Sei G eine Gruppe. Dann gilt*

(1) $[e]_{R_H} = H$ *für alle Normalteiler H von G;*

(2) $R_{[e]_R} = R$ *für alle Kongruenzen R auf G.*

Beweis:

(1) $x \in [e]_{R_H} \Leftrightarrow x \; R_H \; e \Leftrightarrow xe^{-1} \in H \Leftrightarrow x \in H$

(2) $x \; R_{[e]_R} \; y \Leftrightarrow xy^{-1} \in [e]_R \Leftrightarrow xy^{-1} \; R \; e \Leftrightarrow x \; R \; y$ □

Bemerkung:

1. Wenn G eine *abelsche* bzw. *kommutative* Gruppe ist, d.h. $x \cdot y = y \cdot x$ für alle $x, y \in G$, dann ist *jede* Untergruppe von G ein Normalteiler von G, da $yxy^{-1} = yy^{-1}x = ex = x$.

2. Für eine Kongruenz R auf G gilt $[e]_R = \mathrm{Ker}(\mathrm{quot}_R)$ und $R = \ker(\mathrm{quot}_R)$. Da $R = R_{[e]_R}$, gilt also

   $$x \ker(\mathrm{quot}_R) y \iff xy^{-1} \in \mathrm{Ker}(\mathrm{quot}_R)$$

   was den Zusammenhamg zwischen ker und Ker klarstellt!

I.4 Induktive Definitionen

In diesem Abschnitt studieren wir bestimmte implizite Definitionen von Teilmengen einer Menge M mit Hilfe so genannter *Regelsysteme*.

Definition 4.1 *Ein* Regelsystem *über einer Menge M ist eine Menge von Regeln $\mathcal{R} \subseteq \mathcal{P}(M) \times M$. Eine* $\mathcal{R}$-Regel *ist ein Paar $(H, a) \in \mathcal{R}$, wobei H* Prämissenmenge *und a* Konklusion *der Regel (H, a) genannt wird.*

Für ein Regelsystem $\mathcal{R} \subseteq \mathcal{P}(M) \times M$ sei $\Phi_{\mathcal{R}} : \mathcal{P}(M) \to \mathcal{P}(M)$ wie folgt definiert

$$a \in \Phi_R(A) \;:\Leftrightarrow\; \exists H \subseteq A.\ (H, a) \in \mathcal{R} \;\; . \qquad \Diamond$$

Intuitiv lässt sich eine Regel (H, a) folgendermaßen als „Erzeugungsregel“ verstehen: wenn alle Elemente aus H bereits erzeugt wurden (Voraussetzung), dann kann auch a erzeugt werden (Konklusion). Die Funktion $\Phi_R(A)$ besteht dann aus allen Elemente, die mithilfe einer Regel aus $\mathcal{R}$ aus den Elementen von A erzeugt werden können.

Eine Menge $A \subseteq M$ ist abgeschlossen unter der Regelmenge $\mathcal{R}$ genau dann, wenn für jede Regel $(H, a) \in \mathcal{R}$ gilt, dass $a \in A$, falls $H \subseteq A$, d.h. durch Anwendung von Erzeugungsregeln aus R nichts neues dazukommt.

Definition 4.2 *Sei $\mathcal{R} \subseteq \mathcal{P}(M) \times M$ ein Regelsystem. Eine Teilmenge A von M heißt* $\mathcal{R}$-abgeschlossen *genau dann, wenn*

$$\forall (H, a) \in \mathcal{R}.\ H \subseteq A \Rightarrow a \in A \;\; . \qquad \Diamond$$

Folgendes Lemma ist im weiteren nützlich.

Lemma 4.3 *Sei $\mathcal{R} \subseteq \mathcal{P}(M) \times M$. Dann gilt*

1) $\Phi_{\mathcal{R}}$ *ist* monoton *d.h. für alle $A, B \in \mathcal{P}(M)$ gilt*

$$A \subseteq B \Rightarrow \Phi_{\mathcal{R}}(A) \subseteq \Phi_{\mathcal{R}}(B)$$

2) *A ist genau dann $\mathcal{R}$-abgeschlossen, wenn $\Phi_{\mathcal{R}}(A) \subseteq A$.*

Beweis:

1) Sei $A \subseteq B \subseteq M$. Wenn $a \in \Phi_{\mathcal{R}}(A)$, dann gibt es eine Regel $(H, a) \in \mathcal{R}$ mit $H \subseteq A$. Dann gilt aber auch $H \subseteq B$, da $A \subseteq B$, und somit auch $a \in \Phi_{\mathcal{R}}(B)$.

2) $\Phi_R(A) \subseteq A$ ist äquivalent zu

$$\forall a \in M.\ (\exists (H, a) \in \mathcal{R}.\ H \subseteq A) \Rightarrow a \in A$$

was logisch äquivalent ist zu

$$\forall A \in M. \forall (H, a) \in \mathcal{R}.\ H \subseteq A \Rightarrow a \in A$$

d.h., dass A $\mathcal{R}$–abgeschlossen ist. $\square$

Intuitiv ist die Menge der mit Hilfe von $\mathcal{R}$ aus der leeren Menge erzeugbaren Elemente die kleinste $\mathcal{R}$-abgeschlossene Teilmenge von M. Dass diese auch existiert, wird durch folgendes Lemma abgesichert.

Lemma 4.4 *Sei $\mathcal{C} \subseteq \mathcal{P}(M)$ eine Menge $\mathcal{R}$–abgeschlossener Teilmengen von M, wobei $\mathcal{R} \subseteq \mathcal{P}(M) \times M$. Dann ist*

$$\bigcap \mathcal{C} := \{a \in M \mid \forall\, C \in \mathcal{C}.\, a \in C\}$$

auch $\mathcal{R}$–abgeschlossen.

Beweis: Sei $\mathcal{C}$ eine Menge $\mathcal{R}$–abgeschlossener Teilmengen von M. Wenn $(H, a) \in \mathcal{R}$ und $H \subseteq \bigcap \mathcal{C}$, dann gilt für alle $C \in \mathcal{C}$, dass $H \subseteq C$ und somit $a \in C$. Also ist $\bigcap \mathcal{C}$ eine $\mathcal{R}$–abgeschlossene Menge. □

Definition 4.5 *Für ein Regelsystem $\mathcal{R} \subseteq \mathcal{P}(M) \times M$ sei* $\mathrm{Ind}(\mathcal{R})$ *der Durchschnitt* aller *$\mathcal{R}$–abgeschlossenen Teilmengen von M.* $\mathrm{Ind}(\mathcal{R})$ *ist wegen Lemma 4.4 selbst $\mathcal{R}$–abgeschlossen, d. h.* $\mathrm{Ind}(\mathcal{R})$ *ist die kleinste $\mathcal{R}$–abgeschlossene Teilmenge von M.* ◊

Theorem 4.6 *Sei $\mathcal{R} \subseteq \mathcal{P}(M) \times M$. Dann gilt*

1) $\Phi_{\mathcal{R}}(A) \subseteq A \Rightarrow \mathrm{Ind}(\mathcal{R}) \subseteq A$

2) $\mathrm{Ind}(\mathcal{R}) = \Phi_{\mathcal{R}}(\mathrm{Ind}(R))$

Beweis: (1) gilt, weil $\mathrm{Ind}(\mathcal{R})$ die kleinste $\mathcal{R}$–abgeschlossene Teilmenge von A ist. Da $\mathrm{Ind}(\mathcal{R})$ $\mathcal{R}$-abgeschlossen ist, gilt $\Phi_{\mathcal{R}}(\mathrm{Ind}(\mathcal{R})) \subseteq \mathrm{Ind}(R)$. Außerdem ist $\Phi_{\mathcal{R}}$ monoton, weshalb

$$\Phi_{\mathcal{R}}(\Phi_{\mathcal{R}}(\mathrm{Ind}(\mathcal{R}))) \subseteq \Phi_{\mathcal{R}}(\mathrm{Ind}(\mathcal{R}))$$

und somit ist $\Phi_{\mathcal{R}}(\mathrm{Ind}(\mathcal{R}))$ selbst auch $\mathcal{R}$–abgeschlossen. Also gilt auch $\mathrm{Ind}(\mathcal{R}) \subseteq \Phi_{\mathcal{R}}(\mathrm{Ind}(\mathcal{R}))$, da $\mathrm{Ind}(\mathcal{R})$ die kleinste $\mathcal{R}$–abgeschlossene Teilmenge ist, womit Behauptung (2) bewiesen ist. □

Bemerkung
Behauptung (1) von Theorem 4.6 lässt sich als *Induktionsprinzip* – genannt *Regelinduktion* – verwenden. Wenn $P \subseteq M$ ein Prädikat auf M ist, so gilt $\mathrm{Ind}(\mathcal{R}) \subseteq P$ d. h., dass alle Elemente aus $\mathrm{Ind}(R)$ das Prädikat P erfüllen, falls P selbst $\mathcal{R}$–abgeschlossen ist. Dies läßt sich formal folgendermaßen ausdrücken

$$\forall P \in \mathcal{P}(M).\ (\forall\, (H, a) \in \mathcal{R}.\ H \subseteq P \Rightarrow a \in P) \Rightarrow \forall a \in \mathrm{Ind}(\mathcal{R}).\ a \in P\ .$$

Regelinduktion ist *a priori* nur ein hinreichendes Kriterium für $\mathrm{Ind}(\mathcal{R}) \subseteq P$, falls P bloß eine beliebige Teilmenge von M ist. Für $P \subseteq \mathrm{Ind}(\mathcal{R})$ jedoch, d.h. den interessierenden Fall, gilt $P = \mathrm{Ind}(\mathcal{R})$ genau dann, wenn P $\mathcal{R}$–abgeschlossen ist, d.h. formal ausgedrückt

$$\forall P \subseteq \mathrm{Ind}(\mathcal{R}).\ (\forall (H,a) \in \mathcal{R}.\ H \subseteq P \Rightarrow a \in P) \Leftrightarrow \forall a \in \mathrm{Ind}(\mathcal{R}).\ a \in P$$

Behauptung (2) von Theorem 4.6 besagt, dass jedes $a \in \mathrm{Ind}(\mathcal{R})$ Konklusion einer $\mathcal{R}$-Regel (H,a) ist mit $H \subseteq \mathrm{Ind}(\mathcal{R})$, d.h.

$$\forall a \in \mathrm{Ind}(\mathcal{R}).\ \exists H \subseteq \mathrm{Ind}(\mathcal{R}).\ (H,a) \in \mathcal{R}\ \ .$$

Wir führen nun noch einige Beispiele für Regelsysteme an, die interessante induktive Mengen definieren.

1) Die Relation $\mathrm{ADD} = \{(\mathrm{n},\mathrm{m},\mathrm{n}+\mathrm{m}) \mid \mathrm{n},\mathrm{m} \in \mathbb{N}\}$ wird durch folgendes Regelsystem auf $\mathbb{N}^3$ definiert:

   $$\mathcal{R} = \{(\emptyset,(n,0,n)) \mid n \in \mathbb{N}\} \cup \{(\{(n,m,k)\},(n,m+1,k+1)) \mid n,m,k \in \mathbb{N}\}$$

   Es erweist sich als nützlich, Regelmengen etwas übersichtlicher durch so genannte *Regelschemata* anzugeben:

   $$\frac{}{\mathrm{ADD}(\mathrm{n},0,\mathrm{n})}\ (\mathrm{ADD1}) \qquad \frac{\mathrm{ADD}(\mathrm{n},\mathrm{m},\mathrm{k})}{\mathrm{ADD}(\mathrm{n},\mathrm{m}+1,\mathrm{k}+1)}\ (\mathrm{ADD2})$$

   Vergleiche diese Regelschemata mit den Klauseln des folgenden *Logikprogramms*

   $$\mathrm{ADD}(\mathrm{n},0,\mathrm{n}) \leftarrow \qquad \mathrm{ADD}(\mathrm{n},\mathrm{m}+1,\mathrm{k}+1) \leftarrow \mathrm{ADD}(\mathrm{n},\mathrm{m},\mathrm{k})$$

   zur Berechnung der Addition.
   Eigentlich sind die Namen ADD1, ADD2 in den Regeln (ADD1), (ADD2) überflüssig; ihre einzige Funktion ist, einen „sprechenden“ Namen für die durch diese Regeln induktiv definierte Teilmenge festzulegen.

2) Folgendes Regelsystem definiert den Graphen der Funktion $\mathrm{rev} : \mathrm{A}^* \to \mathrm{A}^*$, die eine Liste $a_1 \dots a_n$ auf die umgekehrte Liste $a_n \dots a_1$ abbildet.

   $$\mathcal{R} = \{(\emptyset,(\varepsilon,\varepsilon))\} \cup \{(\{(u,v)\},(au,va)) \mid u,v \in A^*, a \in A\}$$

   Als Logikprogramm liest sich die Definition von $\mathrm{REV} = \mathrm{Graph}(\mathrm{rev})$ folgendermaßen

   $$\mathrm{REV}(\varepsilon,\varepsilon) \leftarrow \qquad \mathrm{REV}(au,va) \leftarrow \mathrm{REV}(u,v)$$

In der Mathematik sind auch Regelsysteme anzutreffen, wo Regeln auch unendliche Prämissenmengen haben. Ein typisches Beispiel dafür sind die sogenannten *Borel Mengen*, die in der Analysis und Wahrscheinlichkeitstheorie von grundlegender Bedeutung sind.

3) Die Menge $\mathcal{B} \subseteq \mathcal{P}(\mathbb{R})$ der *Borel Mengen* wird induktiv wie folgt definiert

   (1) jedes Intervall $[a, b]$ ist in $\mathcal{B}$ (einschließlich der leeren Menge);
   (2) wenn $A \in \mathcal{B}$, so auch $\mathbb{R} \setminus A \in \mathcal{B}$;
   (3) wenn $(A_n \mid n \in \mathbb{N})$ eine Familie von Borel Mengen ist, so ist auch $\bigcup_{n\in\mathbb{N}} A_n$ eine Borel Menge.

   Als Regelsystem lässt sich diese induktive Definition formal als $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2 \cup \mathcal{R}_3$ schreiben, wobei

$$\begin{aligned} \mathcal{R}_1 &= \{(\emptyset, [a, b]) \mid a, b \in \mathbb{R}\} \\ \mathcal{R}_2 &= \{(\{A\}, \mathbb{R} \setminus A) \mid A \in \mathcal{P}(\mathbb{R})\} \\ \mathcal{R}_3 &= \{(\mathcal{S}, \bigcup \mathcal{S}) \mid \mathcal{S} \subseteq \mathcal{P}(\mathbb{R})\,,\ \mathcal{S} \text{ abzählbar}\} \end{aligned}$$

4) In der Logik und theoretischen Informatik ist die Menge WT (wellfounded trees) der *wohlfundierten Bäume* von großem Interesse. Sie wird induktiv als Teilmenge von $\mathcal{P}(\mathbb{N}^*)$ durch folgendes Regelsystem definiert:

   (1) $\emptyset, \{\varepsilon\} \in$ WT
   (2) wenn $(T_n \mid n \in \mathbb{N})$ eine Familie in WT ist, so ist $\{nw \mid n \in \mathbb{N}, w \in T_n\} \in$ WT

   Man zeigt leicht durch Induktion über WT, dass jedes $\mathcal{T} \in$ WT folgende zwei Eigenschaften erfüllt:

   (tr) $uv \in \mathcal{T} \Rightarrow u \in \mathcal{T}$
   (wf) für jede Folge $a : \mathbb{N} \to \mathbb{N}$ gibt es ein $n \in \mathbb{N}$, so dass

$$\bar{a}(n) := a_0 \dots a_{n-1} \notin \mathcal{T}$$

   Die Elemente aus $\mathcal{T}$ werden als „Pfade" zu den „Knoten" von $\mathcal{T}$ interpretiert, wobei die Zahlen angeben, welche Abzweigung man jeweils wählen muss. Die Bedingung (tr) bringt zum Ausdruck, dass $\mathcal{T}$ ein Baum ist, die Bedingung (wf) (für wohlfundiert) drückt aus, dass es in $\mathcal{T}$ keinen unendlichen Pfad gibt. (Allerdings können die Bäume unendliche Breite haben; Bäume endlicher Breite erhält man, wenn man für fast alle T_n in (2) die leere Menge wählt.)

   Darüber hinaus gilt, dass die Bedingungen (tr) und (wf) genau die Elemente aus WT charakterisieren, was man folgendermaßen einsieht. Eine Menge $\mathcal{T} \subseteq \mathbb{N}^*$, die (tr) erfüllt und nicht in WT ist, muss einen Unterbaum besitzen, der auch nicht in $\mathcal{T}$ ist, d.h. es gibt ein $n \in \mathbb{N}$, so dass $\mathcal{T}_n := \{s \mid ns \in \mathcal{T}\} \notin$ WT. Durch wiederholte Anwendung dieses Arguments erhielte man somit einen unendlichen Pfad in $\mathcal{T}$ und somit würde $\mathcal{T}$ die Bedingung (wf) verletzen.

I.5 Ordnungen und Verbände

Definition 5.1 *Eine Relation* $R \subseteq A \times A$ *heißt* Präordnung *auf der Menge* A *genau dann, wenn sie reflexiv und transitiv ist, also folgende Bedingungen für alle* $x, y, z \in A$ *gelten:*

(refl)	$x \; R \; x$ *für alle* $x \in A$
(trans)	$x \; R \; y \wedge y \; R \; z \Rightarrow x \; R \; z$

Eine Relation $R \subseteq A \times A$ *heißt* Ordnung *auf* A*, wenn* R *eine Präordnung ist und außerdem gilt, dass* R antisymmetrisch *ist, also*

(antisym) $\quad x \; R \; y \wedge y \; R \; x \Rightarrow x = y.$

Eine Ordnung (A, R) *heißt* total*, wenn für alle* $x, y \in A$ *gilt:*

$$x \; R \; y \;\text{ oder }\; \text{y R x}$$

Man benutzt oft auch Symbole wie $\lesssim$ *oder* $\leq$ *für* R*.* ◊

Bemerkung: Oft werden in der Literatur Ordnungen als partielle Ordnungen bezeichnet und nur totale Ordnungen als Ordnungen.

Wir betrachten nun einige Beispiele.

(1) $\leq$ auf $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$ ist eine totale Ordnung.

(2) $n|m :\Leftrightarrow \exists k.\; n \cdot k = m$ ist eine (partielle) Ordnung auf den positiven natürlichen Zahlen $\mathbb{N}^+$.

(3) für jede Menge X ist $(\mathcal{P}(X), \subseteq)$ eine (partielle) Ordnung.

(4) Wenn $(A, \leq)$ eine Ordnung ist, so ist auch folgende Relation auf A^I eine Ordnung

$$f \leq g \;:\Leftrightarrow\; \forall i \in I.\; f(i) \leq g(i)$$

Man beachte allerdings, dass $(A^I, \leq)$ im allgemeinen partiell sein wird, auch wenn $(A, \leq)$ selbst total ist.

Wir betrachten nun strukturerhaltende Abbildungen zwischen (Prä-)Ordnungen.

Definition 5.2 *Es seien* $(P, \leq)$ *und* $(Q, \sqsubseteq)$ *zwei Präordnungen. Eine Funktion* $f : P \to Q$ *heißt* monoton *genau dann, wenn für alle* $x, y \in P$ *gilt:*

$$x \leq y \Rightarrow f(x) \sqsubseteq f(y) \quad .$$

Eine monotone Abbildung $f : (P, \leq) \to (Q, \sqsubseteq)$ *heißt* Präordnungs-Isomorphismus*, wenn* f *bijektiv ist und die Umkehrabbildung ebenfalls monoton ist.* ◊

Bemerkung: Man beachte, dass nicht jede bijektive monotone Funktion ein Isomorphismus sein muss. Identische Abbildungen auf Präordnungen sind monoton und die monotonen Abbildungen sind unter Komposition abgeschlossen.

Definition 5.3 *Sei $(P, \leq)$ eine Präordnung und $A \subseteq P$, $x \in P$.*

- *x ist eine* untere Schranke von A *($x \leq A$) gdw $x \leq z$ für alle $z \in A$*
- *x ist eine* obere Schranke von A *($A \leq x$) gdw $z \leq x$ für alle $z \in A$*
- *x ist* Infimum *von A gdw* (i) $x \leq A$
  (ii) *$y \leq x$ für alle $y \in P$ mit $y \leq A$*
- *x ist* Supremum *von A gdw* (i) $A \leq x$
  (ii) *$x \leq y$ für alle $y \in P$ mit $A \leq y$.* ◊

Wenn $(P, \leq)$ eine (partielle) Ordnung ist, sind Infima und Suprema eindeutig bestimmt, sofern sie existieren. Wir bezeichnen sie in diesem Fall mit $\inf A$ bzw. $\sup A$. Wenn x Infimum bzw. Supremum von A ist, so ist x die größte untere bzw. kleinste obere Schranke von A.

Im allgemeinen erhalten monotone Abbildungen nicht Infima und Suprema, d.h. wenn $f : (P_1, \leq_1) \to (P_2, \leq_2)$ und x Infimum/Supremum von A ist, so muss $f(x)$ *nicht* notwendigerweise Infimum/Supremum von $f[A] = \{f(x) | x \in A\}$ sein. Wenn jedoch f ein Ordnungs–Isomorphismus ist, so erhält f alle Infima und Suprema.

Wir wenden uns nun einer speziellen Klasse von partiellen Ordnungen zu, den so genannten *Verbänden.*

Definition 5.4 *Eine Ordnung $(L, \leq)$ heißt* Verband*, wenn jede* endliche *Teilmenge von L, also insbesondere auch die leere Teilmenge, ein Infimum und ein Supremum besitzt. Für $x, y \in L$ bezeichne $x \wedge y$ bzw. $x \vee y$ das Infimum bzw. Supremum von $\{x, y\}$. Das Infimum bzw. Supremum der leeren Menge $\emptyset$ wird mit* 1 *bzw.* 0 *bezeichnet.* ◊

Bemerkung: Verband heißt auf englisch *lattice.* 1 ist das größte Element von $(L, \leq)$ und 0 das kleinste Element, da für alle $x \in L$ gilt $x \leq \emptyset$ und $\emptyset \leq x$.

Da sich Infima und Suprema von endlichen Teilmengen darstellen lassen als

$$\inf\{x_1, \ldots, x_n\} = x_1 \wedge \cdots \wedge x_n \qquad \text{und} \qquad \sup\{x_1, \ldots, x_n\} = x_1 \vee \cdots \vee x_n$$

ist die Existenz endlicher Infima bzw. Suprema gleichbedeutend mit der Existenz eines größten bzw. kleinsten Elements und der Existenz binärer Infima $x \wedge y$ bzw. binärer Suprema $x \vee y$ für alle $x, y \in L$.

Definition 5.5 *Ein Verband $(L, \leq)$ heißt* vollständig *genau dann, wenn er Infima und Suprema* beliebiger *Teilmengen besitzt.* ◊

Für vollständige Verbände reicht es aus, nur die Existenz von Suprema beliebiger Teilmengen zu fordern, da das Infimum von $X \subseteq L$ gegeben ist durch das Supremum der unteren Schranken von X (Beweis als Übung!). Genauso reicht es aus, die Existenz von Infima beliebiger Teilmengen von L zu fordern.

Wir betrachten nun einige Beispiele für Verbände.

(1) Für jede Menge A ist $(\mathcal{P}(A), \subseteq)$ ein vollständiger Verband.

(2) $\{X \in \mathcal{P}(A) \mid X \text{ endlich oder } A \backslash X \text{ endlich}\}$ ist bezüglich der Inklusion $\subseteq$ ein Verband, jedoch *kein* vollständiger Verband, wenn A *unendlich* ist.

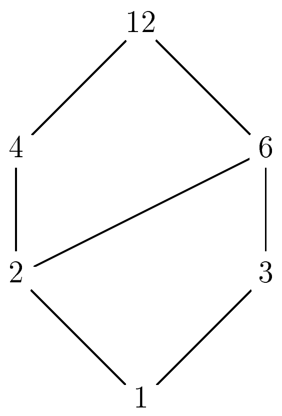
(3) Sei $(P, \leq)$ eine Ordnung. Dann ist die Menge der nach unten abgeschlossenen Mengen (*downward closed sets*)

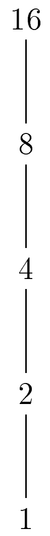
$$\mathsf{dcl}(P, \leq) := \{X \subseteq P \mid \forall x, y \in P.\ y \leq x \in X \Rightarrow y \in X\}$$

bezüglich der Inklusion $\subseteq$ ein vollständiger Verband.

(4) $(\mathbb{N}^+, |\,)$ ist kein Verband, weil er kein größtes Element besitzt. Allerdings ist für jedes $n \in \mathbb{N}^+$ die Menge $\{i \in \mathbb{N}^+ \mid i | n\}$ ein vollständiger Verband bezüglich der Teilbarkeitsrelation $|$. Wir erhalten folgende Hasse-Diagramme:

für $n = 12$ und für $n = 16$:





Nun kommen wir zu einigen wichtigen Eigenschaften von Verbänden.

Definition 5.6 *Ein Verband* $(L, \leq)$ *heißt* distributiv*, wenn*

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

für alle $x, y, z \in L$ *gilt.* $\lozenge$

Lemma 5.7 *In einem distributiven Verband gilt auch*

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

für alle $x, y, z \in L$.

Beweis:

$$\begin{array}{rlr} & (x \vee y) \wedge (x \vee z) & \text{(Distrib.)} \\ = & ((x \vee y) \wedge x) \vee ((x \vee y) \wedge z) & (\text{weil } x = (x \vee y) \wedge x) \\ = & x \vee ((x \vee y) \wedge z) & \text{(Distrib.)} \\ = & x \vee (x \wedge z) \vee (y \wedge z) & (\text{weil } x = x \vee (x \wedge z)) \\ = & x \vee (y \wedge z) & \end{array}$$

□

Definition 5.8 *Ein Verband $(L, \leq)$ heißt* Heyting-Verband*, wenn für alle $x, y \in L$ die Menge*

$$\{z \in L \mid x \wedge z \leq y\}$$

ein größtes Element besitzt. Dieses eindeutig bestimmte Element wird mit $x \to y$ bezeichnet. ◇

Bemerkung: Es gilt $z \leq x \to y$ gdw $x \wedge z \leq y$.

Typische Beispiele für Heyting-Verbände sind $\mathsf{dcl}(P, \leq)$ für Ordnungen $(P, \leq)$. Denn sind $A, B \in \mathsf{dcl}(P, \leq)$, dann ist

$$\begin{array}{rcl} A \to B & = & \{z \in P \mid \forall y \in A.\ y \leq z \Rightarrow y \in B\} \\ & = & \{z \in P \mid \forall y \leq z.\ y \in A \Rightarrow y \in B\} \end{array}$$

(Beweis als Übung!). Insbesondere ist also $(\mathcal{P}(X), \subseteq)$ ein Heyting–Verband, wobei mit $A \to B = (X \backslash A) \cup B$.

Lemma 5.9 *Jeder Heyting–Verband ist distributiv.*

Beweis: Übung. □

In einem Heyting–Verband bezeichnen wir $\neg a := a \to 0$ als *Pseudokomplement* von a, da $\neg a$ das größte b mit $a \wedge b = 0$ ist.

Achtung! Obwohl $a \wedge \neg a = 0$ immer gilt, muss $a \vee \neg a$ im allgemeinen nicht gelten. Falls jedoch $a \vee \neg a = 1$ immer gilt, dann heißt der Verband *Boolesch.*

Definition 5.10 *Ein* Boolescher Verband *ist ein Heyting-Verband, so dass für alle $a \in L$ gilt*

$$a \vee \neg\, a = 1 \quad . \qquad \Diamond$$

Offenbar sind die Verbände der Gestalt $(\mathcal{P}(X), \subseteq)$ immer Boolesch. Ein Beispiel für einen nichtvollständigen Booleschen Verband ist

$$\{A \in \mathcal{P}(X) \mid A \text{ endlich oder } X \backslash A \text{ endlich}\}$$

für unendliche Mengen A.
Wir werden in Teil II über Logik sehen, dass *Heyting–Verbände* und *Boolesche Verbände* als verallgemeinerte Strukturen von Wahrheitswerten dienen, in denen man aussagenlogische Formeln interpretieren kann.

II Allgemeine Algebra

In Mathematik und Informatik ist es oft aus „organisatorischen“ Gründen notwendig, eine Menge von *Objekten* zusammen mit einer Menge von ausgewählten *Grundoperationen* auf dieser Menge zu einer Einheit zusammenzufassen. Solche Einheiten nennt man *algebraische Strukturen* oder einfach *Algebren*.

Beispiele dafür sind Gruppen, Monoide, Vektorräume, Listen, Keller (Stacks) etc. Eine grobe Unterteilung der Algebren erfolgt nach ihrem *Strukturtyp*; ein solcher ist durch die Angabe der Menge der *Funktionssymbole* zusammen mit ihrer Stelligkeit gegeben und wird als *Signatur* bezeichnet. Da eine algebraischen Struktur oft Objekte verschiedener Natur umfasst, (etwa Skalare und Vektoren im Fall von Vektorräumen) empfiehlt es sich, diese in verschiedene Sorten aufzuteilen und für Argumente und Werte einer Funktion ihre Sorten vorzuschreiben. Das vermeidet partielle Funktionen, die die Theorie erheblich verkomplizieren würden. Konkret heißt das, dass jedes Funktionssymbol f mit einer *Stelligkeit* versehen wird.

II.1 Mehrsortige Algebren

Zuerst formalisieren wir den Begriff einer Signatur.

Definition 1.1 *Eine* Signatur *bzw. ein* Strukturtyp *ist ein Tripel*

$$\Sigma = (S, F, \mathrm{ar})$$

wobei S und F Mengen und $\mathrm{ar} : F \to S^* \times S$ *eine Funktion ist. S heißt Menge der* Sortensymbole, *F heißt Menge der* Funktionssymbole *und* ar *heißt* Stelligkeitsfunktion *(englisch* arity function) *der Signatur Σ.*

Statt S, F und ar schreiben wir oft $\Sigma_{\mathrm{sort}}, \Sigma_{\mathrm{func}}, \Sigma_{\mathrm{ar}}$, und für $\mathrm{ar}(f) = (s_1 \dots s_n, s)$ schreiben wir auch $f : s_1 \times \dots \times s_n \to s$. $\Diamond$

Als nächstes definieren wir den Begriff einer Σ-Algebra für eine Signatur Σ.

Definition 1.2 *Sei $\Sigma = (S, F, \mathrm{ar})$ eine Signatur. Eine Σ-Algebra A besteht aus der Angabe*

i) *einer Menge A_s für jedes $s \in S$ und*

ii) *einer Funktion $f^A : A_{s_1} \times \dots \times A_{s_n} \to A_s$ für jedes Funktionssymbol $f \in F$, wobei $\mathrm{ar}(f) = (s_1 \dots s_n, s)$.*

A_s heißt Trägermenge *der Sorte s in A (englisch* carrier set*) und f^A heißt* Interpretation von f in A. $\Diamond$

Notation:
Für $\vec{s} = s_1 \ldots s_n \in S^*$ bezeichne $A_{\vec{s}}$ als Abkürzung die Menge $A_{s_1} \times \ldots \times A_{s_n}$. Also schreiben wir $f^A : A_{\vec{s}} \to A_s$ für $\mathrm{ar}(f) = (\vec{s}, s)$.

Falls $\mathrm{ar}(f) = (\varepsilon, s)$, so schreiben wir $f : \; \to s$. In diesem Fall nennt man f eine *Konstante* der Sorte s, da $A_\varepsilon = \{\emptyset\}$.

Wenn S aus genau einer Sorte besteht, so ist jedes $(\vec{s}, s) \in S^* \times S$ eindeutig durch $n := \mathrm{length}(\vec{s})$ bestimmt und wir können die Stelligkeit eines Funktionssymbols eindeutig durch die *Anzahl* seiner Argumente beschreiben. Allerdings ist in den meisten Informatikanwendungen meist mehr als eine Sorte vonnöten (wie z.B. bei Listen, Stacks, labelled trees...).

Wir werden nun einige Beispiele von Signaturen und Algebren betrachten und einen Formalismus für ihre Präsentation angeben. Im Rahmen der Informatik wird dies oft als *Spezifikation* bezeichnet. In diesem Falle werden aber neben der Signatur zusätzlich Axiome angegeben, wodurch man die *Klasse der Algebren* aussondert, die diese Axiome *erfüllen*. Oft werden diese Axiome in der vertrauten Form von Gleichungen formuliert. Dies ist jedoch nicht immer ausreichend, wie z.B. bei Körpern.

Es ist prinzipiell unmöglich, durch Axiome genau eine einzige Algebra zu beschreiben: Wenn eine Algebra A die Axiome erfüllt und $A \cong A'$, also A isomorph zu A' ist, dann erfüllt die Algebra A' ebenfalls diese Axiome.

Beispiel:

NAT	$\equiv$
sorts	nat
funcs	zero $:\to$ nat succ : nat $\to$ nat
axioms	

Wir haben also eine Signatur $\Sigma = (S, F, \mathrm{ar})$ mit

$$S = \{\mathrm{nat}\} \quad F = \{\mathrm{zero}, \mathrm{succ}\} \quad \mathrm{ar}(\mathrm{zero}) = (\varepsilon, \mathrm{nat}) \quad \mathrm{ar}(\mathrm{succ}) = (\mathrm{nat}, \mathrm{nat})$$

Folgende drei Algebren erfüllen obige Spezifikation.

a) $\mathbb{N}$ mit: $\mathbb{N}_{\mathrm{nat}} = \mathbb{N}$ $\quad \mathrm{zero}^{\mathbb{N}} = 0$ $\quad \mathrm{succ}^{\mathbb{N}}(n) = n + 1$

b) $\mathbb{R}$ mit: $\mathbb{R}_{\mathrm{nat}} = \mathbb{R}$ $\quad \mathrm{zero}^{\mathbb{R}} = 0$ $\quad \mathrm{succ}^{\mathbb{R}}(x) = x + 1$

c) $\mathbb{Z}_n$ mit: $(\mathbb{Z}_n)_{\mathrm{nat}} = \mathbb{Z}/{\equiv_n} =: \mathbb{Z}_n$ (also die ganze Zahlen modulo n)
$\mathrm{zero}^{\mathbb{Z}_n} = [0]_n$ $\quad \mathrm{succ}^{\mathbb{Z}_n}([k]_n) = [k+1]_n$

Beispiel (a) ist das *intendierte Modell*, (b) enthält Objekte, die nicht durch einen Term der Form $\mathrm{succ}^n(\mathrm{zero})$ ausgedrückt werden können und (c) identifiziert Terme, etwa $\mathrm{succ}^n(\mathrm{zero})$ und zero, die im intendierten Modell nicht gleich sein sollen (und es auch nicht sind).

Die Menge der (sinnvollen) Operationen auf $\mathbb{N}$ ist nicht eindeutig durch die Menge $\mathbb{N}$ bestimmt. Das sieht man daran, dass etwa die Addition nicht aus zero und succ durch einen expliziten Term definierbar ist (Übung).

Wir können aber die Addition und die sie bestimmenden Axiome zu unserer Spezifikation dazu geben.

NAT–ADD	$\equiv$
sorts	nat
funcs	zero $:\to$ nat succ : nat $\to$ nat add : nat $\times$ nat $\to$ nat
axioms	$\mathrm{add}(x, \mathrm{zero}) = x$ $\mathrm{add}(x, \mathrm{succ}(y)) = \mathrm{succ}(\mathrm{add}(x, y))$

Es erhebt sich nun die Frage, ob folgende in $\mathbb{N}$ offensichtlich gültigen Eigenschaften in allen Modellen der Spezifikation NAT–ADD gelten:

- $\mathrm{add}(\mathrm{add}(x, y), z) = \mathrm{add}(x, \mathrm{add}(y, z))$
- $\mathrm{add}(x, y) = \mathrm{add}(y, x)$
- $\mathrm{zero} \neq \mathrm{succ}(x)$
- $\mathrm{succ}(x) = \mathrm{succ}(y) \Rightarrow x = y$
- $x \neq \mathrm{succ}(x)$

Leider ist dies nicht der Fall, wie wir in den Übungen sehen werden. Dieses Problem kann allerdings behoben werden, indem wir die **Axiome** $\mathrm{zero} \neq \mathrm{succ}(x)$ und $\mathrm{succ}(x) = \mathrm{succ}(y) \Rightarrow x = y$ sowie folgendes

Induktionsschema

$$A(\mathrm{zero}) \wedge (\forall x\colon \mathrm{nat}.\ A(x) \Rightarrow A(\mathrm{succ}(x))) \Rightarrow \forall x\colon \mathrm{nat}.\ A(x)$$

für alle Prädikate $A(x)$ zu den Axiomen dazunehmen. Man beachte, dass dies eine unendliche Menge von Axiomen ist, da man für *jedes* Prädikat $A(x)$ obige Behauptung als Axiom hinzunehmen muss!

Wir betrachten nun Beispiele mit mehr als einer Sorte.

NAT–EQ $\equiv$

sorts nat, bool

funcs zero : $\to$ nat
succ : nat $\to$ nat
true : $\to$ bool
false : $\to$ bool
eq : nat $\times$ nat $\to$ bool

axioms eq(zero, succ(x)) = false
eq(succ(x), zero) = false
eq(zero, zero) = true
eq(succ(x), succ(y)) = eq(x, y)

Die Signatur eines Automaten ist auch zweisortig.

AUTOMAT $\equiv$

sorts act, stat
funcs trans : act $\times$ stat $\to$ stat

Als nächstes betrachten wir die Spezifikation von Listen bzw. Stacks (über einer Menge von Elementen).

LIST $\equiv$

sorts elem, list

funcs empty : $\to$ list
cons : elem $\times$ list $\to$ list
head : list $\to$ elem
tail : list $\to$ list

axioms head(cons(a, l)) = a
tail(cons(a, l)) = l
tail(empty) = empty

STACK $\equiv$

sorts elem, stack

funcs empty : $\to$ stack
push : elem $\times$ stack $\to$ stack
top : stack $\to$ elem
pop : stack $\to$ stack

axioms top(push(a, s)) = a
pop(push(a, s)) = s
pop(empty) = empty

Offenbar unterscheiden sich die beiden letzten Spezifikationen bloß durch Umbenennung der Sorten- und Funktionssymbole, d.h. rein algebraisch laufen sie auf dasselbe hinaus. Die in der Informatik übliche Unterscheidung zwischen Listen und Stacks bezieht sich auf die verschiedenen üblichen *Implementierungen* ein und derselben Datenstruktur.

In den beiden Spezifikationen wird absichtlich offen gelassen, was head(empty) (bzw. top(empty)) ist. Insbesondere können wir nicht fordern, dass

$$l = \mathrm{cons}(\mathrm{head}(l), \mathrm{tail}(l))$$

allgemein gilt (obwohl das für Elemente l der Form $\mathrm{cons}(a, r)$ durchaus der Fall ist), weil die Gleichung $\mathrm{empty} = \mathrm{cons}(\mathrm{head}(\mathrm{empty}), \mathrm{tail}(\mathrm{empty}))$ in den intendierten Modellen nicht gilt. Dies widerspiegelt die in der Informatik übliche Redeweise „head(empty) ist undefiniert".

Es ist jedoch nicht klar was diese Redeweise im Kontext von (totalen) Algebren überhaupt heißen soll. Wenn Operationen als partielle Funktionen interpretiert werden können — man also *partielle Algebren* zulässt — würde „Undefiniertheit" einen präzisen mathematischen Sinn erhalten (siehe Vorlesungen von P. Burmeister bzw. sein Buch zu diesem Thema!). Alternativ könnte man in elem ein *error element* $\mathrm{err} : \to \mathrm{elem}$ einführen und verlangen, dass $\mathrm{head}(\mathrm{empty}) = \mathrm{err}$ und $f(t_1, \ldots, t_n) = \mathrm{err}$, falls mindestens eins der $t_i = \mathrm{err}$. Allerdings sind dann die üblicherweise geforderten Gleichungen nur mehr in eingeschränktem Sinne gültig, nämlich nur, wenn man für die Variablen Werte einsetzt, die von den neu hinzugefügten error–Elementen verschieden sind. Ein ähnliches Problem taucht schon in der wohletablierten Theorie der Körper auf, wo die Division durch Null „undefiniert" ist.

Als nächstes betrachten wir „Spezifikationen" der „handelsüblichen" algebraischen Strukturen Monoid, Gruppe, Ring, Körper etc., die in der Mathematik bis ins (letzte?) Detail studiert wurden, jedenfalls mehr als die Datentypen der Informatik.

- MONOID $\equiv$

  sorts elem

  funcs $\mathrm{add} : \mathrm{elem} \times \mathrm{elem} \to \mathrm{elem}$
  $\mathrm{zero} : \to \mathrm{elem}$

  axioms $\mathrm{add}(\mathrm{add}(x, y), z) = \mathrm{add}(x, \mathrm{add}(y, z))$
  $\mathrm{add}(\mathrm{zero}, x) = x$
  $\mathrm{add}(x, \mathrm{zero}) = x$

- GROUP $\equiv$ MONOID +

  funcs $\mathrm{inv} : \mathrm{elem} \to \mathrm{elem}$

  axioms $\mathrm{add}(x, \mathrm{inv}(x)) = \mathrm{zero}$

- GROUP-ABELIAN $\equiv$ GROUP +

  axioms $\quad$ $\mathrm{add}(x, y) = \mathrm{add}(y, x)$

- RING $\equiv$ GROUP-ABELIAN +

  funcs $\quad$ mult : elem $\times$ elem $\rightarrow$ elem

  axioms $\quad$ $\mathrm{mult}(x,\mathrm{add}(y, z)) = \mathrm{add}(\mathrm{mult}(x, y),\mathrm{mult}(x, z))$
  $\mathrm{mult}(\mathrm{add}(y, z), x) = \mathrm{add}(\mathrm{mult}(y, x), \mathrm{mult}(z, x))$
  $\mathrm{mult}(\mathrm{mult}(x, y),z) = \mathrm{mult}(x,\mathrm{mult}(y, z))$

- RING-UNIT $\equiv$ RING +

  funcs $\quad$ one : $\rightarrow$ elem

  axioms $\quad$ $\mathrm{mult}(\mathrm{one},x) = x$
  $\mathrm{mult}(x,\mathrm{one}) = x$

- RING-COMM $\equiv$ RING-UNIT +

  axioms $\quad$ $\mathrm{mult}(x, y) = \mathrm{mult}(y, x)$

Die Struktur der *Körper (engl.* fields) kann nicht mehr rein durch Gleichungen axiomatisiert werden.

- FIELD $\equiv$ RING-COMM +

  axioms $\quad$ $\forall x : \mathrm{elem}.\, x \neq \mathrm{zero} \rightarrow \exists y : \mathrm{elem}.\, \mathrm{mult}(x, y) = \mathrm{one}$

Abschließend bringen wir das Beispiel der Vektorräume, um zu illustrieren, dass auch außerhalb der Informatik mehrsortige Algebren natürlich auftauchen:

- VECTOR-SPACE $\equiv$ FIELD +

  sorts $\quad$ vector

  funcs $\quad$ null : $\rightarrow$ vector
  add_vect : vector $\times$ vector $\rightarrow$ vector
  mult_scal : elem $\times$ vector $\rightarrow$ vector

  axioms $\quad$ $\mathrm{add\_vect}(\mathrm{add\_vect}(x, y),z) = \mathrm{add\_vect}\ (x,\mathrm{add\_vect}(y, z))$
  $\mathrm{add\_vect}(x,y) = \mathrm{add\_vect}(y,x)$
  $\mathrm{add\_vect}(\mathrm{null},x) = x$
  $\mathrm{add\_vect}(x,\mathrm{null}) = x$
  $\mathrm{mult\_scal}(a,\mathrm{add\_vect}(x, y)) =$
  $\quad \mathrm{add\_vect}(\mathrm{mult\_scal}(a, x),\mathrm{mult\_scal}(a, y))$
  $\mathrm{mult\_scal}(a,\mathrm{null}) = \mathrm{null}$
  $\mathrm{mult\_scal}(\mathrm{add}(a, b),x) =$
  $\quad \mathrm{add\_vect}(\mathrm{mult\_scal}(a, x),\mathrm{mult\_scal}(b, x))$
  $\mathrm{mult\_scal}(\mathrm{zero},x) = \mathrm{null}$
  $\mathrm{mult\_scal}(\mathrm{mult}(a, b),x)) = \mathrm{mult\_scal}(a,\mathrm{mult\_scal}(b, x))$
  $\mathrm{mult\_scal}(\mathrm{one},x) = x$

Wenn man in obiger Spezifikation „FIELD“ durch „RING_COMM“ ersetzt, erhält man den Begriff eines „Moduls“, d.h. eines „Vektorraums“ über einem kommutativen Ring, nicht notwendigerweise über einem Körper.

II.2 Unteralgebren, Homomorphismen und Kongruenzen mehrsortiger Algebren

Zuerst verallgemeinern wir den Begriff der Unteralgebra auf beliebige Σ–Algebren.

Definition 2.1 *Sei $\Sigma = (S, F, \mathrm{ar})$ eine Signatur und A eine Σ–Algebra. Eine Σ-Algebra B heißt* Unter–Σ–Algebra *von A genau dann, wenn*

(i) $B_s \subseteq A_s$ *für alle* $s \in S$

(ii) $f^B(y_1, \ldots, y_n) = f^A(y_1, \ldots y_n)$ *für alle* $f \in F$ *und* $y_i \in B_{s_i}$ *für* $i = 1, \ldots, n$, *wobei* $\mathrm{ar}(f) = (s_1 \ldots s_n, s)$. ◊

Eine Unteralgebra von A ist also gegeben durch eine Familie $B_s \subseteq A_s$ (für $s \in S$), die unter den Operationen der Algebra A *abgeschlossen* sind: durch Anwendung beliebiger Operationen bleibt man immer in B, d.h. für $f \in F$ mit $\mathrm{ar}(f) = (s_1 \ldots s_n, s)$ und $y_i \in B_{s_i}$ für $i = 1, \ldots, n$ ist auch $f^A(y_1, \ldots y_n) \in B_s$. Die Operationen von A *vererben* sich auf B durch *Einschränkung* (Restriktion).
Dabei ist die Abgeschlossenheit von B unter den Operationen von A wesentlich, da sonst die Restriktion der Operationen aus A auf B zu partiellen Operationen führen könnte.

Definition 2.2 *Sei $\Sigma = (S, F, \mathrm{ar})$ eine Signatur und A und B Σ–Algebren. Ein* Σ–Homomorphismus *(oder kurz* Homomorphismus*) von A nach B ist eine Familie von Funktionen $(h_s : A_s \to B_s)_{s \in S}$, so dass für $(s_1 \ldots s_n, s) \in S^* \times S$ und alle $f \in F$ mit $\mathrm{ar}(f) = (s_1 \ldots s_n, s)$ gilt:*

$$h_s(f^A(x_1, \ldots, x_n)) = f^B(h_{s_1}(x_1), \ldots, h_{s_n}(x_n))$$

für alle $i = 1, \ldots, n$ und $x_i \in A_{s_i}$. ◊

Intuitiv bedeutet obige Bedingung, dass Algebraoperationen und (Komponenten von) h *vertauschbar* sind, der Homomorphismus also die Operationen der Algebra *respektiert*. Dies veranschaulicht folgendes Diagramm:

$$\begin{tikzcd}
A_{\vec{s}} \arrow[r, "f^A"] \arrow[d, "h_{\vec{s}}"'] & A_s \arrow[d, "h_s"] \\
B_{\vec{s}} \arrow[r, "f^B"'] & B_s
\end{tikzcd}$$

wobei $A_{\vec{s}} = A_{s_1} \times \ldots \times A_{s_n}$, $B_{\vec{s}} = B_{s_1} \times \ldots \times B_{s_n}$ und $h_{\vec{s}}$ definiert ist als die Funktion $h_{\vec{s}}(x_1, \ldots, x_n) = (h_{s_1}(x_1), \ldots, h_{s_n}(x_n))$.

Für eine konkrete Algebra genügt oft die Vertauschbarkeit von h mit allen f aus einer *Teilmenge* F' aller Operationen F, um die Vertauschbarkeit von h mit allen $f \in F$ zu garantieren. Beispielsweise erhält jeder Monoidhomomorphismus zwischen Gruppen (aufgefasst als Monoide) automatisch die Inversenoperation (Übung!).

Als nächstes verallgemeinern wir den Begriff „Kongruenz“ auf allgemeine Σ-Algebren.

Definition 2.3 *Es sei* $\Sigma = (S, F, \mathrm{ar})$ *eine Signatur und* A *eine* Σ*–Algebra. Eine* Σ-Kongruenz(relation) R *auf* A *besteht aus einer Familie von Äquivalenzrelationen* $R_s \subseteq A_s \times A_s$ *(für jede Sorte* $s \in S$*), so dass für alle* $(s_1 \ldots s_n, s) \in S^* \times S$ *und* $f \in F$ *mit* $\mathrm{ar}(f) = (s_1 \ldots s_n, s)$ *gilt:*

$$f^A(x_1, \ldots, x_n)\ R_s\ f^A(y_1, \ldots, y_n) \qquad \textit{falls } x_i\ R_{s_i}\ y_i \quad \textit{für alle } i = 1, \ldots, n.$$

Man sagt auch, dass in diesem Fall alle Operationen von A *die Äquivalenzrelationen* $(R_s)_{s \in S}$ respektieren. $\Diamond$

Wenn nun eine Kongruenz R auf einer Algebra A gegeben ist, so können wir eine *Faktoralgebra* wie folgt definieren.

Definition 2.4 *Sei* R *eine* Σ*-Kongruenz auf der* Σ*–Algebra* A*, dann ist die* Faktoralgebra A/R *wie folgt definiert:*

i) *Trägermengen sind* $(A/R)_s := A_s/R_s$ *für jedes* $s \in S$

ii) *Für* $f : s_1 \times \ldots \times s_n \to s$ *ist*

$$f^{A/R} : (A/R)_{s_1} \times \ldots \times (A/R)_{s_n} \to (A/R)_s$$

definiert als

$$f^{A/R} : ([x_1]_{R_{s_1}}, \ldots, [x_n]_{R_{s_n}}) \mapsto [f^A(x_1, \ldots, x_n)]_{R_s} \quad .$$ $\Diamond$

Man beachte, dass $f^{A/R}$ wohldefiniert ist, also nicht von den Repräsentanten x_i der Äquivalenzklassen $[x_i]_{R_{s_i}}$ abhängt, da R von f respektiert wird. Würde man einen anderen Repräsentanten y_i wählen, wäre also $[x_i]_{R_{s_i}} = [y_i]_{R_{s_i}}$ für $i = 1, \ldots, n$, dann gilt auch $x_i\ R_{s_i}\ y_i$ und somit $f^A(x_1, \ldots, x_n)\ R_s\ f^A(y_1, \ldots, y_n)$. Das bedeutet $[f^A(x_1, \ldots, x_n)]_{R_s} = [f^A(y_1, \ldots, y_n)]_{R_s}$ und somit ist der Funktionswert identisch, da dann $f^{A/R}([x_1]_{R_{s_1}}, \ldots, [x_n]_{R_{s_n}}) = f^{A/R}([y_1]_{R_{s_1}}, \ldots, [y_n]_{R_{s_n}})$.

Wiederum gilt in manchen Fällen, dass R von allen $f \in F$ respektiert wird, falls R von allen $f \in F'$ für ein bestimmtes $F' \subseteq F$ respektiert wird. Beispielsweise ist eine Monoidkongruenz auf einer Gruppe ganz automatisch auch eine Gruppenkongruenz.

Wir beweisen nun einige Sätze über die bisher definierten Begriffe und erläutern, wie sie zusammenhängen.

Theorem 2.5 *Sei* $\Sigma = (S, F, \mathrm{ar})$ *eine Signatur und* A *eine* Σ*–Algebra. Die Menge* $\mathrm{Sub}_\Sigma(A)$ *der Unter-*Σ*–Algebren von* A *ist unter beliebigen Durchschnitten abgeschlossen. Also enthält* A *eine kleinste Unteralgebra* A°*.*

Beweis: Sei $(B_i)_{i\in I}$ eine Familie in $\mathrm{Sub}_\Sigma(A)$. Der Durchschnitt $B = \bigcap_{i\in I} B_i$ ist die Unteralgebra von A mit Trägermenge

$$B_s = \{a \in A_s \mid \forall i \in I.\, a \in (B_i)_s\} = \bigcap_{i\in I} (B_i)_s$$

für jedes $s \in S$. Um zu zeigen, dass B auch tatsächlich eine Unteralgebra von A ist, müssen wir zeigen, dass B unter den Operationen von A abgeschlossen ist. Sei $f : s_1 \times \ldots \times s_n \to s$ und $a_j \in B_{s_j}$ für $j = 1, \ldots, n$. Wir zeigen, dass $f^A(a_1, \ldots, a_n) \in B_s$. Betrachte ein beliebiges $i \in I$. Dann ist $a_j \in (B_i)_{s_j}$, da es ja auch im Durchschnitt liegt, und zwar für alle $j = 1, \ldots, n$. Weil nun B_i eine Unteralgebra von A ist, gilt auch $f^A(a_1, \ldots, a_n) \in (B_i)_s$. Deshalb liegt dieser Wert auch im Durchschnitt aller $(B_i)_s$, also $f^A(a_1, \ldots, a_n) \in B_s$.

Bildet man den Durchschnitt *aller* Unteralgebren von A, erhält man A°. □

Bemerkung: Die Unteralgebra A° besteht aus denjenigen Elementen von A, die durch einen variablenfreien Σ-Term ausdrückbar sind. Solche Objekte nennen wir *denotierbar*. Denotierbare Elemente von A sind in jeder Unteralgebra von A enthalten. Andererseits sind die denotierbaren Elemente von A unter den Operationen von A abgeschlossen: sei $f : s_1 \times \ldots \times s_n \to s$ und $a_i \in A_i$ durch den Term t_i denotierbar für $i = 1, \ldots, n$, dann ist $f^A(a_1, \ldots, a_n)$ durch den Term $f(t_1, \ldots, t_n)$ denotierbar.

Diese Beobachtung gibt Anlass zu folgender Definition.

Definition 2.6 *Eine* Σ*-Algebra* A *heißt* termerzeugt *genau dann, wenn für jede Unteralgebra* B *von* A *gilt, dass* $A = B$*.* ◊

In der Mathematik untersucht man üblicherweise Algebren, die nicht termerzeugt sind, wie etwa die Körper $\mathbb{R}$ und $\mathbb{C}$ der reellen bzw. komplexen Zahlen oder Vektorräume über diesen Körpern. In der Informatik hat man eher das Bedürfnis, dass die betrachteten Algebren ausschließlich Objekte enthalten, die man sukzessiv aus den Grundoperationen in endlich vielen Schritten aufbauen kann, d.h. die durch Terme denotierbar sind. Man lebt also nach der „formalistischen“ Devise

> „Was man nicht hinschreiben kann, das gibt es nicht!“

Dieses Bedürfnis nach Konkretheit stößt allerdings an seine Grenzen, wenn man Strukturen wie

$$\mathbb{R},\ \mathbb{N}^{\mathbb{N}},\ \mathcal{P}(\mathbb{N}),\ \mathbb{R}^{\mathbb{N}},\ \mathbb{N}^{(\mathbb{N}^{\mathbb{N}})}\ldots$$

betrachtet. Hier verkneifen wir uns allerdings, diese interessanten und provokativen Fragen an dieser Stelle weiter zu verfolgen.

Die Annahme, dass eine Σ–Algebra termerzeugt ist, hat unter anderem den Vorteil, dass auf sie folgendes *Induktionsprinzip* anwendbar ist:

Theorem 2.7 *Sei A eine termerzeugte Σ–Algebra und $B = (B_s)_{s\in S}$ eine Familie von Teilmengen $B_s \subseteq A_s$ der Trägermengen von A. Wenn B unter den Operationen von A abgeschlossen ist, dann gilt $B_s = A_s$ für alle $s \in S$.*

Beweis: Wenn B unter den Operationen von A abgeschlossen ist, dann ist B eine Unteralgebra von A. Wegen der Termerzeugtheit von A ist dann $B = A$. □

Theorem 2.8 *Sei wieder A eine Σ–Algebra und $B = (B_s)_{s\in S}$ eine Familie von Teilmengen $B_s \subseteq A_s$ der Trägermengen von A. Dann existiert eine kleinste Unteralgebra $\langle B\rangle$ von A, die B enthält. Die Unteralgebra $\langle B\rangle$ heißt die von B* erzeugte *Unteralgebra von A.*

Beweis: Wegen Satz 2.5 ist der Durchschnitt aller Unteralgebren von A, die B enthalten, wieder eine Unteralgebra von A und enthält außerdem B. Also ist dieser Durchschnitt die kleinste Unteralgebra von A, die B enthält. □

Bemerkung. Die Elemente von $\langle B\rangle$ sind diejenigen Elemente aus A, die mit Hilfe der Operationen von A aus den Elementen von B in endlich vielen Schritten erzeugt werden können. Die Unteralgebra $\langle B\rangle$ enthält also genau diejenigen Elemente von A, die durch Terme denotierbar sind, die aus Operationen von Σ und zusätzlichen *Konstanten für die Elemente aus B* aufgebaut sind.

Wenn A termerzeugt ist, dann ist $A = \langle B\rangle$ für alle Teilmengen B von A (selbst wenn die Teilmengen B_s für alle Sorten $s \in S$ leer sind).

Als nächstes studieren wir den Begriff der Isomorphie allgemeiner Algebren.

Definition 2.9 *Seien A und B Σ–Algebren. Wir nennen einen Σ–Homomorphismus $h : A \to B$ einen* Σ–Isomorphismus *oder kurz Isomorphismus genau dann, wenn ein Σ-Homomorphismus $k : B \to A$ existiert, so dass für alle Sorten s gilt:*

$$k_s \circ h_s = \mathrm{id}_{A_s} \qquad h_s \circ k_s = \mathrm{id}_{B_s}$$

oder kürzer formuliert: $k \circ h = \mathrm{id}_A$ und $h \circ k = \mathrm{id}_B$. ◇

Offenbar ist k durch h eindeutig bestimmt, da $k_s = h_s^{-1}$ für alle Sorten s. Die Bedingung, die in obiger Definition gefordert wird, ist also, dass h^{-1} ein Homomorphismus ist. Dies ist jedoch immer der Fall, sobald h_s bijektiv ist.

Lemma 2.10 *Ein Homomorphismus $h : A \to B$ von Σ–Algebren ist ein Isomorphismus genau dann, wenn für alle Sorten s die Funktion $h_s : A_s \to B_s$ eine Bijektion ist.*

Beweis: Übung! $\square$

Als nächstes studieren wir den Zusammenhang zwischen Homomorphismen und Kongruenzrelationen.

Theorem 2.11 *Sei $h : A \to B$ ein Σ-Homomorphismus von Σ–Algebren. Dann ist die Familie $(\ker(h)_s)_{s\in S}$ mit*

$$\ker(h)_s = \ker(h_s) = \{(x,y) \mid h_s(x) = h_s(y)\} \subseteq A_s \times A_s \qquad (s \in S)$$

eine Kongruenz auf A.

Beweis: Wir zeigen, dass die Operationen von A die Relation $\ker(h)$ respektiert. Sei $f : s_1 \times \ldots \times s_n \to s$ und $a_i \ker(h)_{s_i}\, b_i$ für $i = 1, \ldots, n$. Wir zeigen nun, dass $f^A(a_1, \ldots, a_n) \ker(h)_s\, f^A(b_1, \ldots, b_n)$:

$$\begin{array}{lll} & h_s(f^A(a_1,\ldots,a_n)) & (h \text{ Homomorphismus}) \\ = & f^B(h_{s_1}(a_1),\ldots,h_{s_n}(a_n)) & (a_i \ker(h)_{s_i}\, b_i) \\ = & f^B(h_{s_1}(b_1),\ldots,h_{s_n}(b_n)) & (h \text{ Homomorphismus}) \\ = & h_s(f^A(b_1,\ldots,b_n)) & \end{array}$$

da ja nach Definition $a_i \ker(h)_{s_i}\, b_i$ bedeutet, dass $h_{s_i}(a_i) = h_{s_i}(b_i)$. $\square$

Umgekehrt induziert jede Kongruenzrelation R auf A einen Homomorphismus von A nach der Faktoralgebra A/R.

Theorem 2.12 *Sei R eine Kongruenzrelation auf der Σ-Algebra A. Die Familie* $\text{quot}_R = (\text{quot}_{R_s})_{s\in S}$ *von Abbildungen*

$$\text{quot}_{R_s} : A_s \to (A/R)_s \qquad \textit{mit} \qquad (\text{quot}_{R_s}(x) = [x]_{R_s}$$

ist ein Homomorphismus von A nach A/R.

Beweis: Sei $f : s_1 \times \ldots \times s_n \to s$ eine Operation aus Σ. Dann gilt

$$\begin{array}{rcl} \text{quot}_{R_s}(f^A(a_1,\ldots,a_n)) & = & [f^A(a_1,\ldots,a_n)]_{R_s} = f^{A/R}([x_1]_{R_{s_1}},\ldots,[x_n]_{R_{s_n}}) \\ & = & f^{A/R}(\text{quot}_{R_{s_1}}(a_1),\ldots,\text{quot}_{R_{s_n}}(a_n)) \end{array}$$

für alle $a_i \in A_{s_i}$ (wobei $i = 1, \ldots, n$), womit gezeigt ist, dass quot_R ein Σ–Homomorphismus ist. $\square$

Unser Ziel ist es zu zeigen, dass jeder Homomorphismus sich darstellen lässt als die Hintereinanderausführung eines surjektiven Homomorphismus gefolgt von einem injektiven Homomorphismus. Daraus folgt dann, dass A/Ker_h isomorph zu $h[A]$, dem Bild von A unter h ist.

Zuerst zeigen wir, dass das Bild einer Algebra unter einem Homomorphismus wiederum eine Algebra ist.

Lemma 2.13 *Sei $h : A \to B$ ein Σ–Homomorphismus. Dann ist die Familie*

$$h[A]_s := h_s[A_s] = \{h_s(x) \mid x \in A_s\} \qquad (s \in S)$$

eine Unteralgebra von B.

Beweis: Wir zeigen, dass $h[A]$ eine Unteralgebra von B ist, d.h., dass $h[A]$ unter den Operationen von B abgeschlossen ist. Sei $f : s_1 \times \cdots \times s_n \to s$ eine Operation in Σ und $h_{s_i}(a_i) \in h[A]_{s_i}$ für $i = 1, \ldots, n$. Dann ist aber

$$f^B(h_{s_1}(a_1), \ldots, h_{s_n}(a_n)) = h_s(f^A(a_1, \ldots, a_n)) \in h[A]_s$$

da h voraussetzungsgemäß ein Σ-Homomorphismus ist. $\square$

Theorem 2.14 *Sei $h : A \to B$ ein Homomorphismus von Σ–Algebren. Dann gibt es einen eindeutig bestimmten Σ–Isomorphismus $i_h : A_{\mathrm{ker}(h)} \to h[A]$, so dass*

$$\begin{tikzcd}
A \arrow[r, "h"] \arrow[d, "\mathrm{quot}_{\ker h} = e_h"', "surj."] & B \\
A/\mathrm{ker}(h) \arrow[r, "\cong", "i_h"'] & h[A] \arrow[u, hook, "inj."', "m_h"]
\end{tikzcd}$$

also $m_h \circ i_h \circ e_h = h$, wobei m_h die Inklusion von $h[A]$ in B ist. Der Isomorphismus i_h ist explizit gegeben durch

$$(i_h)_s([x]_{\mathrm{ker}(h)_s}) = h_s(x) \qquad (s \in S)$$

Beweis: Die Funktion i_h ist eindeutig bestimmt, weil $e_h = \mathrm{quot}_{\mathrm{ker}(h)}$ surjektiv (in jeder Komponente) und m_h injektiv (in jeder Komponente) ist. Außerdem ist i_h wohldefiniert, weil

$$x \ \mathrm{ker}(h)_s \ x' \Rightarrow h_s(x) = h_s(x')$$

Wir zeigen $m_h \circ i_h \circ e_h = h$. Sei dazu $x \in A_s$, dann gilt

$$
\begin{array}{rll}
& (m_h \circ i_h \circ e_h)_s(x) & \\
= & (m_h)_s((i_h \circ e_h)_s\,(x)) & \text{(Da } m \text{ Inklusion)} \\
= & (i_h \circ e_h)_s(x) = (i_h)_s((e_h)_s(x)) = (i_h)_s([x]_{\ker(h)_s}) & \\
= & h_s\,(x) & \text{(Nach Def. von i)}
\end{array}
$$

Wir zeigen weiter, dass i_h ein Homomorphismus ist. Sei $f : s_1 \times \ldots \times s_n \to s$ und $[x_i]_{\ker(h)_{s_i}} \in (A/\ker_h)_{s_i}$ für $i = 1, \ldots, n$; dann gilt

$$
\begin{array}{rl}
& (i_h)_s\big(f^{A/\ker(h)}([x_1]_{\ker(h)_{s_1}},\ \ldots,\ [x_n]_{\ker(h)_{s_n}})\big) \\
= & (i_h)_s\big([f^A(x_1,\ \ldots,\ x_n)]_{\ker(h)_s}\big) \\
= & h_s\big(f^A(x_1,\ \ldots,\ x_n)\big) \\
= & f^B\big(h_{s_1}(x_1),\ \ldots,\ h_{s_n}(x_n)\big) \\
= & f^B\big((i_h)_{s_1}([x_1]_{\ker(h)_{s_1}}),\ \ldots,\ (i_h)_{s_n}([x_n]_{\ker(h)_{s_n}})\big)
\end{array}
$$

Es bleibt zu zeigen, dass i_h komponentenweise bijektiv ist. Sei s eine Sorte und $y \in h[A]_s$. Dann ist $y = h_s(x)$ für ein $x \in A_s$, und somit $y = h_s(x) = (i_h)_s([x]_{\ker(h)_s})$. Also ist $(i_h)_s$ surjektiv.
Sind weiter $[x]_{\ker(h)_s}, [x']_{\ker(h)_s} \in (A/\ker(h))_s$ mit $(i_h)_s([x]_{\ker(h)_s}) = (i_h)_s\,([x']_{\ker(h)_s})$, dann folgt $h_s(x) = h_s(x')$ auf Grund der Definition von i_h. Somit ist $x \ker(h)_s\, x'$, d.h. $[x]_{\ker(h)_s} = [x']_{\ker(h)_s}$. Also ist $(i_h)_s$ auch injektiv. □

Aus diesem Satz ergeben sich einige wichtige Konsequenzen:

- Unteralgebren von B entsprechen *bis auf Isomorphie* injektiven Homomorphismen nach B
- Kongruenzrelationen auf A entsprechen bis auf Isomorphie surjektiven Homomorphismen.

Denn ist $h : A \to B$ ein Homomorphismus, dann gilt (Beweis als Übung!)

i) h ist injektiv genau dann, wenn $e_h = \mathrm{quot}_{\ker(h)}$ ein Isomorphismus ist;

ii) h ist surjektiv genau dann, wenn m_h ein Isomorphismus ist.

Abschließend zeigen wir, dass nicht nur die Unteralgebren einer Algebra A einen vollständigen Verband bzgl. der komponentenweisen Mengeninklusion bilden, sondern auch die Kongruenzen auf A.

Theorem 2.15 *Sei A eine Σ-Algebra. Wenn $(R_i \mid i \in I)$ eine Familie von Kongruenzen auf A ist, so ist auch ihr komponentenweiser Durchschnitt*

$$\Big(\bigcap_{i\in I} R_i\Big)_s = \bigcap_{i\in I} (R_i)_s$$

eine Kongruenz auf A.

Wenn ferner $(E_s \subseteq A_s \times A_s \mid s \in S)$ *eine Familie beliebiger Relationen ist, so existiert eine kleinste Kongruenz* R *auf* A*, die* E *enthält (also* $E_s \subseteq R_s$ *für alle* $s \in S$*). Diese kleinste* E *enthaltende Kongruenz wird als* $\mathrm{Cong}_A(E)$ *bezeichnet und ist folgendermaßen definiert:*

$$\mathrm{Cong}_A(E)_s = \bigcap \{R_s \mid E \subseteq R, R \textit{ ist Kongruenz auf } A\}$$

Beweis: Sei $(R_i \mid i \in I)$ eine Familie von Kongruenzen auf A. Wir zeigen, dass der Durchschnitt $\overline{R}_s := \bigcap_{i \in I}(R_i)_s$ wieder eine Kongruenzrelation ist.
Weil der Durchschnitt von Äquivalenzrelationen wieder eine solche ist, ist $\overline{R}_s$ eine Äquivalenzrelation.
Für die Kongruenzbedingung sei $f : s_1 \times \ldots \times s_n \to s$ eine Σ-Operation. Wir zeigen, dass f die $\overline{R}_s$ alle respektiert.
Gelte dazu $x_j\ \overline{R}_{s_j}\ y_j$ für $j = 1, \ldots, n$. Das bedeutet, für alle $i \in I$, $j \in \{1, \ldots, n\}$ gilt $x_j\ (R_i)_{s_j}\ y_j$. Also folgt für alle $i \in I$, dass $f^A(x_1, \ldots x_n)\ (R_i)_s\ f^A(y_1, \ldots y_n)$, weil alle R_i selbst Kongruenzen sind. Daher gilt nun $f^A(x_1, \ldots x_n)\ \overline{R}_s\ f^A(y_1, \ldots y_n)$.
Somit ist $\overline{R}$ tatsächlich eine Kongruenz auf A.
Der Durchschnitt *aller* Kongruenzen R_i, die eine vorgegebene Relationenfamilie E enthalten, ist also ebenfalls wieder eine Kongruenz und enthält E. □

Bemerkung: Die kleinste Kongruenz auf A ist die (komponentenweise) Gleichheitsrelation (auch *Diagonale* Δ genannt). Die größte Kongruenz auf A ist diejenige Kongruenz R mit $R_s = A_s \times A_s$ für alle $s \in S$ (auch *Allrelation* ∇ genannt).

Für Kongruenzenrelationen R und R' auf einer Algebra A mit $R \subseteq R'$ gibt es einen eindeutig bestimmten Homomorphismus $h : A/R \to A/R'$, so dass

$$h \circ \mathrm{quot}_R = \mathrm{quot}_{R'}$$

d.h.

$$\begin{tikzcd} & A \arrow[dl, "\mathrm{quot}_R"'] \arrow[dr, "\mathrm{quot}_{R'}"] & \\ A/R \arrow[rr, "h"'] & & A/R' \end{tikzcd}$$

wobei $h_s([x]_{R_s}) = [x]_{R'_s}$ für $s \in S$.

Man beachte, dass nicht alle $h : A/R \to A/R'$ von dieser Form sind, etwa wenn $R = R'$ die kleinste Kongruenz Δ auf A und h irgendein von id_A verschiedener Isomorphismus ist. (Das ist möglich, da in diesem Fall $A/\Delta \cong A$, und daher alle Endomorphismen von A auch auf den Faktoralgebren Sinn machen.)

Im nächsten Abschnitt werden wir allerdings sehen, dass für beliebige Homomorphismen $h : A/R \to A/R'$ gilt $\mathrm{quot}_{R'} = h \circ \mathrm{quot}_R$, falls A termerzeugt ist.

Übung: Erkläre die Konstruktion von $\mathrm{Cong}_A(E)$ als induktive Definition (durch Angabe einer erzeugenden Regelmenge).

II.3 Produkte und Egalisatoren

Die Begriffe „Produkt“ und „Egalisator“ (engl. equalizer) erfahren ihre allgemeine Erklärung im Rahmen der so genannten Kategorientheorie. (Eine Kategorie ist eine mathematische Präzisierung und Verallgemeinerung der Situation, dass eine Klasse von Strukturen und Homomorphismen dazwischen gegeben ist).
Wir studieren diese Begriffe in ihrer konkreten Erscheinungsform in der „Kategorie der Σ-Algebren und Σ- Homomorphismen“.

Definition 3.1 *Sei $(A_i \mid i \in I)$ eine Familie von Σ-Algebren. Die Σ-Algebra $\Pi_{i\in I} A_i$, das „Produkt der A_i“, sei folgendermaßen definiert: Trägermengen sind*

$$\left(\Pi_{i\in I} A_i\right)_s = \Pi_{i\in I}(A_i)_s \qquad (s \in S)$$

Formal kann man die Tupel im Produkt auffassen als Funktionen und entsprechend schreiben $(\Pi_{i\in I} A_i)_s = \{a : I \to \bigcup_{i\in I} (A_i)_s \mid \forall i \in I.\, a_i \in (A_i)_s\}$. Deswegen schreiben wir auch $a(i)$ für die i-te Komponente des Tupels, wenn $a \in \Pi_{i\in I} A_i$ ist.

Die Operationen bekommt man so: Für $f : s_1 \times \ldots \times s_n \to s$ in F sei

$$f^{\Pi_{i\in I} A_i}(a_1, \ldots a_n)(i) = f^{A_i}((a_1)_i, \ldots, (a_n)_i)$$

wobei $a_j \in (\Pi_{i\in I} A_i)_{s_j}$, für $j = 1, \ldots, n$, und $i \in I$. $\diamondsuit$

Theorem 3.2 *Sei $(A_i \mid i \in I)$ eine Familie von Σ-Algebren. Für $i \in I$ und $s \in S$ sei die i-te* Projektion *π_i die Funktionenfamilie*

$$(\pi_i)_s : (\Pi_{i\in I} A_i)_s \to (A_i)_s$$

die a auf die i-te Komponente a_i abbildet.
Diese S-indizierte Familie $\pi_i = ((\pi_i)_s \mid s \in S)$ ist ein Σ-Homomorphismus

$$\pi_i : \Pi_{i\in I} A_i \to A_i.$$

Ist ferner $(h_i : B \to A_i \mid i \in I)$ eine Familie von Σ-Homomorphismen, dann gibt es genau einen Σ- Homomorphismus $\overline{h} : B \to \Pi_{i\in I} A_i$, so dass für alle $i \in I$ gilt $\pi_i \circ \overline{h} = h_i$ (also konkret für alle $s \in S$, dass $(\pi_i) \circ \overline{h}_s = (h_i)_s$).

Beweis: Wir verwenden als Abkürzung $\overline{A} := \Pi_{i\in I} A_i$. Sei $i \in I$. Wir zeigen, dass $\pi_i : A \to A_i$ ein Homomorphismus ist. Für $f \in F$ mit $f : s_1 \times \ldots \times s_n \to s$ und $a_j \in \overline{A}_{s_j}$ mit $j = 1, \ldots, n$ gilt

$$\begin{aligned}
& (\pi_i)_s(f^{\overline{A}}(a_1, \ldots, a_n)) \\
= \; & f^{\overline{A}}(a_1, \ldots, a_n)(i) && \text{(Nach Def. von } f^{\overline{A}}\text{)} \\
= \; & f^{A_i}((a_1)_i, \ldots, (a_n)_i) \\
= \; & f^{A_i}((\pi_i)_{s_1}(a_1), \ldots, (\pi_i)_{s_n}(a_n))
\end{aligned}$$

Sei nun $(h_i : B \to A_i \mid i \in I)$ eine Familie von Homomorphismen. Wenn aber $\overline{h} : B \to \overline{A}$ ein Homomorphismus ist, der $\pi_i \circ \overline{h} = h_i$ für alle $i \in I$ erfüllt, dann gilt für alle $s \in S$ und $b \in B_s$, dass

$$(\overline{h}_s(b))_i = (\pi_i)_s(\overline{h}_s(b)) = (\pi_i \circ \overline{h})_s(b) = (h_i)_s(b)$$

Also muss $\overline{h} : B \to \overline{A}$, wenn es überhaupt als Homomorphismus existiert, folgendermaßen aussehen:

$$(\overline{h}_s\,(b))_i = (h_i)_s(b) \qquad \text{für } i \in I,\ s \in S \text{ und } b \in B_s$$

Nimmt man dies als Definition einer entsprechenden Familie, hat man die Eindeutigkeit schon gezeigt und muss nur noch nachrechnen, dass es wirklich ein Homomorphismus von B nach $\overline{A}$ ist. Das verbleibt als Übung.
Aus der Definition von $\overline{h}$ folgt sofort, dass für $i \in I$ tatsächlich gilt $\pi_i \circ \overline{h} = h_i$ wie gefordert war. $\square$

Bemerkung Die Aussage von 3.2 lässt sich folgendermaßen veranschaulichen:

$$\begin{tikzcd}
B \arrow[d, "\overline{h}"'] \arrow[dr, "h_i"] & \\
\overline{A} \arrow[r, "\pi_i"'] & A_i
\end{tikzcd} \qquad (i \in I)$$

Wenn $I = \emptyset$, dann ist $\Pi_{i \in \emptyset} A_i$ die triviale Algebra, in der jede Sorte durch die einelementige Menge $\{\emptyset\}$ interpretiert wird.

Theorem 3.3 (Egalisatoren) *Seien $h_1, h_2 : A \to B$ zwei Homomorphismen von $\Sigma-$ Algebren. Die Familie $(E_s \subseteq A_s \mid s \in S)$ mit*

$$E_s = \{a \in A_s \mid (h_1)_s(a) = (h_2)_s(a)\}$$

ist eine Unteralgebra von A.
Sei weiter $\iota_s : E_s \hookrightarrow A_s$ die Inklusion von E_s in A_s, d.h. $\iota_s(a) = a$ für $a \in E_s$.
Dann ist die Familie $\iota = (\iota_s \mid s \in S)$ ein Homomorphismus von E nach A.
Ist schließlich $k : C \to A$ ein Homomorphismus mit

$$h_1 \circ k = h_2 \circ k \quad ,$$

dann gibt es einen eindeutigen Homomorphismus $\overline{k} : C \to E$, so dass

$$\iota \circ \overline{k} = k \quad .$$

Mit anderen Worten, das Bild von k ist schon in E enthalten.

Beweis: Wir zeigen, dass E unter den Operationen von A abgeschlossen ist. Sei also $f : s_1 \times \ldots \times s_n \to s$ eine Operation und $a_i \in E_{s_i}$ für $i = 1, \ldots, n$. Dann gilt

$$\begin{array}{lll} & (h_1)_s\,(f^A(a_1,\,\ldots,\,a_n)) & \\ = & f^B((h_1)_{s_1}(a_1),\,\ldots,\,(h_1)_{s_n}(a_n)) & \qquad (\text{da } a_i \in E_{s_i}) \\ = & f^B((h_2)_{s_1}(a_1),\,\ldots,\,(h_2)_{s_n}(a_n)) & \\ = & (h_2)_s(f^A(a_1,\ldots a_n)) & \end{array}$$

also $f^A(a_1, \ldots a_n) \in E_s$. Somit ist E Unteralgebra von A.

Die Familie $\iota = (\iota_s : E_s \hookrightarrow A_s \mid s \in S)$ ist ein injektiver Homomorphismus, weil das für Unteralgebren allgemein gilt.

Ist schließlich $h_1 \circ k = h_2 \circ k$, dann ist das Bild von k in E enthalten (da alle $k_s(c)$ die Egalisatorbedingung erfüllen). Man kann also für $c \in C_s$ definieren

$$\overline{k}_s\,(c) = k_s(c)$$

für jedes $s \in S$, denn da $(h_1)_s\,(k_s(c)) = (h_1)_s(k_s(c))$ ist wegen der Annahme $h_1 \circ k = h_2 \circ k$, liegt $k_s(c) \in E_s \subseteq A_s$ wie gefordert.
Weiter ist $\overline{k}$ ist ein Homomorphismus, weil k einer ist, und $\overline{k}$ ist eindeutig mit der Eigenschaft $\iota \circ \overline{k} = k$, weil ι injektiv ist. □

Der Name „Egalisator“ (engl. equalizer) rührt daher, dass man für zwei Funktionen bzw. Homomorphismen die größte Teilmenge ihres Definitionsbereiches betrachten kann, in dem diese gleich sind (oder „egalisiert“ werden).
So ist zum Beispiel der Egalisator zweier polynominaler Funktionen $p, q : \mathbb{C}^n \to \mathbb{C}$ identisch mit der *Lösungsmenge* der Gleichung $p(\vec{x}) = q(\vec{x})$.

Satz 3.3 hat folgende wichtige Konsequenz:

Theorem 3.4 *Seien $h_1, h_2 : A \to B$ Homomorphismen. Wenn A termerzeugt ist, dann gilt $h_1 = h_2$.*

Beweis: Aufgrund von 3.3 ist der Egalisator E von h_1 und h_2 eine Unteralgebra von A. Wegen der Termerzeugtheit von A ist daher $E = A$ und somit $h_1 = h_2$. □

Von einer termerzeugten Algebra A gibt es also für jede Algebra B höchstens einen Homomorphismus $A \to B$. Wenn B selbst termerzeugt ist, ist dieser Homomorphismus surjektiv, weil sein Bild eine Unteralgebra von B und somit gleich B ist. Wenn also $h : A \to B$ ein Homomorphismus zwischen termerzeugten Algebren ist, so ist B isomorph zu $A/\ker_h$. Damit entsprechen Homomorphismen zwischen termerzeugten Algebren den Faktorisierungen.

Aus diesen Beobachtungen ergibt sich folgender Satz.

Theorem 3.5 *Sei Σ eine Signatur, und sei A eine* initiale termerzeugte Σ-*Algebra (d.h. A ist termerzeugt, und für jede termerzeugte Σ-Algebra gibt es einen (eindeutigen) Homomorphismus $A \to B$).*
Dann ist jede termerzeugte Σ-Algebra B isomorph zu A/R für eine eindeutig bestimmte Kongruenz R auf A.

Beweis: Sei B eine termerzeugte Σ-Algebra. Setze $R = \mathrm{Ker}_h$, wobei $h : A \to B$ der nach obigem Satz eindeutige Homomorphismus ist. Dann folgt die Isomorphie aus der Faktorisierung nach Satz 2.14, wenn noch gezeigt werden kann, dass die Inklusion m_h surjektiv ist. Das gilt aber, denn da $h[A]$ eine Unteralgebra von B ist und da B termerzeugt ist, muss schon $h[A] = B$ sein. Daher sind m_h und i_h Isomorphismen, deren Verkettung den gewünschten Isomorphismus liefert. $\square$

Ein Beispiel für eine solche initiale termerzeugte Algebra für die Signatur

NAT	$\equiv$
sorts	nat
funcs	zero :$\to$ nat succ : nat $\to$ nat
axioms	

ist $(\mathbb{N}, 0, n \mapsto n{+}1)$.

Übung: Sei Σ eine Signatur, und A eine Σ-Algebra. Zeige: A ist eine initiale termerzeugte Algebra genau dann, wenn es für jede Σ-Algebra einen eindeutigen Homomorphismus $A \to B$ gibt.

II.4 Freie Algebren und Interpretationen von Termen

In diesem Abschnitt werden wir Algebren konstruieren, deren Trägermengen aus syntaktischen Objekten, sogenannten *Termen*, bestehen. Dies ist insofern ungewöhnlich, als man üblicherweise Terme dazu verwendet, um Elemente aus (den Trägermengen von) Algebren zu bezeichnen. Normalerweise ist ein Zeichen (z.B. ein Term) etwas völlig anderes als das von ihm bezeichnete Objekt (z.B. ein Element einer Algebra). Wenn man beispielsweise das Wort „Tisch“ betrachtet, so ist es selbst einfach eine Kette von Buchstaben (oder eine Lautfolge). Der von ihr bezeichnete Gegenstand (in einem speziellen Kontext) hingegen ist ein ganz anders geartetes Objekt als diese Buchstaben- oder Lautfolge. Eben dieser Unterschied zwischen Zeichen und Bezeichneten bewirkt die Ökonomie der natürlichen oder künstlichen Sprachen.

Da die Gegenstände, über die die Mathematik und die Informatik sprechen, selber oft „sprachlicher“ Natur sind (zumindest nach Auffassung der so genannten „Formalisten“) ist es vielleicht nicht weiter verwunderlich, dass mathematischen Objekte selbst als Äquivalenzklassen derjeniger sprachlicher Zeichen aufgefasst werden können, die das vorgegebene Objekt bezeichnen.
Im Falle der Zahlbereiche $\mathbb{N}, \mathbb{Z}, \mathbb{Q}$ ist dies ganz offensichtlich der Fall. Gänzlich anders gelagert ist der Sachverhalt bei Strukturen wie $\mathbb{R}$, $\mathbb{N}^{\mathbb{N}}$, $\mathcal{P}(\mathbb{N})$ etc., wo es zu Widersprüchen führte, wenn man annähme, dass *alle* Objekte sprachlich ausdrückbar wären. Man braucht also gewisse (hypothetische?) „ideale Objekte“, um über die eigentlich interessierenden konkreten Objekte in abstrakter und eleganter Weise sprechen zu können.

Vorerst wollen wir uns aber unproblematischeren Dingen zuwenden und den Begriff „Term“, der bisher informell verwendet wurde, mathematisch präzise definieren.

Für den Rest dieses Abschnitts sei $\Sigma = (S, F, \mathrm{ar})$ eine beliebige Signatur.

Definition 4.1 *Sei $\mathcal{V} = (\mathcal{V}_s \mid s \in S)$ eine Familie von Mengen (die als Mengen von* Variablen *aufgefasst werden). Die Familie*

$$T_\Sigma(\mathcal{V}) = (T_\Sigma(\mathcal{V})_s \mid s \in S)$$

von „Σ–Termen über Variablen aus $\mathcal{V}$“ ist dann induktiv wie folgt definiert:

i) *Für jede Variable $x \in \mathcal{V}_s$ ist auch $x \in T_\Sigma(\mathcal{V})_s$.*
Anders ausgedrückt: $\mathcal{V}_s \subseteq T_\Sigma(\mathcal{V})_s$ für alle $s \in S$ oder einfach $\mathcal{V} \subseteq T_\Sigma(\mathcal{V})$.

ii) *Wenn $f : s_1 \times \ldots \times s_n \to s$ in F und $t_i \in T_\Sigma(\mathcal{V})_{s_i}$ für $i = 1, \ldots, n$, dann ist auch*

$$f(t_1, \ldots, t_n) \in T_\Sigma(\mathcal{V})_s\,.$$

Die Familie $T_\Sigma(\mathcal{V})$ trägt überdies die Struktur einer Σ–Algebra, wenn man jedes Funktionssymbol $f : s_1 \times \ldots \times s_n \to s$ durch eine Funktion

$$f^{T_\Sigma(V)} : T_\Sigma(\mathcal{V})_{s_1} \times \ldots \times T_\Sigma(\mathcal{V})_{s_n} \to T_\Sigma(\mathcal{V})_s$$

in folgender Weise interpretiert:

$$f^{T_\Sigma(V)}(t_1, \ldots, t_n) := f(t_1, \ldots t_n) \in T_\Sigma(\mathcal{V})_s$$

wobei $t_i \in T_\Sigma(\mathcal{V})_{s_i}$ für $i = 1, \ldots, n$. Diese Algebra heißt auch „freie Σ-Algebra über $\mathcal{V}$“ und wird (ebenfalls) mit $T_\Sigma(\mathcal{V})$ bezeichnet. $\Diamond$

Bemerkung: Man beachte, dass in obiger Definition $f^{T_\Sigma(\mathcal{V})}$ und f definitiv verschieden sind, da $f^{T_\Sigma(\mathcal{V})}$ ist eine *Funktion* ist, die Tupel von Termen auf Terme abbildet, wohin gegen f einfach ein *Funktionssymbol* ist.

Definition 4.2 *Sei A eine Σ–Algebra und $\mathcal{V} = (V_s \mid s \in S)$ eine Familie von Mengen (von Variablen). Eine* Variablenbelegung *oder $\mathcal{V}$–Belegung in A ist eine Familie von Funktionen*

$$\rho = (\rho_s : \mathcal{V}_s \to A_s \mid s \in S)$$

die jeder Variablen einer Sorte einen Wert derselben Sorte in A zuordnen. Wir schreiben oft kurz $\rho : \mathcal{V} \to A$. Die Menge der $\mathcal{V}$–Belegungen in A wird auch als $A^{\mathcal{V}}$ bzw. $\mathrm{Val}^A_{\mathcal{V}}$ *bezeichnet.* $\Diamond$

Bemerkung: Belegung heißt auf englisch *valuation.*

Der folgende Satz rechtfertigt die Bezeichnung „freie (Σ)-Algebra über $\mathcal{V}$“ für $T_\Sigma(\mathcal{V})$ (man vergleiche mit dem freien Wortmonoid, Theorem 3.11).

Theorem 4.3 *Sei $\mathcal{V} = (\mathcal{V}_s \mid s \in S)$ eine Familie von Mengen (von Variablen) und A eine Σ–Algebra. Für jede Belegung $\rho : \mathcal{V} \to A$ gibt es einen eindeutig bestimmten Σ–Homomorphismus $h : T_\Sigma(\mathcal{V}) \to A$, der diese Belegung fortsetzt, d.h., dass $h_s(x) = \rho_s(x)$ für alle $x \in \mathcal{V}_s$ gilt , d.h., dass das Diagramm*

$$\begin{tikzcd}
\mathcal{V} \arrow[rr, hook, "\iota"] \arrow[dr, "\rho"'] & & T_\Sigma(\mathcal{V}) \arrow[dl, dashed, "h"] \\
& A &
\end{tikzcd}$$

kommutiert. Dieser eindeutig bestimmte Homomorphismus h wird mit $[\![\_]\!]^A_\rho$ bzw. $\mathrm{Int}^A(\rho)$ *bezeichnet und heißt „die durch ρ induzierte Interpretationsfunktion“.*

Beweis: Da $T_\Sigma(\mathcal{V})$ die kleinste Unteralgebra von $T_\Sigma(\mathcal{V})$ ist, die $\mathcal{V}$ enthält, gilt für zwei beliebige Homomorphismen $h_1, h_2 : T_\Sigma(\mathcal{V}) \to A$, die auf $\mathcal{V}$ übereinstimmen, dass der Egalisator von h_1 und h_2 gleich $T_\Sigma(\mathcal{V})$ ist. Daraus folgt $h_1 = h_2$ und somit die Eindeutigkeit von h bezüglich eines gegebenen $\rho \in \mathrm{Val}^A_{\mathcal{V}}$.

Die Existenz von $h : T_\Sigma(\mathcal{V}) \to A$ mit der gewünschten Fortsetzungseigenschaft zeigen wir dadurch, dass wir $h = [\![\_]\!]_\rho$ durch *strukturelle Rekursion* über $T_\Sigma(\mathcal{V})$ definieren:

i) $[\![x]\!]_\rho = \rho(x)$ für $x \in \mathcal{V}$

ii) $[\![f(t_1, \ldots, t_n)]\!]_\rho = f^A([\![t_1]\!]_\rho, \ldots, [\![t_n]\!]_\rho)$
wobei $t_1, \ldots, t_n$ bereits Terme in $T_\Sigma(\mathcal{V})$ sind.

In der Rekursion ist (i) der Basisfall, und in (ii) wird in der rechten Seite nur auf die Interpretation von „kleineren" früher erzeugten Termen Bezug genommen.

Die Funktion $h = [\![.]\!]^A_\rho$ ist total, weil auch $T_\Sigma(\mathcal{V})$ induktiv definiert wurde. Dass h ein Homomorphismus ist, wird durch (ii) erzwungen, und (i) garantiert die Fortsetzungseigenschaft. $\square$

Bemerkung: Die Σ-Algebra $T_\Sigma(\mathcal{V})$ heißt „frei über $\mathcal{V}$", weil sich *jede* Belegung ρ zu einem Homomorphismus fortsetzen lässt. Es werden auf $\mathcal{V}$ keine weiteren Gleichheiten erzwungen: Variablen $x, y \in \mathcal{V}_s$ sind gleich als Element von $T_\Sigma(\mathcal{V})_s$ genau dann, wenn sie gleich sind als Element von $\mathcal{V}_s$. Ebenso sind zwei Terme $t_1, t_2 \in T_\Sigma(\mathcal{V})_s$ genau dann gleich, wenn sie als Zeichenketten über dem Alphabet

$$F \cup \mathcal{V} \cup \{"(", \ ")", \ ","\}$$

gleich sind.

Als nächstes zeigen wir, dass jede Σ–Algebra isomorph zum Quotienten einer freien Σ–Algebra ist.

Theorem 4.4 *Sei A eine Σ–Algebra. Dann existiert ein Homomorphismus*

$$h : T_\Sigma(A) \to A$$

mit $h_s(a) = a$ für alle $s \in S$ und $a \in A_s$. Damit ist h auch surjektiv, und es gilt

$$A \cong T_\Sigma(A) / \ker(h)$$

Beweis: Der Homomorphismus h existiert wegen Satz 4.3, wenn man als Belegung gerade $\rho(a) = a$ wählt. $\square$

Dieser Satz eröffnet die Möglichkeit, Σ–Algebren bis auf Isomorphie durch die Angabe einer *Generatormenge* (oder genauer gesagt einer Generatormengenfamilie) $G = (G_s \mid s \in S)$ und einer Familie $E = (E_s \subseteq T_\Sigma(G)^2_s \mid s \in S)$ von *Gleichheitsbedingungen* zu beschreiben, nämlich als $T_\Sigma(G)/\langle E \rangle$, den Quotienten von $T_\Sigma(G)$ modulo der von E erzeugten kleinsten Kongruenzrelation R auf $T_\Sigma(G)$ mit $E \subseteq R$ (vgl. Satz 2.15).

Definition 4.5 *Sei G eine S–indizierte Familie von Mengen und*

$$E = (E_s \subseteq T_\Sigma(G)_s^2 \mid s \in S)$$

eine Familie von Gleichheitsbedingungen, dann sei

$$T_\Sigma(G, E) := T_\Sigma(G)/\mathrm{Cong}_{T_\Sigma(G)}(E)$$

der oben erwähnte Quotient. Wir kürzen $\mathrm{Cong}_{T_\Sigma(G)}$ *auch als* $\langle E \rangle$ *ab und schreiben dann etwas kompakter*

$$T_\Sigma(G, E) := T_\Sigma(G)/\langle E \rangle \qquad \Diamond$$

Aufgrund von Satz 2.14 und der Bemerkung am Ende von II.2 erhalten wir für jede Σ-Algebra A eine bijektive Entsprechung zwischen Homomorphismen von $T_\Sigma(G)/\langle E \rangle$ nach A und Homomorphismen von $T_\Sigma(G)$ nach A, die „die Bedingungen von E erfüllen".

Theorem 4.6 *Sei G eine Familie von Generatoren wie oben und*

$$E = (E_s \subseteq T_\Sigma(G)_s^2 \mid s \in S)$$

eine Familie von Gleichheitsbedingungen. Sei weiter A eine beliebige Σ–Algebra.

i) *Für jeden Homomorphismus $h : T_\Sigma(G, E) \to A$ ist $k := h \circ \mathrm{quot}_{\langle E \rangle}$ ein Homomorphismus, der E erfüllt, d.h. es gilt $k(t_1) = k(t_2) \in A_s$ für alle $(t_1, t_2) \in E_s$.*

ii) *Für jeden Homomorphismus $k : T_\Sigma(G) \to A$, der die Bedingungen E erfüllt, gibt es einen eindeutig bestimmten Homomorphismus $h : T_\Sigma(G, E) \to A$, so dass $h \circ \mathrm{quot}_{\langle E \rangle} = k$.*

Beweis:

i) Wenn $t_1 \; E \; t_2$, dann auch $t_1 \; \langle E \rangle \; t_2$, da $\langle E \rangle$ die Relation E enthält, und somit

$$\mathrm{quot}_{\langle E \rangle}(t_1) = [t_1]_{\langle E \rangle} = [t_2]_{\langle E \rangle} = \mathrm{quot}_{\langle E \rangle}(t_2)$$

woraus aber unmittelbar folgt, dass

$$k(t_1) = h(\mathrm{quot}_{\langle E \rangle}(t_1)) = h(\mathrm{quot}_{\langle E \rangle}(t_2)) = k(t_2)$$

wie gefordert.

ii) Wenn $k : T_\Sigma(G) \to A$ die Bedingungen E erfüllt, so gilt $E \subseteq \ker_k$. Da $\ker_k$ eine Kongruenzrelation ist, gilt also auch $\langle E \rangle \subseteq \ker_k$. Damit respektiert k die Kongruenz $\langle E \rangle$ und es gibt wegen der Bemerkung zu Satz 2.15 einen Homomorphismus

$$h' : T_\Sigma(G)/\langle E \rangle \to T_\Sigma(G)/\ker_k$$

mit $[t]_{\langle E \rangle} \mapsto [t]_{\ker_k}$, so dass $h' \circ \mathrm{quot}_{\langle E \rangle} = \mathrm{quot}_{\ker_k}$. Dies bedeutet, dass das obere Dreieck in folgendem Diagramm kommutiert:

$$\begin{tikzcd}
& T_\Sigma(G,E) \arrow[d, "h'"] \\
T_\Sigma(G) \arrow[ur, two heads, "\mathrm{quot}_{\langle E\rangle}"] \arrow[r, two heads, "\mathrm{quot}_{\ker_k}"'] \arrow[d, "k"'] & T_\Sigma(G)/\ker_k \arrow[d, "\cong"', "i_k"] \\
A & k[T_\Sigma(G)] \arrow[l, hook', "m_k"]
\end{tikzcd}$$

Wegen Satz 2.14 gilt weiter $k = m_k \circ i_k \circ e_k$ mit $e_k = \mathrm{quot}_{\ker_k}$ und damit kommutiert auch der untere Teil des Diagramms.

Daraus folgt, dass für $h := m_k \circ i_k \circ h'$ gilt, dass $k = h \circ \mathrm{quot}_{\langle E \rangle}$. Dieses h ist eindeutig, weil $\mathrm{quot}_{\langle E \rangle}$ surjektiv ist. □

Bemerkung: Die Quotientenabbildung $\mathrm{quot}_{\langle E \rangle}$ ist sozusagen der „kleinste" Homomorphismus, der E erfüllt, da jeder andere Homomorphismus $h : T_\Sigma(G) \to A$, der E erfüllt, sich längs $\mathrm{quot}_{\langle E \rangle}$ eindeutig fortsetzen lässt.
Interessante Beispiele für die Definition einer Algebra A als $T_\Sigma(G, E)$ werden wir im nächsten Abschnitt über Gleichungslogik kennenlernen.

Zuvor werden wir uns aber die Funktionen $\mathrm{Int}^A(\rho) = [\![\_]\!]^A_\rho : T_\Sigma(\mathcal{V}) \to A$ für $\rho \in \mathrm{Val}^A_\mathcal{V}$ noch etwas näher auf ihren Nutzen hin ansehen.

Wenn t ein Term in den freien Variablen $x_1\colon s_1, \ldots, x_n\colon s_n$ ist und $\mathcal{V}$ die entsprechende Familie von Variablen, so können wir die durch t induzierte *Polynomfunktion*

$$f_t : A_{s_1} \times \cdots \times A_{s_n} \to A_s \qquad (t \in T_\Sigma(\mathcal{V}))$$

betrachten, die folgendermaßen definiert ist:

$$f_t(a_1, \ldots, a_n) = [\![t]\!]_{\rho_{\vec{a}}} \qquad \text{mit } \rho_{\vec{a}}(x_i) = a_i \text{ für } i = 1, \ldots, n.$$

Man belegt also jedes der Variablensymbole mit einem Wert, wertet dann t bezüglich dieser Belegung aus dann aus und erhält auf diese Weise den Funktionwert, der durch t induzierten Polynomfunktion.

Es erweist sich als zweckmässig, für $\mathcal{V}$ nicht bloß die in t echt frei vorkommenden Variablen zu nehmen sondern $\mathcal{V}$ hinreichend groß zu wählen, nämlich als $\mathcal{V}_s = \{x_0^s, x_1^s, \ldots, x_n^s, \ldots\} = \{x_n^s \mid n \in \mathbb{N}\}$ für $s \in S$, da ja abzählbar viele Variablen für beliebig große Terme ausreichen.

Auf diese Art können wir eine beliebige Belegung auf eindeutige Weise zu einem Σ–Homomorphismus fortsetzen.

Definition 4.7 *Es sei* $\Sigma{-}\mathrm{Hom}(A, B)$ *die Menge der* Σ*–Homomorphismen von* A *nach* B*. Für jede Signatur* $\Sigma = (S, F, \mathrm{ar})$*, jede* S*-indizierte Mengenfamilie* $\mathcal{V}$ *und jede* Σ*-Algebra sei die Funktion*

$$[\![\_]\!]^A_{\Sigma,\mathcal{V}} : \mathrm{Val}^A_{\mathcal{V}} \to \Sigma{-}\mathrm{Hom}(T_\Sigma(\mathcal{V}), A)$$

durch $[\![t]\!]^A_{\Sigma,\mathcal{V}}(\rho) = [\![t]\!]^A_\rho$ *definiert. Die Funktion* $[\![\_]\!]^A_{\Sigma,\mathcal{V}}$ *wird auch* Interpretationsfunktion *für* Σ*–Terme über* $\mathcal{V}$ *in* A *genannt und mit* Int^A *bezeichnet.* $\Diamond$

Wenn wir in obiger Definition für A speziell die Algebra $T_\Sigma(\mathcal{V})$ einsetzen, so erhalten wir *Termsubstitution* als Spezialfall der Interpretation, nämlich in der Termalgebra $T_\Sigma(\mathcal{V})$. Dabei sind Substitutionen einfach beliebige Belegungen $\rho \in T_\Sigma(\mathcal{V})^{\mathcal{V}}$.

Man beachte, dass solche ρ geringfügig allgemeiner sind als die üblichen syntaktischen Substitutionen, da ein solches beliebiges ρ unter Umständen *unendlich viele* Variablen in $\mathcal{V}$ durch beliebige Terme ersetzen kann. Die üblichen altvertrauten Substitutionen $[t_1, \ldots, t_n/x_1 \ldots, x_n]$ (lies: die Variable x_i wird durch den Term t_i ersetzt) erhält man wieder, wenn man ρ wie folgt definiert:

$$\rho(x) \equiv \begin{cases} t_i & \text{falls } x \equiv x_i \\ x & \text{sonst} \end{cases}$$

Wir gießen diese Betrachtung in eine Definition.

Definition 4.8 *Sei* $\Sigma = (S, F, ar)$ *eine Signatur und* $\mathcal{V}$ *eine* S*–indizierte Mengenfamilie. Die Menge der* (verallgemeinerten) Substitutionen $\mathrm{Subst}_{\Sigma,\mathcal{V}}$ *sei definiert als* $T_\Sigma(\mathcal{V})^{\mathcal{V}}$*. Für einen Term* t *und ein* $\sigma \in \mathrm{Subst}_{\Sigma,\mathcal{V}}$ *sei die* Anwendung der Substitution σ auf den Term t *als*

$$t[\sigma] \equiv [\![t]\!]^{T_\Sigma(\mathcal{V})}_\sigma$$

definiert. $\Diamond$

Aufgrund der Definition der Interpretationsfunktion $[\![\_]\!]^{T_\Sigma(\mathcal{V})}_{\Sigma,\mathcal{V}}$ kann man sich leicht klar machen, dass $t[\sigma]$ aus t dadurch hervorgeht, dass man jedes Vorkommen einer Variablen $x \in \mathcal{V}$ in t durch den Term $\sigma(x)$ *ersetzt.*

Die Aussage von Satz 4.3, nämlich dass sich jedes $\rho : \mathcal{V} \to A$ auf eindeutige Weise zu einem Homomorphismus $\mathrm{Int}^A(\rho) : T_\Sigma(\mathcal{V}) \to A$ fortsetzen lässt, hat folgende wichtige Konsequenz.

Theorem 4.9 *Sei $\Sigma = (S, F, ar)$ eine Signatur und $\mathcal{V}$ eine Familie von Variablenmengen.*

i) *Für beliebige Σ–Algebren A, B und Σ–Homomorphismen $h : A \to B$ gilt, dass*
$$h \circ \mathrm{Int}^A(\rho) = \mathrm{Int}^B(h \circ \rho)$$
für alle Belegungen $\rho : \mathcal{V} \to A$.

ii) *Für jede Σ–Algebra A, jede Belegung $\rho : \mathcal{V} \to A$ und jede Substitution $\sigma : \mathcal{V} \to T_\Sigma(\mathcal{V})$ gilt*
$$\mathrm{Int}^A(\rho) \circ \mathrm{Int}^{T_\Sigma(\mathcal{V})}(\sigma) = \mathrm{Int}^A(\mathrm{Int}^A(\rho) \circ \sigma)$$
d.h. $[\![t[\sigma]]\!]^A_\rho = [\![t]\!]^A_{(\mathrm{Int}^A(\rho)\circ\sigma)}$ für jeden Term $t \in T_\Sigma(\mathcal{V})$.
Mit anderen Worten: Es spielt keine Rolle, ob man zuerst substituiert und dann interpretiert, oder ob man bezüglich einer gemäß der Substitution abgeänderten Variablenbelegung interpretiert.

Beweis: Nach 4.3 können wir die Gleichheit der Homomorphismen überprüfen, indem wir zeigen, dass sie sich auf $x \in \mathcal{V}$ gleich verhalten.
Für (i) ergibt sich das aus
$$(h \circ \mathrm{Int}^A(\rho))(x) = h(\mathrm{Int}^A(\rho)(x)) = h(\rho(x)) = (h \circ \rho)(x) = \mathrm{Int}^B(h \circ \rho)(x)$$
Ferner erhält man (ii) aus (i), in dem man für ρ die Substitution σ einsetzt und für h den Homomorphismus $\mathrm{Int}^A(\rho)$.
Folgende Diagramme veranschaulichen die Situation

$$\begin{tikzcd}
\mathcal{V} \arrow[rr, hook, "\iota"] \arrow[dr, "\rho"] \arrow[ddr, "h\circ\rho"'] & & T_\Sigma(\mathcal{V}) \arrow[dl, "\mathrm{Int}^A(\rho)"'] \arrow[ddl, "\mathrm{Int}^B(h\circ\rho)"] \\
& A \arrow[d, "h"] & \\
& B &
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathcal{V} \arrow[rr, hook, "\iota"] \arrow[dr, "\sigma"] \arrow[ddr, "\mathrm{Int}(\rho)\circ\sigma"'] & & T_\Sigma(\mathcal{V}) \arrow[dl, "\mathrm{Int}(\sigma)"'] \arrow[ddl, "\mathrm{Int}(\mathrm{Int}(\rho)\circ\sigma)"] \\
& T_\Sigma(\mathcal{V}) \arrow[d, "\mathrm{Int}(\rho)"] & \\
& A &
\end{tikzcd}$$

Aus $\mathrm{Int}^A(\rho) \circ \mathrm{Int}^{T_\Sigma(\mathcal{V})}(\sigma) = \mathrm{Int}^A(\mathrm{Int}^A(\rho) \circ \sigma)$ folgt dann, dass
$$[\![t[\sigma]]\!]_\rho = \mathrm{Int}(\rho)(\mathrm{Int}(\sigma)(t)) = \mathrm{Int}(\mathrm{Int}(\rho) \circ \sigma)(t)) = [\![t]\!]_{(\mathrm{Int}(\rho)\circ\sigma)}$$

für beliebige Terme t. $\square$

Punkt (ii) dieses Satzes ist üblicherweise als so genanntes *Substitutionslemma* bekannt. Es besagt, dass für einen Term t und eine Substitution σ die Interpretation von $t[\sigma]$ in A bezüglich einer Belegung $\rho : \mathcal{V} \to A$ gleich ist der Interpretation des Terms t unter der Belegung $\mathrm{Int}^A(\rho) \circ \sigma$, die jede Variable x mit $[\![\sigma(x)]\!]^A\rho$ belegt. Wenn wir also nur die Variablen $x_1, \ldots x_n$ hätten, dann würde man schreiben

$$[\![t[t_1, \ldots, t_n/x_1, \ldots, x_n]]\!]^A\rho = [\![t]\!]^A[[\![t_1]\!]^A\rho, \ldots, [\![t_n]\!]^A\rho/x_1, \ldots x_n]$$

d.h. statt einen substituierten Term unter ρ auszuwerten, kann man den Term unter der Belegung auswerten, die jeder Variablen x die Interpretation unter ρ desjenigen Terms zuordnet, der für x bei Durchführung der Substitution eingesetzt wurde.

II.5 Gleichungslogik und initiale Modelle

Wir haben bereits informell Sprechweisen verwendet wie „eine Gleichung $t_1 = t_2$ gilt in einer Σ–Algebra A“ oder „die Klasse aller Σ–Algebren, die eine vorgegebene Menge von Gleichungen erfüllt“. Wir werden diese Begriffe vorerst einmal mathematisch präzise fassen. Später geben wir einen *Herleitungskalkül* an, der für eine vorgegebene Menge E von Gleichungen genau diejenigen Gleichungen herzuleiten gestattet, die in *jeder* Algebra A gelten, in der alle Gleichungen aus E gelten.

Sei $\Sigma = (S, F, \mathrm{ar})$ eine Signatur und $\mathcal{V}$ eine S–indizierte Familie von abzählbar unendlichen Mengen. Ohne Beschränkung der Allgemeinheit setzen wir

$$\mathcal{V}_s = \{x_0^s, x_1^s, \ldots, x_n^s, \ldots\} = \{x_n^s \mid n \in \mathbb{N}\} \quad .$$

Wir halten uns also für jede Sorte $s \in S$ eine abzählbare Menge von Variablen der Sorte s zur Verfügung.

Definition 5.1 *Eine* Gleichung (über Σ in den Variablen $\mathcal{V}$) *ist ein Paar von Termen in $T_\Sigma(\mathcal{V})_s$ für ein $s \in S$. Für ein solches Paar $(t_1, t_2) \in T_\Sigma(\mathcal{V})_s^2$ schreiben wir $t_1 =_s t_2$ oder einfach $t_1 = t_2$.*

Sei $t_1 =_s t_2$ eine Gleichung und A eine Σ–Algebra. Die Gleichung $t_1 =_s t_2$ gilt *in A unter der Belegung $\rho \in \mathrm{Val}_{\mathcal{V}}^A$ genau dann, wenn*

$$[\![t_1]\!]_\rho^A = [\![t_2]\!]_\rho^A \; .$$

Wenn $t_1 = t_2$ in A unter der Belegung ρ gilt, so schreiben wir abkürzend

$$A \models_\rho t_1 = t_2$$

für diesen Sachverhalt. Eine Gleichung $t_1 = t_2$ gilt in der Algebra *A genau dann, wenn $A \models_\rho t_1 = t_2$ für alle $\rho \in \mathrm{Val}_{\mathcal{V}}^A$, wofür wir abkürzend*

$$A \models t_1 = t_2$$

schreiben. Für den Sachverhalt $A \models t_1 = t_2$ sagen wir auch, dass A die Gleichung $t_1 = t_2$ erfüllt. ◊

Mithilfe dieser Begriffe können wir die Klasse von Algebren definieren, die eine vorgegebene Menge von Gleichungen erfüllen.

Definition 5.2 *Sei $E = (E_s \subseteq T_\Sigma(\mathcal{V})_s^2 \mid s \in S)$ eine Familie von Gleichungen. Eine Σ–Algebra A* erfüllt die Gleichungen in *E genau dann, wenn $A \models t_1 = t_2$ für jede Gleichung $t_1 = t_2$ in E.*

Die Klasse der Σ–Algebren, die alle Gleichungen in E erfüllen, bezeichnen wir mit $\mathrm{Alg}_\Sigma(E)$ *oder einfach* $\mathrm{Alg}(E)$. *Wir schreiben auch* $\mathrm{Mod}(E)$ *für* $\mathrm{Alg}(E)$, *da wir eine Algebra, die E erfüllt, auch als ein Modell für E bezeichen.*

Wir schreiben

$$E \models t_1 = t_2$$

für die Aussage, dass $A \models t_1 = t_2$ *für alle* $A \in \mathrm{Alg}(E)$. *Ferner sei die* Theorie *der Gleichungsfamilie* E *definiert als*

$$\mathrm{Th}(E) := \{t_1 = t_2 \mid E \models t_1 = t_2\}$$

d.h. als die Menge der Gleichungen, die in allen Algebren gelten, die alle Gleichungen aus E *erfüllen.* $\Diamond$

Warnung! Wenn A eine Σ–Algebra ist, in der $A_s = \emptyset$ für eine Sorte $s \in S$, dann ist auch $\mathrm{Val}_{\mathcal{V}}^{A} = \emptyset$, weil ja $\rho(x_0^s) \in A_s$ gälte, wenn es ein $\rho \in \mathrm{Val}_{\mathcal{V}}^{A}$ gäbe. Dies hat aber den (unerwünschten) Effekt, dass in einer solchen Algebra A per Definition alle(!) Gleichungen gelten.

Um dieses Problem zu umgehen, könnte man die Definition von Σ–Algebren dahingehend abändern, dass man verlangt, dass alle Trägermengen A_s *nicht leer* sein sollen. Wir ziehen es jedoch vor, die betrachteten Signaturen auf solche Σ einzuschränken, für welche $T_\Sigma(\emptyset)_s \neq \emptyset$ für alle Sorten s, d.h., wo für jede Sorte s ein variablenfreier Term dieser Sorte existiert. Solche Signaturen werden als „vernünftig“ (englisch *sensible*) bezeichnet. Die Signatur von Monoiden und Gruppen ist also vernünftig, die von Automaten (ohne einen ausgezeichneten Startzustand) nicht.

Ungeachtet all dieser marginaler Probleme verharren wir bei unserer obigen Definition von Gültigkeit von Gleichungen und **schränken uns im weiteren auf „vernünftige“ Signaturen ein**.

Als nächstes betrachten wir etwas eingehender die Struktur von Gleichungsmengen der Gestalt $\mathrm{Th}(E)$.

Theorem 5.3 *Sei* E *eine Gleichungsmenge. Für* $\mathrm{Th}(E)$ *gelten dann folgende Abschlußeigenschaften:*

a) $E \models t = t$ *für jeden Term* t.

b) *Wenn* $E \models t_1 = t_2$, *dann auch* $E \models t_2 = t_1$.

c) *Wenn* $E \models t_1 = t_2$ *und* $E \models t_2 = t_3$, *dann auch* $E \models t_1 = t_3$.

d) *Wenn* $f : s_1 \times \ldots \times s_n \to s$ *ein Operationssymbol ist und* $t_i, t_i' \in T_\Sigma(\mathcal{V})_{s_i}$ *für* $i = 1, \ldots, n$, *dann folgt aus* $E \models t_i = t_i'$ *für* $i = 1, \ldots, n$, *dass auch* $E \models f(t_1, \ldots, t_n) = f(t_1', \ldots, t_n')$.

e) *Wenn* $\sigma \in \mathrm{Val}_{\mathcal{V}}^{T_\Sigma(\mathcal{V})}$ *eine Substitution ist und* $E \models t_1 = t_2$, *dann auch* $E \models t_1[\sigma] = t_2[\sigma]$.

f) *Wenn* $t_1 = t_2$ *in* E *ist, dann ist* $t_1 = t_2$ *auch in* $\mathrm{Th}(E)$.

Beweis: Die Eigenschaften a), b), c) und f) sind trivial. Eigenschaft d) folgt unmittelbar aus der Definition der Interpretationsfunktion (Übung!) und e) beweist man folgendermaßen mithilfe des Substitutionslemmas:

Angenommen, es gilt $E \models t_1 = t_2$ und $\sigma \in \mathrm{Val}_{\mathcal{V}}^{T_\Sigma(\mathcal{V})}$. Wir müssen zeigen, dass $E \models t_1[\sigma] = t_2[\sigma]$. Sei also $A \in \mathrm{Alg}(E)$ und $\rho \in \mathrm{Val}_{\mathcal{V}}^{A}$. Dann gilt

$$[\![t_1[\sigma]]\!]^A \rho = [\![t_1]\!]^A(\mathrm{Int}(\rho) \circ \sigma) = [\![t_2]\!]^A(\mathrm{Int}(\rho) \circ \sigma) = [\![t_2[\sigma]]\!]^A \rho$$

wobei sich die erste und die letzte Gleichheit aus dem Substitutionslemma ergeben und die mittlere Gleichheit unmittelbar aus der Annahme $E \models t_1 = t_2$ folgt. $\square$

Aufgrund des eben bewiesenen Theorems wissen wir also, dass $\mathrm{Th}(E)$ eine Kongruenzrelation auf $T_\Sigma(\mathcal{V})$ ist, die wegen e) überdies unter beliebigen Substitutionen abgeschlossen ist.

Wir werden nun im weiteren zeigen, dass $\mathrm{Th}(E)$ in der Tat auch die *kleinste* Kongruenzrelation auf $T_\Sigma(\mathcal{V})$ ist, die unter beliebigen Substitutionen abgeschlossen ist und E enthält. Daraus eröffnet sich die Möglichkeit, $\mathrm{Th}(E)$ mithilfe der Regelschemata a) bis f) aus Theorem 5.3 induktiv zu definieren. Diese nennt man auch die **Schlußregeln** der **Gleichungslogik** (über der Axiomenmenge E).

Zu diesem Ziel benötigen wir aber eine Definition und einen zentralen Hilfssatz.

Definition 5.4 *Eine Kongruenz* R *auf* $T_\Sigma(\mathcal{V})$ *heißt* substitutionsabgeschlossen *genau dann, wenn für alle* $t_1 \, R \, t_2$ *und* $\sigma : \mathcal{V} \to T_\Sigma(\mathcal{V})$ *gilt, dass* $t_1[\sigma] \, R \, t_2[\sigma]$. $\Diamond$

Lemma 5.5 *Sei* R *eine substitutionsabgeschlossene Kongruenz auf* $T_\Sigma(\mathcal{V})$*. Dann gilt* $T_\Sigma(\mathcal{V})/R \models t_1 = t_2$ *genau dann, wenn* $t_1 R \, t_2$.

Beweis: Zuerst beobachten wir, dass es für jede Belegung $\rho : \mathcal{V} \to T_\Sigma(\mathcal{V})/R$ eine Substitution $\sigma : \mathcal{V} \to T_\Sigma(\mathcal{V})$ gibt mit $\rho = \mathrm{quot}_R \circ \sigma$ wie in dem Diagramm

$$\begin{tikzcd}
\mathcal{V} \arrow[r, "\sigma"] \arrow[dr, "\rho"'] & T_\Sigma(\mathcal{V}) \arrow[d, "\mathrm{quot}_R"] \\
& T_\Sigma(\mathcal{V})/R
\end{tikzcd}$$

veranschaulicht. Also gilt

$$T_\Sigma(\mathcal{V})/R \models t_1 = t_2 \iff \forall \sigma \in \mathrm{Val}_{\mathcal{V}}^{T_\Sigma(\mathcal{V})}. \; [\![t_1]\!]^{T_\Sigma(\mathcal{V})/R}_{(\mathrm{quot}_R \circ \sigma)} = [\![t_2]\!]^{T_\Sigma(\mathcal{V})/R}_{(\mathrm{quot}_R \circ \sigma)} \quad .$$

Außerdem gilt für jeden Term t, dass

$$[\![t]\!]^{T_\Sigma(\mathcal{V})/R}_{(\mathrm{quot}_R \circ \sigma)} = \mathrm{quot}_R(t[\sigma])$$

denn, da $\mathrm{quot}_R : T_\Sigma(\mathcal{V}) \to T_\Sigma(\mathcal{V})/R$ ein Homomorphismus ist, rechnet man

$$\begin{array}{ll} & [\![t]\!]^{T_\Sigma(\mathcal{V})/R}_{(\mathrm{quot}_R \circ \sigma)} = \\ = & \mathrm{Int}(\mathrm{quot}_R \circ \sigma)(t) = (\mathrm{quot}_R \circ \mathrm{Int}(\sigma))(t) = \mathrm{quot}_R(\mathrm{Int}(\sigma)(t)) = \\ = & \mathrm{quot}_R(t[\sigma]) \; . \end{array}$$

Aus diesen beiden Beobachtungen ergibt sich, dass

$$T_\Sigma(\mathcal{V})/R \models t_1 = t_2 \iff \forall \sigma \in \mathrm{Val}^{T_\Sigma(\mathcal{V})}_{\mathcal{V}}.\, \mathrm{quot}_R(t_1[\sigma]) = \mathrm{quot}_R(t_2[\sigma])$$

und somit

$$T_\Sigma(\mathcal{V})/R \models t_1 = t_2 \iff \forall \sigma \in \mathrm{Val}^{T_\Sigma(\mathcal{V})}_{\mathcal{V}}.\, t_1[\sigma] \; R \; t_2[\sigma] \iff t_1 \; R \; t_2$$

wobei die letzte Äquivalenz aufgrund der Substitutionsabgeschlossenheit von R gilt. □

Theorem 5.6 *Sei E eine Menge von Gleichungen. Dann ist* $\mathrm{Th}(E)$ *die kleinste substitutionsabgeschlossene Kongruenz auf* $T_\Sigma(\mathcal{V})$*, die E enthält.*

Beweis: Wegen Theorem 5.3 ist $\mathrm{Th}(E)$ eine substitutionsabgeschlossene Kongruenz auf $T_\Sigma(\mathcal{V})$, die E enthält.

Sei R eine substitutionsabgeschlossene Kongruenz auf $T_\Sigma(\mathcal{V})$, die E enthält. Dann gelten in $T_\Sigma(\mathcal{V})/R$ aufgrund von Lemma 5.5 alle Gleichungen aus E, d.h. $T_\Sigma(\mathcal{V})/R \in \mathrm{Alg}(E)$. Deshalb gilt für $t_1 \; \mathrm{Th}(E) \; t_2$, dass $T_\Sigma(\mathcal{V})/R \models t_1 = t_2$ und somit $t_1 \; R \; t_2$ wegen Lemma 5.5. Also ist $\mathrm{Th}(E) \subseteq R$. □

Die den Abschlußeigenschaften a) bis f) aus Lemma 5.3 entsprechenden Schlußregeln nennen wir (Refl), (Symm), (Trans), (Congr), (Subst) und (Ax_E) für Reflexivität, Symmetrie, Transitivität, Kongruenz Substitution und Axiome (aus E).

Die praktische Relevanz von Theorem 5.6 besteht darin, dass die Gültigkeit einer Gleichung in allen Algebren von $\mathrm{Alg}(E)$, d.h. allen Modellen von E, durch *endlich*malige Anwendung der Schlußregeln (Refl), (Symm), (Trans), (Congr), (Subst) und (Ax_E) nachgewiesen werden kann. Offensichtlich ist $\mathrm{Th}(E)$ durch ein Computerprogramm algorithmisch aufzählbar, falls die Axiomenmenge E durch ein Computerprogramm algorithmisch aufzählbar ist.

Diese *Gleichungslogik* ist ein prototypisches Beispiel für Herleitungskalküle, wie sie in der mathematische Logik studiert werden. Dabei ist die Zielsetzung, für eine rein semantisch definierte Menge „wahrer Sätze“ nachzuweisen, dass sie durch

ein möglichst einfaches und intuitives Regelsystem induktiv definiert wird. Die Regeln eines solchen Systems nennt man *Schlußregeln*, das System selbst *Herleitungskalkül.*
Amüsanter- und nützlicherweise ist die Gültigkeit einer Gleichung in allen Modellen von E äquivalent zur Gültigkeit in einem einzigen Modell, dem so genannten „klassifizierenden Modell“, das man erhält, indem man $T_\Sigma(\mathcal{V})$ nach $\mathrm{Th}(E)$ faktorisiert.

Theorem 5.7 *Für eine Menge E von Gleichungen gilt $E \models t_1 = t_2$ genau dann, wenn $T_\Sigma(\mathcal{V})/\mathrm{Th}(E) \models t_1 = t_2$.*
Deshalb nennen wir $\mathrm{Cl}(E) := T_\Sigma(\mathcal{V})/\mathrm{Th}(E)$ *das* klassifizierende Modell von E.

Beweis: Es gilt

$$E \models t_1 = t_2 \Longleftrightarrow t_1 \,\mathrm{Th}(E)\, t_2 \Longleftrightarrow T_\Sigma(\mathcal{V})/\mathrm{Th}(E) \models t_1 = t_2$$

wobei die letzte Äquivalenz aufgrund von Lemma 5.5 gilt. □

Das klassifizierende Modell von E ist deshalb nützlich, da es uns eine *explizite* Konstruktion der initialen Algebra von $\mathrm{Alg}(E)$ ermöglichen wird. Vorerst unterziehen wir aber das klassifizierende Modell selbst einer etwas eingehenderen Betrachtung.
Wenn etwa E die Gleichungen der Theorie der kommutativen Ringe mit 1 ist, dann ist $\mathrm{Cl}(E)$ der Ring der Polynome über abzählbar unendlich vielen Variablen mit ganzzahligen Koeffizienten (Übung!). Offenbar gilt also eine Gleichung $t = s$ in allen kommutativen Ringen mit 1 genau dann, wenn die durch „Ausrechnen“ von t und s erhaltenen Polynome gleich sind.
Im allgemeinen ist jedoch die Situation nicht so schön. Betrachten wir etwa das klassifizierende Modell Cl(NAT–ADD) der Spezifikation

NAT–ADD	$\equiv$
sorts	nat
funcs	zero : $\to$ nat
	succ : nat $\to$ nat
	add : nat $\times$ nat $\to$ nat
axioms	$\mathrm{add}(x, \mathrm{zero}) = x$
	$\mathrm{add}(x, \mathrm{succ}(y)) = \mathrm{succ}(\mathrm{add}(x, y))$

Offenbar ist die Gleichung $x = \mathrm{succ}^n(\mathrm{zero})$ in Cl(NAT–ADD) für kein $n \in \mathbb{N}$ erfüllt, da ja sonst diese Gleichung im Standardmodell $\mathbb{N}$ der natürlichen Zahlen mit der üblichen Interpretation von zero, succ und add gelten würde, was ja nicht der Fall ist, da ja nicht alle natürlichen Zahlen gleich einer fix vorgegebenen Zahl sind. Aus demselben Grund gilt $x = t$ in Cl(NAT–ADD) für keinen variablenfreien Term t. Außerdem gelten die Gleichungen

$$\mathrm{add}(x, \mathrm{add}(y, z)) = \mathrm{add}(\mathrm{add}(x, y), z) \qquad \mathrm{add}(x, y) = \mathrm{add}(y, x)$$

in Cl(NAT–ADD) nicht, da sie in dem Modell

$$A_{\mathrm{nat}} = \mathbb{Z} \quad \mathrm{zero}^A = 0 \quad \mathrm{succ}^A(x) = x \quad \mathrm{add}^A(x, y) = x - y$$

von NAT–ADD verletzt sind. Also ist Cl(NAT–ADD) grundverschieden von dem intendierten Standardmodell $\mathbb{N}$. Ganz allgemein gilt (Übung!), dass $\mathrm{Cl}_\Sigma(E)$ genau dann termerzeugt ist, wenn *alle* Gleichungen aus E beweisbar sind, d.h. wenn E bloß triviale Modelle hat (wo alle Trägermengen einelementig sind).

Wir zeigen als nächstes, dass $\mathrm{Cl}(E)$ die „freie“ Algebra in $\mathrm{Alg}(E)$ über der Erzeugermenge $\mathcal{V}$ ist.

Theorem 5.8 *Sei E eine Menge von Gleichungen über einer Signatur Σ und $A \in \mathrm{Alg}(E)$. Dann existiert für jede Belegung $\rho : \mathcal{V} \to A$ ein eindeutig bestimmter (Σ–)Homomorphismus $h : \mathrm{Cl}(E) \to A$, sodass*

$$h([x]_{\mathrm{Th}(E)}) = \rho(x)$$

für alle $x \in \mathcal{V}$.

Beweis: Die Bedingung für h ist äquivalent zu $h \circ \mathrm{quot}_{\mathrm{Th}(E)} \circ \iota = \rho$

$$\begin{tikzcd}
 & T_\Sigma(\mathcal{V}) \arrow[d, "\mathrm{quot}_{\mathrm{Th}(E)}"] \\
\mathcal{V} \arrow[ur, hook, "\iota"] \arrow[dr, "\rho"'] & \mathrm{Cl}(E) \arrow[d, "h"] \\
 & A
\end{tikzcd}$$

wobei $\iota : \mathcal{V} \hookrightarrow T_\Sigma(\mathcal{V})$ für die Inklusion der Variablen in $T_\Sigma(\mathcal{V})$ steht. Offensichtlich muss dann gelten, dass $h \circ \mathrm{quot}_{\mathrm{Th}(E)} = \mathrm{Int}^A(\rho)$, womit h eindeutig festgelegt ist, da $\mathrm{quot}_{\mathrm{Th}(E)}$ ein surjektiver Homomorphismus ist. Die Existenz von h ergibt sich durch die Setzung

$$h([t]_{\mathrm{Th}(E)}) = \mathrm{Int}^A(\rho)(t)\ .$$

Die Wohldefiniertheit von h ergibt sich aus dem Sachverhalt, dass $\mathrm{Int}^A(\rho)(t_1) = [\![t_1]\!]^A\rho = [\![t_2]\!]^A\rho = \mathrm{Int}^A(\rho)(t_2)$ für $t_1\, \mathrm{Th}(E)\, t_2$, da A als Modell von E vorausgesetzt war. □

Dieses Theorem ermöglicht uns folgende explizite Konstruktion *initialer Modelle.*

Theorem 5.9 *Sei E eine Menge von Gleichungen über einer „vernünftigen“ Signatur Σ, d.h. für jede Sorte gibt es einen variablenfreien Term dieser Sorte. Dann ist die Algebra*

$$\mathrm{Init}_\Sigma(E) := \mathrm{Cl}_\Sigma(E)^\circ\ .$$

d.h. der termerzeugte Anteil des klassifizierenden Modells von E, initial *in* $\mathrm{Alg}(E)$, *d.h. für jede Algebra $A \in \mathrm{Alg}(E)$ gibt es genau einen Homomorphismus $h : \mathrm{Init}_\Sigma(E) \to A$.*

Beweis: Sei $A \in \mathrm{Alg}(E)$ und $\rho : \mathcal{V} \to A$ eine beliebige Variablenbelegung (es gibt mindestens eine, da die Trägermengen von A nichtleer sind). Aufgrund von Theorem 5.8 gibt es dann einen (eindeutigen) Homomorphismus $\widetilde{h} : \mathrm{Cl}(E) \to A$ (mit $\rho = \widetilde{h} \circ \mathrm{quot}_{\mathrm{Th}(E)} \circ \iota$). Der gesuchte Homomorphismus $h : \mathrm{Init}_\Sigma(E) \to A$ ergibt sich als Einschränkung von $\widetilde{h}$ auf die termerzeugte Unteralgebra $\mathrm{Init}_\Sigma(E)$ von $\mathrm{Cl}(E)$. Dieser Homomorphismus h ist eindeutig, da $\mathrm{Init}_\Sigma(E)$ nach Konstruktion termerzeugt ist. $\square$

Dieser Satz stellt uns eine syntaktische (und deshalb explizite) Konstruktion eines initialen Modells von E bereit: man zeigt leicht (Übung!), dass für *variablenfreie Terme* t (auch *Grundterme* genannt) gilt, dass

$$[\![t]\!]_\rho^{\mathrm{Cl}(E)} = [t]_{\mathrm{Th}(E)}$$

für beliebige Belegungen $\rho : \mathcal{V} \to \mathrm{Cl}(E)$, von denen allerdings die Interpretation eines variablenfreien Terms gar nicht abhängt. Also ist $\mathrm{Init}_\Sigma(E)$ die Unteralgebra von $\mathrm{Cl}(E)$ auf denjenigen Äquivalenzklassen (modulo $\mathrm{Th}(E)$), die einen Grundterm als Element enthalten.
Diese Aussage lässt sich noch folgendermassen verbessern.

Theorem 5.10 *Sei E eine Menge von Gleichungen über einer „vernünftigen“ Signatur Σ. Dann ist* $\mathrm{Init}_\Sigma(E)$ *isomorph zu*

$$T_\Sigma(\emptyset)/(\mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2)$$

d.h. zu dem Quotienten der Algebra $T_\Sigma(\emptyset)$ der Grundterme bezüglich der Kongruenz, die man aus $\mathrm{Th}(E)$ *durch Einschränkung auf Grundterme erhält.*
Man erhält also ein initiales Modell von E aus der Algebra der Grundterme, indem man alle beweisbar gleichen Grundterme identifiziert.

Beweis: Jede Äquivalenzklasse von $\mathrm{Init}_\Sigma(E) = (T_\Sigma(\mathcal{V})/\mathrm{Th}(E))^\circ$ ist von der Gestalt $[t]_{\mathrm{Th}(E)}$ für einen Grundterm $t \in T_\Sigma(\emptyset)$. Für jede Sorte s definieren wir

$$i_s : \mathrm{Init}_\Sigma(E)_s \to (T_\Sigma(\emptyset)/(\mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2))_s : [t]_{\mathrm{Th}(E)} \mapsto [t]_{\mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2}$$

d.h. $i_s([t]_{\mathrm{Th}(E)}) = [t]_{\mathrm{Th}(E)} \cap T_\Sigma(\mathcal{V})$. Offenbar ist i_s bijektiv für jedes s. Man zeigt leicht (Übung!), dass die Familie $i = (i_s)_{s \in S}$ auch ein Homomorphismus ist. Also ist dieser bijektive Homomorphismus i ein Isomorphismus zwischen $\mathrm{Init}_\Sigma(E)$ und $T_\Sigma(\emptyset)/(\mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2)$. □

Abschließend zeigen wir, dass man $\mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2$, d.h. die Einschränkung von $\mathrm{Th}(E)$ auf Grundterme, charakterisieren kann als kleinste Kongruenz auf $T_\Sigma(\emptyset)$, die E enthält.

Theorem 5.11 *Sei E eine Menge von Gleichungen über einer vernünftigen Signatur Σ. Die Menge der Grundinstanzen von E ist definiert als*

$$\mathrm{GI}_\Sigma(E) := \{\, t_1[\sigma] = t_2[\sigma] \mid \sigma : \mathcal{V} \to T_\Sigma(\emptyset) \ \text{ und } \ \mathrm{t_1{=}t_2} \ \text{ in E}\} \;.$$

Dann ist $\mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2$ *die kleinste Kongruenz auf* $T_\Sigma(\emptyset)$*, die* $\mathrm{GI}_\Sigma(E)$ *als Teilmenge enthält, und somit*

$$\mathrm{Init}_\Sigma(E) \cong T_\Sigma(\emptyset)/\langle \mathrm{GI}_\Sigma(E)\rangle \;.$$

Beweis: Offensichtlich ist die Einschränkung von $\mathrm{Th}(E)$ auf Grundterme eine Kongruenz auf $T_\Sigma(\emptyset)$, die alle Grundinstanzen von E enthält. Also gilt $\langle \mathrm{GI}_\Sigma(E)\rangle \subseteq \mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2$, da $\langle \mathrm{GI}_\Sigma(E)\rangle$ die kleinste Kongruenz auf $T_\Sigma(\emptyset)$ ist, die alle Grundinstanzen von E enthält.
Es bleibt also die umgekehrte Mengeninklusion zu zeigen. Offensichtlich ist

$$\mathrm{Th}(E) \cap T_\Sigma(\emptyset)^2 = \mathrm{GI}_\Sigma(\mathrm{Th}(E)) \;.$$

Wir müssen also zeigen, dass $\mathrm{GI}_\Sigma(\mathrm{Th}(E)) \subseteq \langle \mathrm{GI}_\Sigma(E)\rangle$, d.h. dass

$$\forall \sigma : \mathcal{V} \to T_\Sigma(\emptyset).\ t[\sigma]\ \langle \mathrm{GI}_\Sigma(E)\rangle\ s[\sigma]$$

für alle $t = s$ in $\mathrm{Th}(E)$. Da $\mathrm{Th}(E)$ induktiv definiert werden kann durch die Regeln (Refl), (Symm), (Trans), (Congr), (Subst) und (Ax_E), beweisen wir diese Aussage mit Regelinduktion. Substitutionen der Gestalt $\sigma : \mathcal{V} \to T_\Sigma(\emptyset)$ bezeichnen wir im weiteren als *Grundsubstitutionen* und wir schreiben der Einfachheit halber R für $\langle \mathrm{GI}_\Sigma(E)\rangle$.

- Da $R \equiv \langle \mathrm{GI}_\Sigma(E)\rangle$ schon reflexiv, symmetrisch und transitiv ist, sind die fälle (Refl), (Symm) und (Trans) trivial.

- (Congr) Sei $f : s_1 \times \ldots \times s_n \to S$ eine Operation in Σ. Angenommen es gelte für $i = 1, \ldots, n$, dass $t_i[\sigma]\ R\ s_i[\sigma]$ für alle Grundsubstitutionen σ. Dann gilt aber auch für alle Grundsubstitutionen σ, daß $f(t_1, \ldots, t_n)[\sigma]\ R\ f(s_1, \ldots, s_n)[\sigma]$, da $f(t_1, \ldots, t_n)[\sigma] \equiv f(t_1[\sigma], \ldots, t_n[\sigma])$, $f(s_1, \ldots, s_n)[\sigma] \equiv f(s_1[\sigma], \ldots, s_n[\sigma])$ und $f(t_1[\sigma], \ldots, t_n[\sigma])\ R\ f(s_1[\sigma], \ldots, s_n[\sigma])$ da R eine Kongruenzrelation ist.

- (Subst) Sei $\tau : \mathcal{V} \to T_\Sigma(\mathcal{V})$. Angenommen $t_1[\sigma]\ R\ t_2[\sigma]$ für alle Grundsubstitutionen σ. Dann gilt $t_1[\tau][\sigma]\ R\ t_2[\tau][\sigma]$ für alle Grundsubstitutionen σ, da $t_i[\tau][\sigma] \equiv t_i[\sigma \circ \tau]$ für $i = 1, 2$, und ferner $\sigma \circ \tau$ eine Grundsubstitution ist, wenn σ eine Grundsubstitution ist.[2]

- (Ax_E) Es gilt $\mathrm{GI}_\Sigma(E) \subseteq \langle \mathrm{GI}_\Sigma(E) \rangle$, da $\langle \mathrm{GI}_\Sigma(E) \rangle$ nach Definition die kleinste Kongruenz ist, die $\mathrm{GI}_\Sigma(E)$ enthält. □

Dieser Satz besagt im wesentlichen, dass man jeden Beweis einer Gleichung zwischen Grundtermen aus der Axiomenmenge E in einen solchen Beweis umbauen kann, wo die Regel (Subst) bloß auf Axiome aus E angewendet wird. Pragmatisch ist dies jedoch wenig sinnvoll, da man die Regel (Subst) erst so spät wie möglich anwenden will, um unnötige Aufblähung von Termen zu vermeiden.

Informatikrelevanz und Beispiele

Wir haben den mathematischen Apparat der allgemeinen Algebra aus zwei Gründen entwickelt, nämlich um

(1) einen vollständigen Herleitungskalkül für Gleichungstheorien und

(2) initiale Algebren als Mittel zur Definition von Datenstrukturen

bereitzustellen. Die Relevanz von (1) ist evident. Punkt (2) bedarf vielleicht eher einer Erläuterung.

In der Theorie der *abstrakten Datentypen* werden Datenstrukturen als *termerzeugte Algebren* aufgefasst. Es erhebt sich nun die Frage, inwiefern man eine solche termerzeugte Algebra A mit endlichen Methoden beschreiben kann. Nun, zuerst einmal wird die Signatur Σ angegeben. Dann betrachtet man den eindeutig bestimmten Homomorphismus $h : T_\Sigma(\emptyset) \to A$ und beobachtet, dass $A \cong T_\Sigma(\emptyset)/\ker_h$ (vermittels des Isomorphismus $i_A : T_\Sigma(\emptyset)/\ker_h \to A : [t]_{\ker_h} \mapsto [\![t]\!]$). Man hat also A bis auf Isomorphie — und mehr interessiert ohnedies nicht — angegeben, wenn man die Kongruenz $\ker_h$ auf $T_\Sigma(\mathcal{V})$ angegeben hat. Da die Menge $\ker_h$ fast immer unendlich ist, kann man sie nicht gut durch Aufzählung angeben. Deshalb identifiziert man eine Relation R auf $T_\Sigma(\emptyset)$ mit $\ker_h = \langle R \rangle$. Auch dieses R wird meist noch unendlich sein. Sehr oft jedoch findet man eine endliche Menge E von Gleichungen über der Signatur Σ mit $\ker_h = \langle \mathrm{GI}_\Sigma(E) \rangle$. Somit kann man also die Datenstruktur A durch das Paar (Σ, E) vollständig beschreiben. Solche Paare (Σ, E) nennen wir algebraische Spezifikationen[3]. Die *Bedeutung* oder *Semantik* einer Spezifikation (Σ, E) ist die termerzeugte Σ–Algebra $\mathrm{Init}_\Sigma(E) = T_\Sigma(\emptyset)/\langle \mathrm{GI}_\Sigma(E) \rangle$. Es

[2] Wir schreiben $\sigma \circ \tau$ als Abkürzung für $\mathrm{Int}(\sigma) \circ \tau$.

[3] Einen Formalismus für die übersichtliche Mitteilung von Spezifikationen haben wir zu Anfang des 2. Vorlesungsteils über Allgemeine Algebra kennengelernt.

sei erwähnt, dass alle Gleichungen aus E automatisch in $T_\Sigma(\emptyset)/\langle \mathrm{GI}_\Sigma(E)\rangle$ gelten, da sie nach Theorem 5.7 in $\mathrm{Cl}_\Sigma(E)$ gelten und somit auch in der initialen Algebra $\mathrm{Init}_\Sigma(E)$, da diese isomorph zu der kleinsten Unteralgebra von $\mathrm{Cl}_\Sigma(E)$ ist.[4] *Also kann man, um Eigenschaften von* $\mathrm{Init}_\Sigma(E)$ *und darauf operierender Programme zu beweisen, sowohl die Gleichungen aus E als auch Terminduktion verwenden.* Dies haben wir ganz am Anfang der Vorlesung anhand des Datentypes der natürlichen Zahlen ausführlich demonstriert.

Wir betrachten nun einige Beispiele für initiale Algebren.

Beispiel 5.12 *Die natürlichen Zahlen* $\mathbb{N}$ *mit* 0 *und der üblichen Nachfolgerfunktion und Addition sind isomorph zur initialen Algebra der Spezifikation*

NAT–ADD $\equiv$	
sorts	nat
funcs	zero: $\to$ nat
	succ: nat $\to$ nat
	add: nat $\times$ nat $\to$ nat
axioms	$\mathrm{add}(x, \mathrm{zero}) = x$
	$\mathrm{add}(x, \mathrm{succ}(y)) = \mathrm{succ}(\mathrm{add}(x, y))$

Man zeigt leicht, dass (durch „Ausrechnen“) jeder Grundterm beweisbar gleich zu einem *kanonischen* Grundterm der Gestalt $\mathrm{succ}^n(\mathrm{zero})$ ist. Verschiedene kanonische Terme können mithilfe der Spezifikationsgleichungen nicht als gleich bewiesen werden, da sie im Standardmodell der natürlichen Zahlen verschieden interpretiert werden.

Beispiel 5.13 *Die freie Worthalbgruppe über einem nichtleeren Alphabet A ist isomorph zur initialen Algebra folgender Spezifikation*

sorts	word	
funcs	a : $\to$ word	(a $\in A$)
	conc : word $\times$ word $\to$ word	
axioms	$\mathrm{conc}(x,\mathrm{conc}(y,z)) = \mathrm{conc}(\mathrm{conc}(x,y),z)$	

Die Einschränkung, dass A nichtleer sei, war notwendig, weil ansonsten die Signatur nicht vernünftig wäre, da es keinen Grundterm gäbe.

Beispiel 5.14 *Der Ring* $\mathbb{Z}[X_1, \ldots, X_n]$ *der Polynome mit ganzzahligen Koeffizienten in n Variablen ist isomorph zur initialen Algebra derjenigen Spezifikation, die man aus der Spezifikation* RING_COMM *der kommutativen Ringe mit* 1 *erhält, indem man n frische Konstanten, also neue nullstellige Funktionssymbole, hinzunimmt, die den Variablen* $X_1, \ldots, X_n$ *des Polynomrings* $\mathbb{Z}[X_1, \ldots, X_n]$ *entsprechen.*

[4] Es sei als Übung empfohlen zu zeigen, dass, wenn eine Gleichung in A gilt, diese Gleichung auch in allen Unter– und Faktoralgebren von A gilt!

Beispiel 5.15 *Die Datenstruktur der Listen über einer nichtleeren Menge A erhält man als initiale Algebra der Spezifikation, die aus der Spezifikation* LIST *dadurch entsteht, dass man für jedes $a \in A$ eine Konstante* $\mathrm{c}_a : \to \mathrm{elem}$ *hinzunimmt.*

Beispiel 5.16 *Die Datenstruktur der endlichen Multimengen über einer nichtleeren Menge A erhält man dadurch, dass man zur Spezifikation* MONOID *für jedes $a \in A$ eine Konstante* $\mathrm{c}_a : \to \mathrm{elem}$ *sowie das Kommutativgesetz* $\mathrm{add}(x, y) = \mathrm{add}(y, x)$ *als zusätzliche Gleichung hinzunimmt. Wenn man noch zusätzlich die Gleichung* $\mathrm{add}(x, x) = x$ *zur Spezifikation hinzunimmt, erhält man als initiale Algebra die Datenstruktur der endlichen Teilmengen von A.*

Abschließend sei erwähnt, dass (Gleichungs)Logik und Terminduktion nicht (immer) ausreichen, um alle in der initialen Algebra einer Spezifikation gültigen Gleichungen herzuleiten. Betrachte man nämlich die Spezifikation PRIM_REC, die man aus der Spezifikation NAT dadurch erhält, dass man für jede primitiv rekursive Funktion(sdefinition) eine Konstante zusammen mit ihren definierenden Gleichungen hinzunimmt. Das initiale Modell von PRIM_REC ist isomorph zu der Algebra mit unterliegender Menge $\mathbb{N}$, wo jedes Funktionssymbol f als diejenige primitiv rekursive Funktion interpretiert wird, die durch die definierenden Gleichungen für f eindeutig bestimmt ist. Die Menge der in $\mathbb{N}$ gültigen Gleichungen zwischen primitiv rekursiven Funktionen kann *nicht* durch einen Algorithmus aufgezählt werden, da andernfalls das Halteproblem entscheidbar wäre (vgl. etwa die Vorlesung Informatik IV). Da aber die Menge der Gleichungen, die aus den Gleichungen der Spezifikation PRIM_REC mithilfe von Induktion (d.h. Terminduktion bzgl. zero und succ) hergeleitet werden können, durch einen Algorithmus aufgezählt werden können und in $\mathbb{N}$ gültig sind, folgt also, dass man auf diese Weise *nicht alle* wahren Gleichungen zwischen primitiv rekursiven Funktionen herleiten kann.

Inhaltsverzeichnis