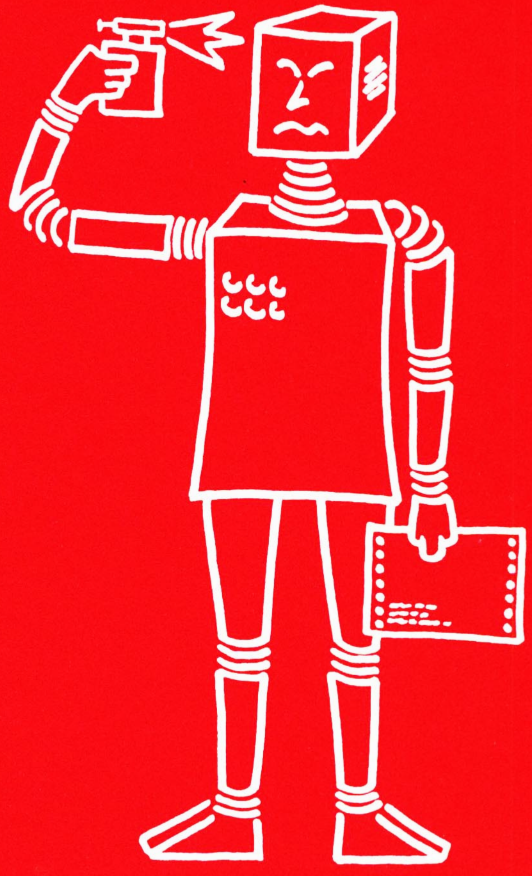
SEKI-PROJEKT

SEKI MEMO

Institut für Informatik III
Universität Bonn
Bertha-von-Suttner-Platz 6
D 5300 Bonn 1, W. Germany

Institut für Informatik I
Universität Karlsruhe
Postfach 6380
D-7500 Karlsruhe 1, W. Germany



MEMO SEKI-BN-81-05

Implementierungen Abstrakter Datentypen in terminaler Algebrasemantik

Erich Rome

Universität Bonn
Institut für Informatik III
Bertha-von-Suttner-Platz 6
D-5300 Bonn 1

IMPLEMENTIERUNGEN ABSTRAKTER DATENTYPEN IN TERMINALER ALGEBRASEMANTIK

von

ERICH ROME

MEMO SEKI-BN-81-05

November 1981

Abstract

In this paper we add a concept of implementation of abstract data types (adts) by adts in terminal algebra semantics to the notions of terminal algebra semantics of adts, which are introduced in [HR 79]. Our concept corresponds to that of initial implementations of adts described in [EKP 79a]. We give also a correctness proof of a T-implementation of stack(nat) by array(nat,nat) as an application example for the methods developed in [HR 79].
As in [EKP 79a], the implementing data type is extended in two steps, called SORT-IMPLEMENTATION and OPERATIONS-IMPLEMENTATION, to take over the tasks of the implemented type. These two steps are syntactical constructions. The semantics of the T-implementation concept is layed down in the EXTENSION- and RESTRICTION-step. A third step of the initial case, called IDENTIFICATION, is not necessary here for reasons which result from the use of the terminal algebra semantics and which are explained in this paper. As in [EKP 79a], the concept of T-implementation distinguishes syntax, semantics and correctness.

Keywords
Abstract data types, terminal algebra semantics, T-implementations, conditional axioms, correctness proofs, t-extensions, t-enrichments, consistency, completeness, t-consistency, t-completeness, context categories.

0. Inhaltsverzeichnis

1. Vorbemerkungen

1.1. Einleitung

Aus den verschiedenen Ansätzen zur Spezifikation abstrakter Datentypen haben sich die algebraischen Ansätze, welche einem syntaktischen Spezifikationsschema implizit eine Semantik, die als heterogene oder mehrsortige Algebra aufgefaßt wird, zuordnen, als die wichtigsten herausgeschält. Man kann diese impliziten Ansätze in zwei Gruppen aufteilen; es existiert zum einen

- die initiale Algebrasemantik abstrakter Datentypen, die vor allem auf Guttag [Gut 75], Liskov und Zilles [LZ 74], [Zil 74] und auf die ADJ-Gruppe [ADJ 77] zurückgeht

und zum anderen

- die terminale Algebrasemantik abstrakter Datentypen, wie sie von Wand in [Wan 78] und insbesondere von Hornung in [H 79] und von Hornung und Raulefs in [HR 79] beschrieben wurde.

Hornung betrachtet in [H 79] Spezifikationen abstrakter Datentypen, die gegenüber [ADJ 77] um bedingte Axiome erweitert wurden, überträgt die meisten in der initialen Algebrasemantik existierenden Konzepte in die terminale Algebrasemantik und fügt neue Ergebnisse für beide Ansätze hinzu.
In dieser Arbeit soll ein Konzept für die Implementierung von abstrakten Datentypen durch abstrakte Datentypen dem Ansatz der terminalen Algebrasemantik hinzugefügt werden. Als Vorlage dient das Implementierungskonzept, das Ehrig, Kreowski und Padawitz (EKP) in [EKP 79a] in initialer Algebrasemantik für Spezifikationen ohne bedingte Axiome entwickelt haben.
Nach den Vorbemerkungen des ersten Kapitels werden im zweiten Kapitel die mathematischen Grundkonzepte der initialen und terminalen Algebrasemantik angegeben, eine Notation für Spezi-

fikationen eingeführt und die Unterschiede des initialen und terminalen Ansatzes an einem einfachen Beispiel veranschaulicht. Im dritten Kapitel folgt auf die Übersicht über einige der vorhandenen Implementierungskonzepte dann die Angabe des Konzeptes der T-Implementierungen. Am Beispiel der T-Implementierung von stacks durch pointer/array-Paare wird außerdem verdeutlicht, wie der Begriffsapparat des terminalen Ansatzes "arbeitet". Im vierten Kapitel wird abschließend ein Vergleich des I-Implementierungskonzeptes von EKP mit dem T-Implementierungskonzept angestellt und auf noch zu lösende Aufgabenstellungen hingewiesen.

1.2. Technische Vorbemerkungen

Die in dieser Arbeit verwendete Notation wird in diesem Abschnitt vorgestellt.

1.2.1. Symbole für logische Operatoren

Es werden die folgenden Quantorsymbole verwendet werden:

$\forall$	Allquantor	"für alle ..."
$\exists$	Existenzquantor	"es gibt ein ..."
$\overset{1}{\exists}$	eindeutiger Existenzquantor	"es gibt genau ein ...".

Es werden folgende Symbole für logische Operatoren verwendet:

(,)	Klammern	
$\neg$	Negation	"nicht ..."
=>	Implikation	"wenn ... dann ..."
<=>	Äquivalenz	"... genau dann, wenn ..."
$\wedge$	Konjunktion	"... und ..."
$\vee$	Disjunktion	"... oder ..."
$\oplus$	Antivalenz	"entweder ... oder ..." .

Für logische Operatoren gilt in logischen Ausdrücken folgende Prioritätenliste:

(6)	(,)
(5)	$\neg$
(4)	=> und <=>
(3)	$\wedge$
(2)	$\vee$
(1)	$\oplus$,

wobei Operatoren mit höherer Priorität stärker binden als solche mit niedrigerer Priorität.

1.2.2. Mengensymbolik

Es bezeichnen

$\omega_0 := \{0,1,2,...\}$ die Menge der nicht-negativen ganzen Zahlen,

$\omega := \{1,2,3,...\}$ die Menge der positiven ganzen Zahlen.

Sei $n \in \omega$ und seien $A, A_1, ..., A_n$ Mengen. Dann bezeichnen

$\mathcal{P}(A) := \{B \mid B$ ist Teilmenge von $A\}$ die Potenzmenge von A,

$\bigtimes_{i=1}^{n} A_i := A_1 \times ... \times A_n := \{(a_1,...,a_n) \mid (\forall i \in \{1,...,n\})(a_i \in A_i)\}$

das kartesische Produkt von $A_1,...,A_n$,

$\bigcup_{i=1}^{n} A_i := A_1 \cup ... \cup A_n := \{a \mid (\exists i \in \{1,...,n\})(a \in A_i)\}$

die Vereinigungsmenge von $A_1,...,A_n$,

$\bigcap_{i=1}^{n} A_i := A_1 \cap ... \cap A_n := \{a \mid (\forall i \in \{1,...,n\})(a \in A_i)\}$

die Durchschnittsmenge von $A_1,...,A_n$,

$A^n := \{a_1...a_n \mid (\forall i \in \{1,...,n\})(a_i \in A)\}$

die Menge aller Worte über A der Länge n und zweideutig

$A^n := \{(a_1,...,a_n) \mid (\forall i \in \{1,...,n\})(a_i \in A)\}$

die Menge aller Listen über A der Länge n,

$A^+ := \bigcup_{n \in \omega} A^n$

die Menge der nicht-leeren Worte (Listen) über A und

$A^* := \bigcup_{n \in \omega_0} A^n$ die Menge aller Worte (Listen) über A.

Für n=0 sei

$(a_1,...,a_n) := ()$ die leere Liste (das leere Tupel),

$a_1...a_n := \lambda$ das leere Wort,

$\bigtimes_{i=1}^{n} A_i := \{()\}$ die Menge, die nur die leere Liste (das leere Tupel) enthält und

$\bigcup_{i=1}^{n} A_i := \bigcap_{i=1}^{n} A_i := \{1,...,n\} := \emptyset$ die leere Menge.

Ist I eine (Index-) Menge und sind für alle $i \in I$ die A_i Mengen, dann bezeichnet

$(A_i)_{i \in I}$ die aus den A_i gebildete (Mengen-) Familie.

Darüberhinaus bezeichnen wie üblich die Symbole

$\subseteq$ die Teilmengenbeziehung,

$\subset$ die echte Teilmengenbeziehung und

$\in$ die Elementbeziehung.

1.2.3. Relationen und Funktionen

Seien $A, A_1, \ldots, A_n$ Mengen.

$R \subseteq A \times A$ ist eine (binäre) Relation auf A. Für $(a,b) \in R$ wird auch aRb geschrieben. Sind R_1 und R_2 binäre Relationen auf A, dann wird für $aR_1b \wedge bR_2c$ auch aR_1bR_2c geschrieben. Ist R eine Äquivalenzrelation auf A, dann bezeichnet für alle $a \in A$ $[a]_R := \{b \mid b \in A \wedge aRb\}$ die Äquivalenzklasse von a bzgl. R.

Funktionen werden wie üblich notiert und $\bullet$ bezeichnet die Hintereinanderausführung von Funktionen.

$\_/i$ ist die i-te Projektionsfunktion, definiert durch

$$(\forall (a_1, \ldots, a_n) \in \mathop{\times}_{j=1}^{n} A_j)(\forall i \in \omega)((a_1, \ldots, a_n)/i := \begin{cases} a_i & \text{falls } 1 \leq i \leq n \\ \text{undefiniert} & \text{sonst} \end{cases}).$$

Als Schluß der technischen Vorbemerkungen wird noch die textuelle Substitution (eines Terms durch einen Term) erklärt. Seien t, t_1, t_2 Terme. Dann ist $t[t_2/t_1]$ der Term, den man erhält, wenn man jedes Auftreten von t_1 in t durch t_2 ersetzt.

2. Grundkonzepte

In diesem Abschnitt werden die für den Hauptteil der Arbeit benötigten Definitionen, Sätze und Lemmata, die größtenteils aus [H 79] und [HR 79] stammen, in einer vereinheitlichten Schreibweise zusammengestellt. Dabei sind zwei Abweichungen zu erwähnen:

- Bei der Definition der t-Spezifikation (Definition 30) wird nur die Existenz von zwei Konstanten der Sorte dis, nämlich tt und ff gefordert, wohingegen in [H 79] noch die Existenz einer dritten Konstante, $\perp_{dis}$ für das undefinierte Element der Sorte dis, gefordert wird. Dies soll die Beweise zum Implementierungsbeispiel überschaubarer machen.
- Für die Lesbarkeit der Beispielspezifikationen scheint es günstiger, die Bedingungen der Axiome auf die linke Seite zu schreiben statt wie in [H 79] auf die rechte. Der Unterschied ist formal in (3) der Definition 21 und in (4) der Definition 30 festgehalten. Die Beweise der Lemmata und Sätze verlaufen infolge dieser unwesentlichen Änderung jedoch völlig analog zu denen in [H 79].

Darüberhinaus wird eine Notation für Spezifikationen und Kombinationen angegeben und die Unterschiede der i- und t-Algebrasemantik werden an einem Beispiel aufgezeigt.

2.1. Die algebraischen Spezifikationen abstrakter Datentypen

Datentypen, wie sie der Informatiker und der Programmierer dauernd benutzen, bestehen aus einer oder mehreren Datenmengen und aus Operationen, die darauf "arbeiten". Dieser Tatbestand legt es nahe, Datentypen im mathematischen Sinn als heterogene (mehrsortige) Algebren aufzufassen.

Abstrakte Datentypen unterscheiden sich von einfachen Datentypen dadurch, daß sie

- allein durch die Operationen erzeugbar und
- unabhängig von einer konkreten Repräsentation ihrer Daten und Operationen sind.

2.1.1. Die initiale Algebrasemantik abstrakter Datentypen

Definition 1 (Signatur)

Eine Signatur ist ein Paar (S,Σ), wobei

S eine Menge von "Sorten" und

$\Sigma=(\Sigma_{w,s})_{w\in S^*,s\in S}$ eine Familie von paarweise disjunkten Mengen von Operationssymbolen ist.

Für $\sigma\in\Sigma_{s_1\ldots s_n,s}$ schreiben wir auch $\sigma:s_1 \ldots s_n \rightarrow s$.

Man sagt: σ hat die Stelligkeit $s_1\ldots s_n,s$, den Rang $s_1\ldots s_n$ und σ ist von der Sorte s. $\square$

Eine Signatur stellt mittels der Sorten Namen für Datenmengen und außerdem Namen für Operationen und deren Argument- und Wertebereich zur Verfügung. Sie bildet ein syntaktisches Schema für die Verschachtelung von formalen Operationsaufrufen und legt damit die Form der Algebren fest. Algebren, die in das von der Signatur Σ vorgegebene syntaktische Schema passen, heißen Σ-Algebren. Ihre Trägermengen A_s sind nach den Sorten $s\in S$ der Signatur und ihre Funktionen σ_A nach den Operationssymbolen der Signatur benannt.

Definition 2 (Σ-Algebra)

Sei (S,Σ) eine Signatur. Dann ist eine Σ-Algebra ein Paar $A = ((A_s)_{s\in S},\Sigma_A)$, wobei

$(A_s)_{s\in S}$ eine Familie von Mengen und

$\Sigma_A=\{\sigma_A \mid \sigma\in\Sigma_{w,s}, w\in S^*, s\in S\}$ eine Menge von Funktionen ist, sodaß $(\forall n\in\omega)(\forall s,s_1,\ldots,s_n\in S)(\forall\sigma\in\Sigma_{s_1\ldots s_n,s})(\sigma_A:A_{s_1}\times\ldots\times A_{s_n}\to A_s)$ und $(\forall s\in S)(\forall\sigma\in\Sigma_{\lambda,s})(\sigma_A:\ \to A_s)$, d.h. σ_A ist eine nullstellige Funktion nach A_s und bezeichnet ein Element aus dieser Menge. Daher ist $\Sigma_{\lambda,s}$ die Menge der (Namen der) Konstanten von A der Sorte s. $\square$

Die Operationssymbole einer Signatur kann man entsprechend ihrer Stelligkeit zusammensetzen und erhält dadurch die Σ-Terme. Ein Term ist von der Sorte $s\in S$, wenn das äußerste in ihm vorkommende Operationssymbol von der Sorte s ist. Die Menge der Σ-Terme einer Sorte s wird mit $T_{\Sigma,s}$ bezeichnet. Diese Mengen bilden die Datenmengen bzw. Trägermengen einer speziellen Σ-Algebra, nämlich der Σ-Termalgebra T_Σ.

<u>Definition 3</u> (Term,Termalgebra)

Sei (S,Σ) eine Signatur. Dann ist für alle $n\in\omega_0$ und $s\in S$ eine Menge $A^\Sigma_{s,n}$ folgendermaßen definiert:

(1) $(\forall s\in S)(A^\Sigma_{s,0}:=\Sigma_{\lambda,s})$ und

(2) $(\forall i\in\omega_0)(\forall s\in S)$

$$(A^\Sigma_{s,i+1}:=\{\sigma(a_1,\ldots,a_n) \mid n\in\omega \wedge s_1,\ldots,s_n\in S \wedge \sigma\in\Sigma_{s_1\ldots s_n,s} \wedge$$
$$(\forall 1\le j\le n)(a_j\in\bigcup_{k=0}^{i} A^\Sigma_{s_j,k})\}).$$

Dabei sind (,) und , Hilfssymbole, die nicht Element von $\bigcup_{w\in S^*,s\in S}\Sigma_{w,s}$ sind.

Dann ist für $s\in S$ $\quad T_{\Sigma,s} := \bigcup_{n\in\omega_0} A^\Sigma_{s,n}$.

Die durch (S,Σ) induzierte <u>Termalgebra</u> ist

$T_\Sigma := ((T_{\Sigma,s})_{s\in S}, \{\sigma_{T_\Sigma} \mid \sigma\in\Sigma_{w,s}, w\in S^*, s\in S\})$.

Hierbei ist für $n\in\omega_0$, $s,s_1,\ldots,s_n\in S$ und $\sigma\in\Sigma_{s_1\ldots s_n,s}$ σ_{T_Σ} die folgende Funktion:

$$\sigma_{T_\Sigma}:T_{\Sigma,s_1}\times\ldots\times T_{\Sigma,s_n}\to T_{\Sigma,s}$$
$$(t_1,\ldots,t_n)\mapsto\sigma(t_1,\ldots,t_n)\ .$$

Ein $t\in T_{\Sigma,s}$ mit $s\in S$ heißt <u>Term der Sorte s</u>. Für $\sigma\in\Sigma_{w,s}$ schreiben wir statt σ_{T_Σ} auch einfach σ_T. □

Offensichtlich ist T_Σ eine Σ-Algebra.

Daß Σ-Algebren auch Unteralgebren besitzen können, wird in der nächsten Definition erklärt.

<u>Definition 4</u> (Unteralgebra)

Seien (S,Σ), (S',Σ') Signaturen, A eine Σ-Algebra und A' eine Σ'-Algebra. A heißt <u>Unteralgebra</u> von A' wenn gilt:

(1) $S\subseteq S'$

(2) $(\forall s\in S)(A_s\subseteq A'_s)$

(3) $(\forall n\in\omega_0)(\forall s,s_1,\ldots,s_n\in S)(\forall\sigma\in\Sigma_{s_1\ldots s_n,s})$
$(\forall a\in A_{s_1}\times\ldots\times A_{s_n})(\sigma\in\Sigma'_{s_1\ldots s_n,s} \wedge \sigma_A(a)=\sigma_{A'}(a))$. □

<u>Definition 5</u> (Einschränkung, (S,Σ)-Anteil)

Sei (S',Σ') eine Signatur, A eine Σ'-Algebra. Sei (S,Σ) eine Signatur mit $S\subseteq S'$ und $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$. Dann ist die <u>Einschränkung von A auf (S,Σ)</u> bzw. der <u>(S,Σ)-Anteil von A</u>, $A|_\Sigma$, wie folgt definiert:

$A|_\Sigma := ((A_s)_{s\in S}, \{\sigma_A \mid \sigma\in\Sigma_{w,s}, w\in S^*, s\in S\})$. □

Offensichtlich ist $A|_\Sigma$ eine Σ-Algebra und Unteralgebra von A.

Im Folgenden sind unter anderem Strukturverträglichkeiten und Beziehungen zwischen Σ-Algebren von Interesse. Daher wird ein Homomorphiebegriff formuliert.

<u>Definition 6</u> (Σ-Homomorphismus)

Sei (S,Σ) eine Signatur und seien A und A' Σ-Algebren. Ein <u>Σ-Homomorphismus</u> von A nach A' ist eine Familie von Abbildungen $h=(h_s:A_s\to A'_s)_{s\in S}$, wobei gilt:

(1) $(\forall s\in S)(\forall\sigma\in\Sigma_{\lambda,s})(h_s(\sigma_A) = \sigma_{A'})$

(2) $(\forall n\in\omega)(\forall s,s_1,\ldots,s_n\in S)(\forall(a_1,\ldots,a_n)\in A_{s_1}\times\ldots\times A_{s_n})$

$$(\forall \sigma \in \Sigma_{s_1 \ldots s_n,s})(h_s(\sigma_A(a_1,\ldots,a_n)) = \sigma_{A'}(h_{s_1}(a_1),\ldots,h_{s_n}(a_n)))$$

Wir schreiben diesen Σ-Homomorphismus als $h:A \to A'$. □

Definition 7 ($g \bullet f$)

Sei (S,Σ) eine Signatur, seien A,B,C Σ-Algebren und $f:A \to B$, $g:B \to C$ Σ-Homomorphismen. Dann ist $g \bullet f$ die folgende Funktionenfamilie:

$$g \bullet f := (g_s \bullet f_s : A_s \to C_s)_{s \in S}. \ \square$$

Die Hintereinanderausführung von Σ-Homomorphismen ist selbst wieder ein Σ-Homomorphismus und verhält sich assoziativ.

Korollar 1

Die Funktionenfamilie aus Definition 7 ist ein Σ-Homomorphismus.

Beweis: Sh. [H 79], Seite 8, Beweis zu Korollar 1.1.1.8.. □

Lemma 1

Sei (S,Σ) eine Signatur, seien A,B,C,D Σ-Algebren und $f:A \to B$, $g:B \to C$, $h:C \to D$ Σ-Homomorphismen. Dann gilt

$$(h \bullet g) \bullet f = h \bullet (g \bullet f).$$

Beweis: Sh. [H 79], Seite 8, Beweis zu Lemma 1.1.1.9.. □

Es folgen einige Aussagen über spezielle Σ-Homomorphismen.

Definition 8 (Σ-Epimorphismus, Σ-Monomorphismus, id_A)

Sei (S,Σ) eine Signatur, seien A,B Σ-Algebren. Ein Σ-Homomorphismus $h:A \to B$ heißt

(1) Σ-Epimorphismus :<=> $(\forall s \in S)(h_s$ ist surjektiv)

(2) Σ-Monomorphismus :<=> $(\forall s \in S)(h_s$ ist injektiv).

Die Familie $id_A := (id_{A,s} : A_s \to A_s)_{s \in S}$ der identischen Abbildungen der Träger von A in sich ist ebenfalls ein Σ-Homomorphismus, für den wir $id_A : A \to A$ schreiben. □

<u>Definition 9</u> (Σ-Isomorphismus)

Sei (S,Σ) eine Signatur und seien A,B Σ-Algebren. Ein Σ-Homomorphismus $h:A \to B$ ist ein <u>Σ-Isomorphismus</u> genau dann, wenn es einen Σ-Homomorphismus $h^{-}:B \to A$ gibt, sodaß

$$h \bullet h^{-} = id_B \quad \text{und} \quad h^{-} \bullet h = id_A. \; \square$$

<u>Lemma 2</u>

Sei (S,Σ) eine Signatur und seien A,B Σ-Algebren. Für alle Σ-Homomorphismen $h:A \to B$ gilt:

h ist ein Σ-Isomorphismus <=> h ist Σ-Epimorphismus und h ist Σ-Monomorphismus.

Beweis: Sh. [H 79], S. 9f, Beweis zu Lemma 1.1.1.12.. $\square$

<u>Definition 10</u> (Alg_Σ)

Sei (S,Σ) eine Signatur. Dann ist Alg_Σ die Kategorie, deren Objekte gerade die Σ-Algebren und deren Morphismen gerade die Σ-Homomorphismen sind. Alg_Σ heißt die <u>Kategorie der Σ-Algebren</u>. $\square$

<u>Definition 11</u> (initiales, terminales Objekt)

Sei $C=(O,M)$ eine Kategorie mit Objektmenge O und Morphismenmenge M. Ein Objekt

$ci \in O$ heißt <u>initial</u> <u>(in C)</u> :<=> $(\forall c \in O)(\overset{1}{\exists} m \in M)(m:ci \to c)$

$ct \in O$ heißt <u>terminal</u> <u>(in C)</u> :<=> $(\forall c \in O)(\overset{1}{\exists} m \in M)(m:c \to ct)$. $\square$

<u>Satz 1</u>

Sei (S,Σ) eine Signatur. Dann ist T_Σ das bis auf Isomorphie eindeutige initiale Objekt in Alg_Σ. Dabei gibt es zu jedem $A \in Alg_\Sigma$ genau einen Σ-Homomorphismus $h_A:T_\Sigma \to A$.

Beweis: Sh. [ADJ 77], S.92ff. $\square$

Man kann also zeigen, daß es für jede Σ-Algebra A genau einen Σ-Homomorphismus $h:T_\Sigma \to A$ gibt. Die Termalgebra nimmt daher eine Sonderstellung unter den Σ-Algebren ein, denn diese Eigenschaft besagt, daß T_Σ initial in der Kategorie Alg_Σ ist, deren Objekte gerade die Σ-Algebren und deren Morphismen gerade die Σ-Homomorphismen sind. T_Σ ist bis auf Isomorphie eindeutig, d.h. alle anderen initialen Σ-Algebren gehen allein durch Umbenennungen aus T_Σ hervor; die Initialitätseigenschaft hängt also nicht von einer konkreten Darstellung der Daten ab. In diesem Sinn ist die Termalgebra "repräsentationsunabhängig". Im Sinne der Definition 15 ist sie auch noch "operationserzeugt". Damit erfüllt sie die beiden Anforderungen, die an einen abstrakten Datentyp zu Anfang gestellt wurden. Man kann also sagen:

"Termalgebren als initiale Algebren sind abstrakte Datentypen." (d.i. These 3 aus [Kre 78]). Dieses vorläufige Resultat wird im Folgenden noch verbessert werden. Als nächstes wird der Begriff der Σ-Kongruenz und der Quotientenalgebra eingeführt, deren Bedeutung jedoch erst weiter unten erläutert wird.

<u>Definition 12</u> (Σ-Kongruenz, Kon_A, SE)

Sei (S,Σ) eine Signatur und A eine Σ-Algebra. Eine <u>Σ-Kongruenz</u> auf A ist eine Familie $\equiv\ = (\equiv_s)_{s\in S}$ von Relationen, für die gilt:

(1) $(\forall s\in S)(\equiv_s$ ist Äquivalenzrelation auf A)

(2) $(\forall n\in\omega)(\forall s,s_1,\ldots,s_n\in S)(\forall\sigma\in\Sigma_{s_1\ldots s_n,s})$
$(\forall(a_1,\ldots,a_n),(b_1,\ldots,b_n)\in A_{s_1}\times\ldots\times A_{s_n})$
$((\forall 1\le i\le n)(a_i \equiv_{s_i} b_i) \Rightarrow \sigma_A(a_1,\ldots,a_n) \equiv_s \sigma_A(b_1,\ldots,b_n))$.

Die Menge aller Σ-Kongruenzen auf A wird mit Kon_A bezeichnet. (2) ist die <u>Substitutionseigenschaft</u> (<u>SE</u>) von $\equiv$. $\square$

<u>Lemma 3</u>

Sei (S,Σ) eine Signatur, A eine Σ-Algebra und K eine Teilmenge von Kon_A. Dann ist auch

$$\bigcap K := \{\equiv_s \mid s \in S \wedge \equiv_s = \bigcap_{k \in K} k_s\}$$

eine Σ-Kongruenz.

Beweis: Sh. [H 79], S. 11, Beweis zu Lemma 1.1.1.17.. □

Definition 13 (Quotientenalgebra)

Sei (S,Σ) eine Signatur, A eine Σ-Algebra und $\equiv$ eine Σ-Kongruenz auf A. Dann heißt

$A/\equiv \;:=\; ((A_s/\equiv)_{s \in S}, \{\sigma_{A/\equiv} \mid \sigma \in \Sigma_{w,s},\ w \in S^*,\ s \in S\})$

die Quotientenalgebra von A über $\equiv$. Hierbei ist

(1) $(\forall s \in S)(\ A_s/\equiv \;:=\; \{[a]_{\equiv_s} \mid a \in A_s\}\ )$

(2) $(\forall n \in \omega)(\forall s,s_1,\ldots,s_n \in S)(\forall \sigma \in \Sigma_{s_1 \ldots s_n,s})$

$(\forall (a_1,\ldots,a_n) \in A_{s_1} \times \ldots \times A_{s_n})$

$(\ \sigma_{A/\equiv}([a_1]_{\equiv_{s_1}},\ldots,[a_n]_{\equiv_{s_n}}) := [\sigma(a_1,\ldots,a_n)]_{\equiv_s}\ )$. □

Über den eindeutigen Σ-Homomorphismus $h_A : T_\Sigma \to A$ erzeugt jede Σ-Algebra A eine Σ-Kongruenz $\equiv_A$ auf T_Σ.

Definition 14 $(\equiv_A)$

Sei (S,Σ) eine Signatur und A eine Σ-Algebra. Dann bezeichnet $\equiv_A$ die Relationenfamilie auf T_Σ, die wie folgt durch den nach Satz 1 eindeutigen Σ-Homomorphismus $h_A : T_\Sigma \to A$ induziert wird:

$(\forall s \in S)(\forall t,t' \in T_{\Sigma,s})(\ t \equiv_{A,s} t' \iff h_{A,s}(t) = h_{A,s}(t')\ )$. □

Lemma 4

Sei (S,Σ) eine Signatur und A eine Σ-Algebra. Dann ist $\equiv_A$ eine Σ-Kongruenz.

Beweis: Folgt direkt aus Lemma 1.4.4.5. in [R 79]. □

Definition 15 (Σ-erzeugt)

Sei (S,Σ) eine Signatur und A eine Σ-Algebra. A heißt Σ-erzeugt, wenn der nach Satz 1 eindeutige Σ-Homomorphismus $h_A: T_\Sigma \to A$ ein Σ-Epimorphismus ist. $\square$

Bisher konnte man jede Operation nur in ihrer Stelligkeit definieren, aber meistens nicht ihr Verhalten beschreiben und auch nicht sagen, was eine Operation "macht". Dies soll nun durch Einführung von "Gleichungen" bzw. "Axiomen" erreicht werden. Zunächst braucht man zur Abkürzung von Termen bzw. zur Erstellung von Termschemata "Variablen". Jeder Sorte s wird eine Menge X_s von Variablenbezeichnern zugeordnet, welche als zusätzliche Konstantensymbole der Sorte s aufgefaßt werden. Die so entstandene Variablensignatur ermöglicht es, Terme mit Variablen zu erhalten.

Definition 16 (Variablensignatur, Variablentermalgebra)

Sei (S,Σ) eine Signatur. Dann ist $(S,\Sigma(X))$ die Variablensignatur zu (S,Σ) mit

(1) $X=(X_s)_{s\in S}$ ist eine Familie von abzählbaren unendlichen Mengen $X_s = \{x_s^1,\dots,x_s^n,\dots\}$

(2) $(\forall s\in S)(\ \Sigma(X)_{\lambda,s} := \Sigma_{\lambda,s} \cup X_s\ )$

$(\forall w\in S^*)(\forall s\in S)(\ \Sigma(X)_{w,s} := \Sigma_{w,s}\ )$.

Die durch $(S,\Sigma(X))$ induzierte Termalgebra $T_{\Sigma(X)}$ heißt die durch (S,Σ) induzierte Variablentermalgebra. Für $\sigma\in\Sigma_{w,s}$ mit $w\in S^*$ und $s\in S$ schreiben wir auch $\sigma_{T(X)}$ statt $\sigma_{T_{\Sigma(X)}}$. $\square$

Für Terme (mit Variablen) wird der Begriff des Subterms definiert und spezielle Arten von Subtermen benannt.

Definition 17 (Subterm, echter Subterm)

Sei (S,Σ) eine Signatur. Für jede Sorte $s\in S$ und für jeden Term $t\in T_{\Sigma,s}$ ist erklärt:

(1) Ein Subterm von t ist

(1.1) t selbst und

(1.2) jeder Subterm von $t_1,\dots,t_n$, wenn gilt

$$(\exists n\in\omega)(\exists s,s_1,\ldots,s_n\in S)(\exists\sigma\in\Sigma_{s_1\ldots s_n,s})$$
$$(\exists(t_1,\ldots,t_n)\in T_{\Sigma,s_1}\times\ldots\times T_{\Sigma,s_n})(\ t=\sigma(t_1,\ldots,t_n)\ ).$$

(2) Ein Subterm $t'\in T_{\Sigma,s'}$, $s'\in S$, von t heißt <u>echt</u> :<=> $t'\neq t$. □

Um bestimmen zu können, welche Variablen ein Term enthält, wird die Funktion var erklärt.

<u>Definition 18</u> (Variablenmenge, var)

Sei (S,Σ) eine Signatur. Die Funktion

$$\mathrm{var}:\bigcup_{s\in S} T_{\Sigma(X),s} \to \mathcal{P}(\bigcup_{s\in S} X_s)$$

ist folgendermaßen induktiv definiert:

(1) $(\forall s\in S)(\forall t\in\Sigma(X)_{\lambda,s})(\ \mathrm{var}(t):=\{t\}\cap X_s\ )$

(2) $(\forall n\in\omega)(\forall s,s_1,\ldots,s_n\in S)(\forall(t_1,\ldots,t_n)\in T_{\Sigma(X),s_1}\times\ldots\times T_{\Sigma(X),s_n})$
$(\forall\sigma\in\Sigma_{s_1\ldots s_n,s})(\ \mathrm{var}(\sigma(t_1,\ldots,t_n)):=\bigcup_{i=1}^{n}\mathrm{var}(t_i)\ ).$

Für $s\in S$, $t\in T_{\Sigma(X),s}$ heißt var(t) die <u>Variablenmenge</u> von t. □

<u>Definition 19</u> (Gleichungsterm, $Glt_{\Sigma,s}$)

Sei (S,Σ) eine Signatur. Dann ist für $s\in S$ $Glt_{\Sigma,s}$, die Menge der <u>Gleichungsterme</u> der Sorte s, wie folgt definiert:

$$Glt_{\Sigma,s} := T_{\Sigma(X),s}\times(\bigcup_{s'\in S}(T_{\Sigma(X),s'})^2)^* .$$

Für $t\in Glt_{\Sigma,s}$ mit $t=(t_1,\lambda)$ schreiben wir auch einfach t_1, d.h. wir identifizieren $T_{\Sigma(X),s}\times\{\lambda\}$ mit $T_{\Sigma(X),s}$.

Für $t\in Glt_{\Sigma,s}$ mit $t=(t_1,((l_1,r_1),\ldots,(l_n,r_n)))$ schreiben wir auch if $l_1\doteq r_1$ &...& $l_n\doteq r_n$ then t_1. □

Mittels der Gleichungsterme lassen sich Gleichungen zusammensetzen. Die Variablenmenge eines Gleichungsterms wird durch die Funktion vgl bestimmt.

<u>Definition 20</u> (vgl)

Sei (S,Σ) eine Signatur. Die Funktion var sei definiert wie in Definition 18. Dann ist die Funktion

$$vgl : \bigcup_{s\in S} Glt_{\Sigma,s} \to \mathcal{P}(\bigcup_{s\in S} X_s)$$

wie folgt definiert:

(1) $(\forall s\in S)(\forall t\in T_{\Sigma(X),s})(\ vgl((t,\lambda)) := var(t)\ )$

(2) $(\forall n\in\omega)(\forall s,s_1,\dots,s_n\in S)(\forall t\in T_{\Sigma(X),s})$

$(\forall(l_1,\dots,l_n),(r_1,\dots,r_n)\in T_{\Sigma(X),s_1}\times\dots\times T_{\Sigma(X),s_n})$

$$(vgl((t,((l_1,r_1),\dots,(l_n,r_n)))) := var(t)\cup\bigcup_{i=1}^{n}(var(l_i)\cup var(r_i))).$$

Für $s\in S$, $t\in Glt_{\Sigma,s}$ heißt $vgl(t)$ die <u>Variablenmenge des Gleichungsterms</u> t. □

Eine Signatur zusammen mit einer Menge von Gleichungen (Paaren von Gleichungstermen) bildet eine Spezifikation. Die zweite Komponente des "linken" Gleichungsterms wird, wenn es nicht das leere Wort ist, als Menge von "Bedingungen" aufgefaßt.

<u>Definition 21</u> (Spezifikation)

Eine <u>Spezifikation</u> ist ein Tripel (S,Σ,E) für das gilt:

(1) (S,Σ) ist eine Signatur,

(2) $E\subseteq\bigcup_{s\in S}(Glt_{\Sigma,s})^2$ ist die Menge der Axiome bzw. Gleichungen, die wir auch als Familie $E=(E_s)_{s\in S}$ auffassen,

(3) $(\forall(L,R)\in E)(\ R/2=\lambda\ )$.

Für $(L,R)\in E$ schreiben wir auch $L=R$.

Ist $SPEC=(S,\Sigma,E)$ eine Spezifikation, dann bezeichnet

- SSPEC die Sortenmenge von SPEC, d.h. es gilt SSPEC=S,
- ΣSPEC die Familie der Operationssymbolmengen von SPEC, d.h. es gilt $\Sigma SPEC=\Sigma$ und
- ESPEC die Axiomenmenge von SPEC, d.h. es gilt ESPEC=E. □

Zur schrittweisen Erweiterung von Spezifikationen wird der Begriff der Kombination eingeführt und zur Notationsvereinfachung werden die Operationen $\cup$ und $-$ für die Familien von Operationssymbolen erklärt.

Definition 22 ($\Sigma \cup \Sigma'$, $\Sigma - \Sigma'$, Kombination)

Sind (S,Σ) und (S',Σ') Signaturen, dann definieren wir:

(1) $\Sigma \cup \Sigma' := (\Sigma''_{w,s})_{w\in(S^* \cup S'^*),s\in(S\cup S')}$ mit

$(\forall w\in(S^* \cup S'^*))(\forall s\in(S\cup S'))$

$$(\Sigma''_{w,s} := \begin{cases} \Sigma'_{w,s} \cup \Sigma_{w,s} & \text{falls } w\in S^* \wedge s\in S \\ \Sigma'_{w,s} \cup \Sigma_{w,s} & \text{falls } w\in S'^* \wedge s\in S' \\ \emptyset & \text{sonst} \end{cases})$$

(2) $\Sigma - \Sigma' := (\Sigma'''_{w,s})_{w\in S^*,s\in S}$ mit

$(\forall w\in S^*)(\forall s\in S)$

$$(\Sigma'''_{w,s} := \begin{cases} \Sigma_{w,s} - \Sigma'_{w,s} & \text{falls } w\in S'^* \wedge s\in S' \\ \Sigma_{w,s} & \text{sonst} \end{cases}).$$

COMB:=SPEC+(S',Σ',E') heißt Kombination, wenn SPEC:=(S,Σ,E) und $(S\cup S', \Sigma\cup\Sigma', E\cup E')$ Spezifikationen sind. □

Nun wird beschrieben, wie unter Vorgabe einer Belegung der Variablen mit Trägerelementen einer Σ-Algebra die Gleichungsterme in dieser interpretiert werden.

Definition 23 (Belegung, Interpretationsfunktion)

Sei (S,Σ) eine Signatur, A eine Σ-Algebra. Eine Belegung durch A ist eine Funktion

$$b: \bigcup_{s\in S} X_s \rightarrow \bigcup_{s\in S} (A_s \cup X_s)$$

für die gilt:

$$(\forall s\in S)(\forall x\in X_s)(\ b(x)\in A_s \cup \{x\}\ ).$$

Die Menge aller Belegungen durch A wird mit Bel(A) bezeichnet.

Seien $t,t' \in \bigcup_{s\in S} Glt_{\Sigma,s}$. Dann ist

$Bel(A,t,t') := \{b\in Bel(A) \mid (\forall x\in vgl(t)\cup vgl(t'))(\ b(x)\neq x\ )\}$.

Sei $t'' \in \bigcup_{s\in S} T_{\Sigma(X),s}$. Dann ist

$Bel(A,t'') := \{b\in Bel(A) \mid (\forall x\in var(t''))(\ b(x)\neq x\ )\}$.

Sei b eine Belegung durch A. Dann induziert b folgendermaßen eine <u>Interpretationsfunktion</u>

$$int_b : \bigcup_{s \in S} Glt_{\Sigma,s} \rightarrow \bigcup_{s \in S} A_s \cup \{undef\} \quad :$$

(1) $(\forall s \in S)(\forall \sigma \in \Sigma_{\lambda,s})(\ int_b((\sigma_T,\lambda)) := \sigma_A\ )$

(2) $(\forall s \in S)(\forall x \in X_s)(\ int_b((x,\lambda)) := \begin{cases} b(x) & \text{falls } b(x) \neq x \\ undef & \text{sonst} \end{cases}\ )$

(3) $(\forall n \in \omega)(\forall s,s_1,\dots,s_n \in S)(\forall \sigma \in \Sigma_{s_1\dots s_n,s})$

$(\forall (t_1,\dots,t_n) \in T_{\Sigma(X),s_1} \times \dots \times T_{\Sigma(X),s_n})$

$(((\forall 1 \leq i \leq n)(int_b((t_i,\lambda)) \neq undef)) \Rightarrow$

$int_b((\sigma(t_1,\dots,t_n),\lambda)) := \sigma_A(int_b((t_1,\lambda)),\dots,int_b((t_n,\lambda)))\ \wedge$

$((\exists 1 \leq i \leq n)(int_b((t_i,\lambda)) = undef)) \Rightarrow$

$int_b((\sigma(t_1,\dots,t_n),\lambda)) := undef)$

(4) $(\forall n \in \omega)(\forall s,s_1,\dots,s_n \in S)(\forall r \in T_{\Sigma(X),s})$

$(\forall (l_1,\dots,l_n),(r_1,\dots,r_n) \in T_{\Sigma(X),s_1} \times \dots \times T_{\Sigma(X),s_n})$

$(((\forall 1 \leq i \leq n)(int_b((l_i,\lambda)) = int_b((r_i,\lambda)) \neq undef)) \Rightarrow$

$int_b(\text{if } l_1 \doteq r_1\ \&\dots\&\ l_n \doteq r_n \text{ then } r) := int_b((r,\lambda))\ \wedge$

$((\exists 1 \leq i \leq n)(\neg int_b((l_i,\lambda)) = int_b((r_i,\lambda)) \neq undef)) \Rightarrow$

$int_b(\text{if } l_1 \doteq r_1\ \&\dots\&\ l_n \doteq r_n \text{ then } r) := undef)$. □

<u>Lemma 5</u>

Sei (S,Σ) eine Signatur, A eine Σ-Algebra. Dann gilt

$(\forall s \in S)(\forall t \in T_{\Sigma(X),s})(\forall b,b' \in Bel(A))$

$((\forall x \in var(t))(b(x) = b'(x)) \Rightarrow int_b(t) = int_{b'}(t))$

Beweis: Sh. [H 79], Beweis von Lemma 1.1.2.5., S.17. □

Jetzt sind alle Voraussetzungen geschaffen, um sagen zu können, wann eine Σ-Algebra eine Spezifikation erfüllt.

<u>Definition 24</u> $((\Sigma,E)$-Algebra, SPEC-Algebra, SPEC-Anteil)

Sei (S,Σ,E) eine Spezifikation. Eine Σ-Algebra A <u>erfüllt</u> (S,Σ,E) bzw. ist eine <u>(Σ,E)-Algebra</u>, wenn gilt:

(1) A ist Σ-erzeugt

(2) $(\forall (L,R) \in E)(\forall b \in Bel(A,L,R))(int_b(L) = int_b(R) \vee int_b(L) = undef)$

(d.h. A erfüllt die Axiome).

Ist SPEC der Name der Spezifikation (S,Σ,E), dann schreiben wir auch SPEC-Algebra statt (Σ,E)-Algebra.

Sind $SPEC=(S,\Sigma,E)$ und $SPEC'=(S',\Sigma',E')$ Spezifikationen mit $S\subseteq S'$, $E\subseteq E'$ und $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$ und ist A eine SPEC'-Algebra, dann wird die Einschränkung von A auf (S,Σ), $A|_\Sigma$, auch SPEC-Anteil von A genannt und mit $A|_{SPEC}$ oder $(A)_{SPEC}$ bezeichnet. □

Definition 25 $(Alg_{\Sigma,E})$

Sei (S,Σ,E) eine Spezifikation. Dann ist $Alg_{\Sigma,E}$ die Kategorie, deren Objekte die (Σ,E)-Algebren und deren Morphismen die Σ-Homomorphismen zwischen diesen Algebren sind. □

Die (Σ,E)-Algebren bilden also wieder eine eigene Kategorie, in der auch wieder ein initiales Objekt gesucht wird. Die Σ-Termalgebra scheidet aus, da sie i.a. die Gleichungen nicht erfüllt; sie bildet aber die Grundlage für das gesuchte Objekt.

Definition 26 $(\equiv_E)$

Eine Spezifikation (S,Σ,E) induziert auf T_Σ folgendermaßen eine Σ-Kongruenz $\equiv_E:=(\equiv_{E,s})_{s\in S}$:

Sei K_E die Menge aller Σ-Kongruenzen $\equiv\in Kon_{T_\Sigma}$, die die folgenden Bedingungen erfüllen:

(1) $(\forall(L,R)\in E)(\forall b\in Bel(T_\Sigma,L,R))(L/2=\lambda \;\Rightarrow\; int_b(L)\equiv int_b(R))$

(2) $(\forall n\in\omega)(\forall s,s_1,\dots,s_n\in S)(\forall(L,R)\in E)(\forall b\in Bel(T_\Sigma,L,R))$
$(\forall(l_1,\dots,l_n),(r_1,\dots,r_n)\in T_{\Sigma(X),s_1}\times\dots\times T_{\Sigma(X),s_n})$
$(\forall l,r\in T_{\Sigma(X),s})((L=\text{if } l_1\doteq r_1 \;\&\dots\&\; l_n\doteq r_n \text{ then } l \wedge R=(r,\lambda)\wedge$
$(\forall 1\leq i\leq n)(int_b(l_i)\equiv int_b(r_i)))\Rightarrow int_b(l)\equiv int_b(r))$.

Dann ist für alle $s\in S$ $\quad \equiv_{E,s}:=\bigcap_{\equiv\in K_E}\equiv_s$. □

Die Σ-Kongruenz $\equiv_E$ wird auch i-Kongruenzrelation oder einfach i-Kongruenz genannt.

Lemma 6

Sei (S,Σ,E) eine Spezifikation. Dann erfüllt $\equiv_E$ die Aussageformen (1) und (2) aus Definition 26.

Beweis: Völlig analog zum Beweis von Lemma 1.1.2.9. in [H 79], S. 19. □

Lemma 7

Seien $(S,\Sigma,E),(S',\Sigma',E')$ Spezifikationen mit $S\subseteq S'$, $E\subseteq E'$ und es gelte $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$. Dann gilt

$$\equiv_{E,s} \subseteq \equiv_{E',s} \quad \text{für alle } s\in S.$$

Beweis:

Völlig analog zum Beweis von Lemma 1.1.2.10. in [H 79], S. 19f. □

Definition 27 (Quotiententermalgebra)

Sei $SPEC=(S,\Sigma,E)$ eine Spezifikation. Dann heißt $T_{\Sigma,E}:=T_\Sigma/\equiv_E$ Quotiententermalgebra von SPEC. Für $T_{\Sigma,E}$ schreiben wir auch T_{SPEC} oder $T_{\Sigma,\equiv}$ oder $T_{\Sigma,\equiv_E}$. □

Satz 2

Sei (S,Σ,E) eine Spezifikation, A eine (Σ,E)-Algebra.

Dann gilt $\quad \equiv_{E,s} \subseteq \equiv_{A,s} \quad$ für alle $s\in S$.

Beweis:

Völlig analog zum Beweis von Satz 1.1.2.12 in [H 79], S. 20ff. □

Satz 3

Sei (S,Σ,E) eine Spezifikation. Dann ist $T_{\Sigma,E}$ initial in $Alg_{\Sigma,E}$, wobei für alle $A\in Alg_{\Sigma,E}$ der eindeutige Σ-Homomorphismus $h:T_{\Sigma,E}\to A$ ein Σ-Epimorphismus ist.

Beweis:

Völlig analog zum Beweis von Satz 1.1.2.13 in [H 79], S.23ff. □

Korollar 2

Sei (S,Σ,E) eine Spezifikation. Jedes initiale Objekt in $Alg_{\Sigma,E}$ ist Σ-isomorph zu $T_{\Sigma,E}$.

Beweis: Folgt direkt aus proposition 1 in [ADJ 77]. □

Für die Quotiententermalgebra gilt entsprechend das hinter Satz 1 für die Σ-Termalgebra Gesagte. Damit ist die folgende Definition gerechtfertigt.

<u>Definition 28</u> (i-abstrakter Datentyp, i-Semantik)

Sei (S,Σ,E) eine Spezifikation. Dann heißt $T_{\Sigma,E}$ der durch (S,Σ,E) definierte <u>i-abstrakte Datentyp</u> oder auch die <u>i-Semantik</u> von (S,Σ,E). □

Will man sichergehen, daß bei der Hinzunahme von Sorten, Operationssymbolen und Axiomen zu einer Spezifikation SPEC der bestehende i-abstrakte Datentyp T_{SPEC} erhalten bleibt, so fordert man, daß die neue Spezifikation SPEC' eine i-Erweiterung von SPEC ist. Ein Spezialfall der i-Erweiterung ist die i-Anreicherung, bei der keine neuen Sorten, sondern nur neue Operationssymbole und Axiome hinzugefügt werden. Die Eigenschaften der i-Konsistenz und der i-Vollständigkeit werden im nächsten Satz dazu verwendet, die i-Erweiterungen zu charakterisieren.

<u>Definition 29</u> (i-Erweiterung,i-Anreicherung,i-vollständig, i-konsistent)

SPEC=(S,Σ,E) und SPEC'=(S',Σ',E') seien Spezifikationen.

(i) SPEC' heißt <u>i-Erweiterung</u> von SPEC, wenn gilt

(1) $S\subseteq S'$

(2) $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$

(3) $E\subseteq E'$

(4) $T_{\Sigma,E}$ ist Σ-isomorph zu $T_{\Sigma',E'}|_\Sigma$.

(ii) SPEC' heißt <u>i-Anreicherung</u> von SPEC, wenn gilt

(5) $S=S'$

(6) SPEC' ist i-Erweiterung von SPEC.

(iii) SPEC' heißt <u>i-vollständig</u> auf SPEC gdw.

$(\forall s \in S)(\forall t \in T_{\Sigma',s})(\exists t' \in T_{\Sigma,s})(t \equiv_{E',s} t')$.

(iv) SPEC' heißt <u>i-konsistent</u> auf SPEC gdw.

$(\forall s \in S)(\forall t,t' \in T_{\Sigma,s})(t \equiv_{E',s} t' \Rightarrow t \equiv_{E,s} t')$. □

<u>Satz 4</u>

Seien SPEC=(S,Σ,E) und SPEC'=(S',Σ',E') Spezifikationen mit

(1) $S \subseteq S'$

(2) $(\forall w \in S^*)(\forall s \in S)(\Sigma_{w,s} \subseteq \Sigma'_{w,s})$

(3) $E \subseteq E'$.

Dann gilt

SPEC' ist i-Erweiterung von SPEC <=>

SPEC' ist i-vollständig und i-konsistent auf SPEC.

Beweis: Sh. [H 79], Beweis zu Satz 1.2.2.6., S.35ff. □

2.1.2. Die terminale Algebrasemantik abstrakter Datentypen

Analog zur initialen Algebrasemantik bzw. darauf aufbauend wird die terminale Algebrasemantik abstrakter Datentypen erklärt. Dabei betrachtet man nur solche Spezifikationen – die t-Spezifikationen – , die eine bestimmte elementare Sorte <u>dis</u> enthalten zusammen mit gewissen Operationssymbolen, darunter mindestens die beiden Konstanten tt und ff, und beschreibenden Gleichungen dieser Sorte. Sie dient zum "unterscheiden" (engl.: <u>dis</u>tinguish) von Termen. Der Unterschied zur i-Algebrasemantik manifestiert sich in der Generierung einer neuen Σ-Kongruenz. Initial gesehen werden zwei Terme t und t' gleicher Sorte zunächst grundsätzlich als verschieden angesehen. Sie können jedoch durch Axiome miteinander identifiziert wer-

den; in dem Fall sind t und t' i-kongruent.
Terminal gesehen werden t und t' zunächst grundsätzlich als (verhaltens-) gleich angesehen. Es wird dann untersucht, ob sie sich, eingesetzt in "Kontexte" der Sorte dis gleich – dann sind t und t' t-kongruent – oder unterschiedlich "verhalten". Hier deutet sich ein Bezug zum Modul-Konzept an, in dem zwei Moduln dann als gleich angesehen werden, wenn ihr Input- und Output-Verhalten gleich ist. Die Begriffe "Kontext" und "Verhalten" werden noch präzisiert werden.

Definition 30 (t-Spezifikation)

Eine Spezifikation (S,Σ,E) heißt t-Spezifikation, wenn gilt

(1) $dis \in S$

(2) $\Sigma_0 := \{tt, ff\} \subseteq \Sigma_{\lambda,dis}$

$\Sigma_1 := \{\neg\} \subseteq \Sigma_{dis,dis}$

$\Sigma_2 := \{\_AND\_\} \subseteq \Sigma_{dis\ dis,dis}$

(3) $E_d := \{\neg tt = ff,\ \neg ff = tt,\ tt \text{ AND } x^1_{dis} = x^1_{dis},$

$\qquad ff \text{ AND } x^1_{dis} = ff \quad \} \subseteq E$

(4) $(\forall (L,R) \in E)(L/2 \in ((T_{\Sigma(X),dis})^2)^*)$. □

Analog zur Kombination definiert man die t-Kombination.

Definition 31 (t-Kombination)

$COMB = SPEC + (S',\Sigma',E')$ heißt t-Kombination, wenn $SPEC = (S,\Sigma,E)$ und $(S \cup S', \Sigma \cup \Sigma', E \cup E')$ t-Spezifikationen sind. □

Unter den t-Spezifikationen sind solche von besonderem Interesse, die vollständig und konsistent sind, d.h. in denen jeder dis-Term entweder zu tt oder zu ff i-kongruent ist und in denen tt und ff nicht i-kongruent sind.

Definiton 32 (vollständig,konsistent)

Eine t-Spezifikation (S,Σ,E) heißt

(i) <u>vollständig</u> gdw. $(\forall t \in T_{\Sigma,dis})(\exists tv \in \{tt,ff\})(t \equiv_{E,dis} tv)$

(ii) <u>konsistent</u> gdw. $tt \not\equiv_{E,dis} ff$. □

Als nächstes wird der Begriff "Kontext" präzisiert.

<u>Definition 33</u> (Kontextkategorie C_Σ, Kontext)

Sei (S,Σ) eine Signatur. Die Kategorie C_Σ, genannt die <u>Kontextkategorie</u>, ist definiert durch

(1) Objektmenge ist $|C_\Sigma| := S$

(2) Morphismenmenge ist

$/C_\Sigma/ := \{ct: s \to s' \mid s,s' \in S$ und $ct \in T_{\Sigma(X),s'}$ und ct enthält genau eine Variable, und diese ist von der Sorte $s\}$

(3) Für je zwei Morphismen $ct_1: s_1 \to s_2$, $ct_2: s_2 \to s_3$ mit $s_1, s_2, s_3 \in S$ und $ct_i \in T_{\Sigma(X),s_{i+1}}$, $i=1,2$, ist deren Komposition definiert durch

$$ct_2 \bullet ct_1 : s_1 \to s_3 := ct_2[ct_1/x_{s_2}].$$

Man schreibt $(\forall s,s' \in S)(C_\Sigma(s,s') := \{ct: s \to s' \mid ct \in /C_\Sigma/\})$.

$C_\Sigma(s,s')$ ist die Menge der <u>Kontexte</u> der Sorte s' für Terme der Sorte s. □

Neben der i-Kongruenzrelation $\equiv_E$ induziert jede t-Spezifikation eine weitere Σ-Kongruenz auf T_Σ, die t-Kongruenzrelation $\sim_E$. Zwei Terme $t,t' \in T_{\Sigma,s}$, $s \in S$, verhalten sich gleich, wenn sie unter $\sim_E$ t-kongruent sind: $t \sim_{E,s} t'$.

<u>Definition 34</u> ($\sim_E$)

Sei SPEC=(S,Σ,E) eine t-Spezifikation. Die Familie $\sim_E := (\sim_{E,s})_{s \in S}$ von Relationen auf T_Σ ist definiert durch

$$(\forall s \in S)(\forall p,q \in T_{\Sigma,s})(p \sim_{E,s} q :\Leftrightarrow (\forall ct \in C_\Sigma(s,dis))(ct[p/x_s] \equiv_{E,dis} ct[q/x_s]))$$

und heißt die durch SPEC induzierte <u>t-Kongruenzrelation</u>. □

Lemma 8

Sei SPEC=(S,Σ,E) eine t-Spezifikation. Dann ist die durch SPEC induzierte t-Kongruenzrelation $\sim_E$ eine Σ-Kongruenz.

Beweis: Sh. [HR 79], Beweis zu Lemma 2.2.4.(1). $\square$

Die Beziehung zwischen den beiden von einer t-Spezifikation erzeugten Σ-Kongruenzen wird in Lemma 9 dargestellt.

Lemma 9

Sei (S,Σ,E) eine t-Spezifikation. Dann gilt

$$(\forall s\in S)(\equiv_{E,s} \subseteq \sim_{E,s}).$$

Beweis: Sh. [HR 79], Beweis zu Lemma 2.2.4.(2). $\square$

Lemma 10

Seien (S,Σ,E) und (S',Σ',E') vollständige und konsistente t-Spezifikationen mit S$\subseteq$S', E$\subseteq$E' und $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$.

Dann gilt $(\forall s\in S)(\forall t,t'\in T_{\Sigma,s})(t \sim_{E',s} t' \Rightarrow t \sim_{E,s} t')$.

Beweis: Sh. [H 79], Beweis zu Lemma 1.3.2.7., S.48f. $\square$

Mit der t-Kongruenzrelation erhält man aus T_Σ auch eine neue Quotiententermalgebra.

Definition 35 ($T_{\Sigma,\sim}$)

Sei SPEC=(S,Σ,E) eine t-Spezifikation. $T_{\Sigma,\sim}$ bezeichnet dann die Σ-Algebra $T_\Sigma/\sim_E$. Wir schreiben auch $T_{SPEC,\sim}$ oder $T_{\Sigma,\sim_E}$ für $T_{\Sigma,\sim}$. $\square$

Lemma 11

Sei (S,Σ,E) eine t-Spezifikation. Dann ist $T_{\Sigma,\sim}$ eine (Σ,E)-Algebra.

Beweis: Sh. [H 79], Beweis zu Lemma 1.3.2.9., S.50ff. $\square$

Definition 36 ($\sim_A$)

Sei (S,Σ,E) eine t-Spezifikation und A eine Σ-Algebra. Dann ist die Familie $\sim_A := (\sim_{A,s})_{s\in S}$ von Relationen auf T_Σ definiert durch

$$(\forall s\in S)(\forall p,q\in T_{\Sigma,s})(p \sim_{A,s} q :\Longleftrightarrow (\forall ct\in C_\Sigma(s,dis))(ct[p/x_s] \equiv_{A,dis} ct[q/x_s])).$$

$\sim_A$ heißt die durch A induzierte t-Kongruenz. □

<u>Lemma 12</u>

Sei (S,Σ,E) eine t-Spezifikation und A eine Σ-Algebra. Dann ist $\sim_A$ eine Σ-Kongruenz und es gilt $(\forall s\in S)(\equiv_{A,s} \subseteq \sim_{A,s})$.

Beweis:

Reflexivität, Symmetrie und Transitivität von $\sim_A$ ergeben sich direkt aus den entsprechenden Eigenschaften von $\equiv_A$. Die Substitutionseigenschaft, also Punkt (2) der Definition 12, zeigt man für $\sim_A$ völlig analog zu dem Nachweis ebendieser Eigenschaft für $\sim_E$ in [HR 79], Beweis zu Lemma 2.2.4.(1). Der Nachweis von $(\forall s\in S)(\equiv_{A,s} \subseteq \sim_{A,s})$ verläuft völlig analog zum Beweis von Lemma 2.2.4.(2) in [HR 79]. □

Für die vollständige und konsistente t-Spezifikation SPEC definiert man zwei neue Kategorien.

<u>Definition 37</u> $(t\text{-}Mod_{\Sigma,E}, t\text{-}Imp_{\Sigma,E})$

Sei (S,Σ,E) eine vollständige und konsistente t-Spezifikation.

(i) $t\text{-}Mod_{\Sigma,E}$ ist die wie folgt definierte Kategorie:

$|t\text{-}Mod_{\Sigma,E}| := \{A\in|Alg_{\Sigma,E}| \mid A_{dis}=\{tt_A,ff_A\} \wedge tt_A\neq ff_A\}$

sei die Objektmenge von $t\text{-}Mod_{\Sigma,E}$ und

$/t\text{-}Mod_{\Sigma,E}/ := \{h:A \to B \mid h$ ist Σ-Homomorphismus $\wedge$ $A,B\in|t\text{-}Mod_{\Sigma,E}|\}$

sei die Morphismenmenge von $t\text{-}Mod_{\Sigma,E}$.

(ii) $t\text{-}Imp_{\Sigma,E}$ ist die wie folgt definierte Kategorie:

(1) $(\forall A\in|Alg_\Sigma|)(A\in|t\text{-}Imp_{\Sigma,E}| :\Longleftrightarrow$

((a) A ist Σ-erzeugt $\wedge$

(b) $A_{dis}=\{tt_A, ff_A\} \wedge tt_A \neq ff_A \wedge$

(c) $(\forall s \in S)(\forall p,q \in T_{\Sigma,s})(p \sim_{E,s} q \Rightarrow p \sim_{A,s} q)$))

(2) Die Morphismenmenge von $t\text{-}Imp_{\Sigma,E}$ ist

$$/t\text{-}Imp_{\Sigma,E}/ := \{h: A \to B \mid h \text{ ist } \Sigma\text{-Homomorphismus } \wedge A,B \in |t\text{-}Imp_{\Sigma,E}|\}. \square$$

Anschaulich gesehen, bedeutet das:

Die Objekte von $t\text{-}Mod_{\Sigma,E}$ erfüllen die Axiome von (S,Σ,E) und modellieren den dis-Anteil von $T_{\Sigma,E}$.

Die Objekte von $t\text{-}Imp_{\Sigma,E}$ brauchen die Axiome nicht zu erfüllen; sie modellieren aber ebenfalls den dis-Anteil von $T_{\Sigma,E}$ und es gilt, daß zwei t-kongruente Terme demselben Element des Objektes entsprechen.

Lemma 13

(S,Σ,E) sei eine vollständige und konsistente t-Spezifikation.

Dann gilt $$|t\text{-}Mod_{\Sigma,E}| \subseteq |t\text{-}Imp_{\Sigma,E}| \quad .$$

Beweis: Sh. [HR 79], Beweis zu Lemma 2.3.2.. $\square$

Lemma 14

Für jede vollständige und konsistente t-Spezifikation (S,Σ,E) und für jede t-Implementation $A \in |t\text{-}Imp_{\Sigma,E}|$ gilt $\sim_E = \sim_A$.

Beweis: Sh. [HR 79], Beweis zu Lemma 2.3.3.. $\square$

Die terminale Quotiententermalgebra nimmt in den Kategorien der t-Modelle und der t-Implementationen eine Sonderstellung ein.

Satz 5

Sei (S,Σ,E) eine vollständige und konsistente t-Spezifikation. Dann ist $T_{\Sigma,\sim}$ sowohl in $t\text{-}Mod_{\Sigma,E}$ als auch in $t\text{-}Imp_{\Sigma,E}$ terminal.

Beweis: Sh. [HR 79], Beweis zu Theorem 2.3.4.. $\square$

Korollar 3

Sei (S,Σ,E) eine vollständige und konsistente t-Spezifikation. Jedes terminale Objekt in t-$Mod_{\Sigma,E}$ bzw. t-$Imp_{\Sigma,E}$ ist Σ-isomorph zu $T_{\Sigma,\sim}$.

Beweis:

Sei A terminales Objekt in t-$Mod_{\Sigma,E}$. Dann gibt es genau einen Σ-Homomorphismus $h:T_{\Sigma,\sim} \to A$ und da $T_{\Sigma,\sim}$ terminal in t-$Mod_{\Sigma,E}$ ist gibt es auch genau einen Σ-Homomorphismus $h':A \to T_{\Sigma,\sim}$. Dann sind auch $h \bullet h':A \to A$ und $h' \bullet h:T_{\Sigma,\sim} \to T_{\Sigma,\sim}$ Σ-Homomorphismen nach Korollar 1. Mit h und h' sind $h \bullet h'$ und $h' \bullet h$ ebenfalls eindeutig, d.h. es gilt $h \bullet h'=id_A$ und $h' \bullet h=id_{T_{\Sigma,\sim}}$. Nach Definition 9 sind h und h' Σ-Isomorphismen. □

Auch $T_{\Sigma,\sim}$ ist repräsentationsunabhängig und operationserzeugt und erfüllt damit die Anforderungen, die an einen abstrakten Datentyp gestellt werden. Man definiert daher:

Definition 38 (t-abstrakter Datentyp, t-Implementation, t-Modell)

Sei $SPEC=(S,\Sigma,E)$ eine vollständige und konsistente t-Spezifikation. Dann heißt $T_{\Sigma,\sim}$ der durch SPEC definierte t-abstrakte Datentyp bzw. die t-Semantik von SPEC.
Eine Σ-Algebra, die Objekt von t-$Imp_{\Sigma,E}$ ist, heißt t-Implementation von $T_{\Sigma,\sim}$ und, wenn sie Objekt von t-$Mod_{\Sigma,E}$ ist, auch t-Modell von $T_{\Sigma,\sim}$. □

Es gibt t-Spezifikationen, deren i- und t-Kongruenzrelation gleich sind. Diese werden kategorisch genannt.

Definition 39 (kategorisch)

Eine t-Spezifikation (S,Σ,E) heißt kategorisch $:\Leftrightarrow \equiv_E = \sim_E$. □

Analog zum initialen Fall werden die Begriffe t-Erweiterung, t-Anreicherung, t-vollständig und t-konsistent definiert und charakterisiert.

Definition 40 (t-Erweiterung, t-Anreicherung, t-vollständig, t-konsistent)

$SPEC=(S,\Sigma,E)$ und $SPEC'=(S',\Sigma',E')$ seien t-Spezifikationen.

(i) SPEC' heißt t-Erweiterung von SPEC, wenn gilt

(1) $S\subseteq S'$

(2) $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$

(3) $E\subseteq E'$

(4) $T_{\Sigma,\sim}$ ist Σ-isomorph zu $T_{\Sigma',\sim}|_{\Sigma}$.

(ii) SPEC' heißt t-Anreicherung von SPEC, wenn gilt

(5) $S=S'$

(6) SPEC' ist t-Erweiterung von SPEC.

(iii) SPEC' heißt t-vollständig auf SPEC gdw.

$(\forall s\in S)(\forall t\in T_{\Sigma',s})(\exists t'\in T_{\Sigma,s})(t \sim_{E',s} t')$.

(iv) SPEC' heißt t-konsistent auf SPEC gdw.

$(\forall s\in S)(\forall t,t'\in T_{\Sigma,s})(t \sim_{E,s} t' \Rightarrow t \sim_{E',s} t')$. □

Satz 6

Seien $SPEC=(S,\Sigma,E)$ und $SPEC'=(S',\Sigma',E')$ vollständige und konsistente t-Spezifikationen mit

(1) $S\subseteq S'$

(2) $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$

(3) $E\subseteq E'$.

Dann gilt

SPEC' ist t-Erweiterung von SPEC <=>

SPEC' ist t-vollständig und t-konsistent auf SPEC.

Beweis: Sh. [HR 79], Beweis zu Theorem 2.5.5.. □

Es folgen drei technische Korollare. Für diese seien $SPEC=(S,\Sigma,E)$ und $SPEC'=(S',\Sigma',E')$ t-Spezifikationen mit

(1) $S\subseteq S'$

(2) $(\forall w\in S^*)(\forall s\in S)(\Sigma_{w,s}\subseteq\Sigma'_{w,s})$

(3) $E\subseteq E'$.

Korollar 4

Wenn SPEC' vollständig ist, dann ist SPEC' auch i-vollständig auf SPEC bzgl. dis.

Beweis:

Es gilt $(\forall t' \in T_{\Sigma',dis})(\exists tv \in \{tt,ff\})(t' \equiv_{E',dis} tv)$ und wegen $\{tt,ff\} \subseteq T_{\Sigma,dis}$ gilt die Behauptung. □

Korollar 5

SPEC' ist t-konsistent auf SPEC bzgl. dis.

Beweis:

Seien $p,q \in T_{\Sigma,dis}$ und es gelte $p \sim_{E,dis} q$. Wegen $\equiv_{E,dis} = \sim_{E,dis}$, $\equiv_{E',dis} = \sim_{E',dis}$ und $\equiv_{E,dis} \subseteq \equiv_{E',dis}$ (d.i. Lemma 7) gilt auch $p \sim_{E',dis} q$. □

Korollar 6

Wenn SPEC' i-vollständig auf SPEC ist, dann ist SPEC' auch t-vollständig auf SPEC.

Beweis: Folgt aus Lemma 9. □

Für spätere Induktionsbeweise wird die Gewichtsfunktion w eingeführt. Da das Symbol = bereits für Axiome verwendet wird, soll die Identität von Termen mit is bezeichnet werden.

Definition 41 (w(t), t is t')

Sei SPEC=(S,Σ,E) eine Spezifikation.

(1) Für jedes $s \in S$ und jeden Term $t \in T_{\Sigma,s}$ ist das Gewicht von t – i.Z. w(t) – wie folgt definiert:

(a) $(\forall s \in S)(\forall t \in \Sigma_{\lambda,s})(\ w(t) := 1\ )$

(b) $(\forall n \in \omega)(\forall s,s_1,\ldots,s_n \in S)(\forall \sigma \in \Sigma_{s_1\ldots s_n,s})$
$(\forall (t_1,\ldots,t_n) \in T_{\Sigma,s_1} \times \ldots \times T_{\Sigma,s_n})$
$(\ w(\sigma(t_1,\ldots,t_n)) := 1 + \sum_{i=1}^{n} w(t_i)\ ).$

(2) Sind $t,t' \in T_{\Sigma,s}$, $s \in S$, dann steht t is t' für "t ist identisch gleich t'". □

2.2. Notation für Spezifikationen und Kombinationen

Konkrete Beispielspezifikationen und -kombinationen werden in der Literatur über abstrakte Datentypen nahezu einheitlich notiert. Wir wollen die in [EKP 79a] verwendete Spezifikationssprache hier informell kurz vorstellen und leicht modifizieren.

Sei $SPEC=(S,\Sigma,E)$ eine Spezifikation und

$COMB=SPEC+(S',\Sigma',E')$ eine Kombination.

SPEC wird wie folgt aufgeschrieben.

SPEC:

$$
\begin{array}{lll}
\text{sorts:} & s_1,\ldots,s_n & \\
\text{opns:} & o_1:w_1 \rightarrow s_{j_1} & \text{abr: } (a_1) \\
 & \vdots & \quad\quad\vdots \\
 & o_m:w_m \rightarrow s_{j_m} & \quad\quad (a_m) \\
\text{eqns:} & E.1 \quad L_1=R_1 & \\
 & \vdots \quad\quad\quad \vdots & \\
 & E.k \quad L_k=R_k &
\end{array}
$$

Dabei gilt:

SPEC ist der Name einer Spezifikation, den wir als Wort aus $\{A,\ldots,Z\}^*\{0,\ldots,9\}^*$ auffassen. Oft wird statt dessen auch der durch SPEC spezifizierte Datentyp dort hingeschrieben. Datentypnamen sind doppelt unterstrichene Wörter aus

$\{a,\ldots,z\}^*\{0,\ldots,9\}^* \cup \{a,\ldots,z\}^*\{0,\ldots,9\}^*\{(\}\{a,\ldots,z\}^*\{0,\ldots,9\}^*\{)\}$,

z.B. set1(nat2) oder stack .

sorts: Hinter dieser Marke werden die Namen der Elemente von S (Sortennamen) durch Kommata voneinander getrennt aufgelistet. Sortennamen sind einfach unterstrichene Wörter aus $\{a,\ldots,z\}^*\{0,\ldots,9\}^*$.

opns: Hinter dieser Marke werden die Operationssymbole und ihre Stelligkeit aufgelistet. Namen von Operationssymbolen sind Wörter aus $\{A,\ldots,Z,\_\}^*\{0,\ldots,9\}^*$. Der Strich _ wird bei Infix-Operatoren verwendet, wie z.B. $\_AND\_:\underline{dis}\ \underline{dis} \rightarrow \underline{dis}$, und gibt an, wo später die Argumente stehen. Die Operationssymbole werden untereinander aufgelistet ohne Kommatrennung. Gibt es mehrere Operationssymbole mit derselben Stelligkeit, so braucht die Stelligkeit nur einmal angegeben zu werden; die Namen der zugehörigen Operationssymbole werden dann davor durch Komma getrennt aufgelistet, z.B. $\_AND\_,\_EQV\_,\_OR\_,\_XOR\_:\underline{dis}\ \underline{dis} \rightarrow \underline{dis}$.

abr: Hinter dieser Marke können für die in derselben Zeile davorstehenden Operationssymbole Abkürzungen angegeben werden, die in runden Klammern stehen. Für eine Liste von n Operationssymbolen gleicher Stelligkeit muß eine Liste von n Abkürzungen angegeben werden (if any). Abkürzungen können beliebige Zeichen sein, jedoch müssen bei Infix-Operatoren die underscore-Striche wieder angegeben werden. Für das Beispiel unter opns: könnten die Abkürzungen so aussehen: $(\_\wedge\_,\_<\doteq>\_,\_\dot{\vee}\_,\_\oplus\_)$.

eqns: Hinter dieser Marke werden die Elemente von E untereinander aufgelistet. Die Axiome werden wie folgt durchnumeriert: Ist E der Bezeichner der Axiomenmenge und besteht dieselbe aus k Elementen, dann fangen die Zeilen hinter der Marke eqns: mit E.1,...,E.k an (s.o.).

COMB wird wie folgt aufgeschrieben.

COMB: SPEC +

sorts: $s'_1,\ldots,s'_l$

opns: $o'_1:w'_1 \rightarrow s'_{j_1}$ abr: (a'_1)

$$\begin{array}{llll} & \vdots & & \vdots \\ & o'_r : w'_r \to s'_{j_r} & & (a'_r) \\ \text{eqns:} & E'.1 \quad L'_1 = R'_1 & & \\ & \vdots \qquad\quad \vdots & & \\ & E'.v \quad L'_v = R'_v & & \end{array}$$

Für COMB gilt dasselbe wie für SPEC.

Ist die Sorten-, Operationssymbol- oder Gleichungsmenge einer Spezifikation oder Kombination leer, so schreiben wir an den entsprechenden Stellen no sorts, no opns oder no eqns. Wird für eine Operation keine Abkürzung gewünscht, lassen wir in der entsprechenden Zeile unter abr: einfach freien Raum. Werden überhaupt keine Abkürzungen gewünscht, entfällt die Spalte abr: ganz.

2.3. Ein Beispiel für die Unterschiede der initialen und terminalen Algebrasemantik abstrakter Datentypen

Der Unterschied zwischen initialer und terminaler Algebrasemantik ist der Unterschied zwischen der i- und der t-Kongruenzrelation. Wie bereits in 2.1.2. gesagt, sind zwei Terme t und t' gleicher Sorte dann i-kongruent, wenn ihre Gleichheit aus den Axiomen ableitbar ist und sie sind dann t-kongruent, wenn die dis-Terme, die durch Einsetzen von t und t' in denselben dis-Kontext entstehen, i-kongruent sind; letzteres muß für alle möglichen dis-Kontexte gelten.
Spezifikationen, die keine t-Spezifikationen sind, sollen ab jetzt i-Spezifikationen heißen.
Man kann sich fragen, wie i- und t-Spezifikationen aussehen

müssen, die (für eine bestimmte Sorte) "dasselbe" spezifizieren. Dazu ein Beispiel:

i-nat:

sorts: nat

opns: NULL: → nat abr: (∅)

SUCC:nat → nat (S)

PLUS:nat nat → nat (_+_)

eqns: i-E.1 $\emptyset + x^1_{nat} = x^1_{nat}$

i-E.2 $S(x^1_{nat}) + x^2_{nat} = S(x^1_{nat} + x^2_{nat})$

ist eine i-Spezifikation, deren i-Semantik $T_{i\text{-}nat}$ Σ-isomorph ist zu der Σ-Algebra $A=(\omega_0,\{0,S,+\})$, deren nat-Träger die Menge der nicht-negativen ganzen Zahlen ist und deren Funktionen die Konstante 0, die Nachfolgerfunktion $S:\omega_0 \to \omega_0$ und die Addition $+:\omega_0 \times \omega_0 \to \omega_0$ mit der üblichen Definition sind.
Eine t-Spezifikation, deren nat-Anteil, d.i. die Einschränkung auf die Sorte nat und die Operationssymbole NULL, SUCC und _PLUS_, ebenfalls zu A Σ-isomorph ist, sieht so aus:

t-nat:

sorts: dis,nat

opns: TT,FF: → dis abr: (tt,ff)

NOT:dis → dis (¬)

AND:dis dis → dis (_∧_)

NULL: → nat (∅)

SUCC:nat → nat (S)

PLUS:nat nat → nat (_+_)

EQN:nat nat → dis

eqns: t-E.1 $\neg tt = ff$

t-E.2 $\neg ff = tt$

t-E.3 $tt \wedge x^1_{dis} = x^1_{dis}$

t-E.4 $ff \wedge x^1_{dis} = ff$

t-E.5 $EQN(\emptyset,\emptyset) = tt$

t-E.6 $EQN(\emptyset,S(x^1_{nat})) = ff$

t-E.7 $EQN(S(x^1_{nat}),\emptyset) = ff$

t-E.8 $EQN(S(x^1_{nat}),S(x^2_{nat})) = EQN(x^1_{nat},x^2_{nat})$

t-E.9 $\emptyset + x^1_{nat} = x^1_{nat}$

t-E.10 $S(x^1_{nat}) + x^2_{nat} = S(x^1_{nat} + x^2_{nat})$

Es gilt also $T_{i\text{-}nat} \cong A \cong T_{t\text{-}nat}|_{\Sigma}$ mit

$\Sigma=(\{\underline{nat}\},\{\emptyset: \to \underline{nat}, S:\underline{nat} \to \underline{nat}, \_+\_:\underline{nat}\ \underline{nat} \to \underline{nat}\})$.

Man sollte hier keiner optischen Täuschung erliegen; wenn man den obligatorischen dis-Anteil vergißt, ist eine t-Spezifikation meist nicht größer als eine entsprechende i-Spezifikation. Es fällt auf, daß t-nat gegenüber i-nat die "zusätzliche Operation" EQN besitzt. Diese wird benötigt, um dis-Kontexte für nat-Terme zu bilden, welche etwa so aussehen:

$tt\ AND\ EQN(S(x^1_{nat}),S(S(\emptyset)))$ oder

$\neg\ EQN(S(\emptyset),x^1_{nat})\ AND\ EQN(S(S(S(\emptyset))),\emptyset)$,

d.h. sie enthalten genau eine Variable, und diese ist von der Sorte nat. Wäre EQN nicht vorhanden, dann gäbe es keine Kontexte für nat-Terme und daher wären alle nat-Terme zueinander t-kongruent. Die Axiome t-E.9 und t-E.10 hätten dann keinerlei Wirkung mehr. EQN unterteilt die nat-Terme in Klassen, die als kanonische Repräsentanten gerade die Terme $\emptyset, S(\emptyset), S(S(\emptyset)),\ldots, S^n(\emptyset),\ldots$ enthalten. Zur Bestimmung der durch eine t-Spezifikation erzeugten t-Kongruenzrelation brauchen nicht alle dis-Kontexte betrachtet zu werden, sondern nur die "contexts of interest". Dies sind im Falle von t-nat alle Kontexte, deren äußerstes auftretendes Operationssymbol EQN ist; weiß man z.B., daß $EQN(t,p) \equiv EQN(t',p)$ ist, dann gilt wegen der SE der i-Kongruenz $\equiv$ auch $\neg EQN(t,p) \equiv \neg EQN(t',p)$,

$tt\ AND\ EQN(t,p) \equiv tt\ AND\ EQN(t',p)$ usw.;

damit gilt also für alle Kontexte $ct \in C_{\Sigma}(nat,dis)$, die $EQN(x^1_{nat},p)$ als Subterm haben $ct[t/x^1_{nat}] \equiv ct[t'/x^1_{nat}]$. Dieses Ergebnis wird später noch in Lemma 17 ausgedrückt.

t-nat erzeugt wie jede t-Spezifikation zwei Σ-Kongruenzen, nämlich $\equiv_{t-E}$ und $\sim_{t-E}$. Bei genauerer Betrachtung fällt auf, daß die beiden Relationen identisch sind: $\equiv_{t-E} = \sim_{t-E}$. Also sind auch die i-Semantik und die t-Semantik von t-nat gleich: $T_{t-nat} = T_{t-nat,\sim}$. Solche Spezifikationen heißen kategorisch (vergl. Definition 39). Die kontexterzeugenden Operationen wie EQN werden im folgenden das Konzept der terminalen Implementierungen beeinflussen.

3. Implementierungen

Es existieren bereits verschiedene Implementierungskonzepte für algebraische Spezifikationen abstrakter Datentypen, von denen einige in den folgenden Abschnitten vorgestellt werden sollen. Es wird kurz auf die Konzepte aus [ADJ 77], [Ehr 78] und [GHM 76a] eingegangen und danach wird das Konzept aus [EKP 79a] etwas ausführlicher behandelt. Daran anschließend wird letzteres soweit möglich in die terminale Algebrasemantik übertragen. An einem Beispiel wird dann die Wirkungsweise des Konzeptes verdeutlicht.
In [ADJ 77],[Ehr 78] und [EKP 79a] wird jeweils von einer initialen Algebrasemantik ausgegangen, während in [GHM 76a] bereits informell eine "terminale Algebrasemantik" zugrundegelegt wird; Guttag, Horowitz und Musser sehen zwei Terme gleicher Sorte solange als gleich an, bis nachgewiesen ist, daß sie verschieden sind. Dies entspricht in der Tat der in Definition 34 eingeführten t-Kongruenzrelation $\sim_E$, jedoch fehlt in [GHM 76a] jegliche Präzisierung.

3.1. Übersicht über einige Implementierungskonzepte

Man unterscheidet generell zwischen zwei Arten der Implementierung abstrakter Datentypen, d.i.

- die Implementierung abstrakter Datentypen durch abstrakte Datentypen, manchmal auch <u>abstrakte Implementierung</u> genannt, und
- die Implementierung abstrakter Datentypen durch beliebige Typen, als da sind Programme, Maschinenzustände, abstrakte

Datentypen usw. Diese werden auch konkrete Implementierungen genannt.

Von den hier vorgestellten Implementierungskonzepten, die alle nur für Spezifikationen ohne bedingte Axiome formuliert sind, gehören [Ehr 78] und [EKP 79a] zur ersten und [ADJ 77] und [GHM 76a] zur zweiten Art.

Allen Konzepten gemein ist, daß sie beschreiben, wie Daten und Operationen eines Typs, der sich, anschaulich gesehen, auf einem "höheren" Abstraktionsniveau befindet, durch die Daten und Operationen eines Typs repräsentiert werden, der sich dementsprechend auf einem "niedrigeren" Abstraktionsniveau befindet.

Die Implementierungen der ersten Art können dazu dienen, aus einfachen Datentypen schrittweise immer "kompliziertere" Datentypen aufzubauen; eine Implementierung dieser Art kann also als refinement step aufgefaßt werden. In diesem Zusammenhang sind "Über-alles"-Implementierungen bzw. Compound-Implementierungen von Interesse, die mehrere solcher refinement steps IMPL1,...,IMPLn zusammenfassen zu IMPL=IMPL1•...•IMPLn. Damit beschäftigt sich Ehrich in [Ehr 78] und neuere Ergebnisse zu diesem Thema sind in [EKMP 80] zu finden.

Implementierungen der zweiten Art können benutzt werden, um algebraisch zu spezifizierende Software-Systeme während der Entwicklung teilweise zu testen.

3.1.1. ADJ

In [ADJ 77] wird die von einer Spezifikation $<S,\Sigma,\mathcal{E}>$ induzierte Termalgebra $T_{\Sigma,\mathcal{E}}$ durch eine "Implementierungsalgebra" B implementiert. Dabei ist B eine Σ'-Algebra, deren Trägerelemente konkrete Datenrepräsentationen wie Maschinenzustände oder primitive Datentypen sein können und deren Operationen auf diesen Datenrepräsentationen definierte Basisoperationen wie Maschinenoperationen, Basisinstruktionen und Programme sind.

Formal besteht eine Implementierung von $<S,\Sigma,\mathcal{E}>$ aus der Σ'-Algebra B, einem "derivor" d von Σ nach Σ' und einer Σ-Kongruenz $\equiv$, sodaß gilt:

$T_{\Sigma,\mathcal{E}}$ ist Σ-isomorph zu einer Unteralgebra von $(dB)/\equiv$,

wobei dB die "d-abgeleitete Algebra von B" ist.

Sind (S,Σ) und (S',Σ') Signaturen, dann ist ein <u>derivor</u> von Σ nach Σ' ein Paar $d=(f,(d_{w,s})_{w\in S^*,s\in S})$, wobei $f:S \to S'$ eine Funktion mit der Eigenschaft

$(\forall n\in\omega)(\forall s_1,\ldots,s_n\in S)(\ f(s_1\ldots s_n)=f(s_1)\ldots f(s_n)\ )$ und

$(d_{w,s}:\Sigma_{w,s} \to (T_{\Sigma'})_{f(w),f(s)})_{w\in S^*,s\in S}$ eine Familie ist.

$(T_{\Sigma'})_{f(w),f(s)}$ bezeichnet die Menge der Σ'-Terme der Sorte $f(s)$, die die Variablen $\{y_1,\ldots,y_n\}$ benutzen, wobei y_i von der Sorte $f(s_i)$ ist. Jedes Operationssymbol $\sigma\in\Sigma_{w,s}$ wird mittels einer abgeleiteten Operation $d_{w,s}(\sigma)$ mit geeigneter Stelligkeit dargestellt. Dann ist

$$dB = ((B_{f(s)})_{s\in S}, \{\sigma_{dB} \mid \sigma\in\Sigma_{w,s},\ w\in S^*,\ s\in S\})$$

die <u>d-abgeleitete Σ-Algebra von B</u>, wobei die Operation σ_{dB} von dB für $\sigma\in\Sigma$ definiert ist als $(d(\sigma))_B$, d.i. die "abgeleitete Operation" des Σ'-Terms $d(\sigma)$.

Bleibt der Begriff der abgeleiteten Operation zu klären.

Sei $t\in T_{\Sigma(X),s}$, $s\in S$, mit $var(t)=\{y_1,\ldots,y_n\}$, $n\in\omega$, und die y_i seien Variable der Sorte $s_i\in S$. Sei A eine Σ-Algebra und $b\in Bel(A,t)$ eine Belegung. Dann ist $int_b(t)\in A_s$. Wenn t fest ist und b variiert, dann erhält man eine Funktion $t_A:A_{s_1}\times\ldots\times A_{s_n} \to A_s$, die die <u>abgeleitete Operation von t in A</u> heißt, mit

wenn $b(y_i)=a_i\in A_{s_i}$ f.a. $i\in\{1,\ldots,n\}$, dann ist

$t_A(a_1,\ldots,a_n) = int_b(t)\in A_s$.

Da $T_{\Sigma,\mathcal{E}}$ Σ-isomorph zu einer Unteralgebra von $(dB)/\equiv$ sein darf und nicht zu der ganzen Algebra $(dB)/\equiv$ Σ-isomorph sein muß, ist es z.B. möglich, eine Spezifikation für nat mit den Operationssymbolen Ø,+,* und SUCC durch eine Algebra zu implementieren, deren Trägermenge die Menge der ganzen Zahlen ist und deren Operationen 0,+,* und SUCC mit den üblichen Definitionen sind.

Der Korrektheitsnachweis der Implementierung läßt sich laut ADJ in speziellen Fällen durch automatische Beweiser führen.

3.1.2. Ehrich

Ehrich behandelt in [Ehr 78] Implementierungen auf syntaktischer Ebene als Beziehung zwischen Spezifikationen. Das mit den Begriffen aus [ADJ 77] kompatible framework bildet eine Kategorie spec, deren Objekte Spezifikationen und deren Morphismen Paare von Abbildungen zwischen Sorten und Operationssymbolen sind, die also dem derivor von ADJ ähneln. Jedem Objekt von spec ist eine Kategorie beigeordnet, deren Objekte (Σ-) Algebren sind und deren Morphismen (Σ-) Homomorphismen zwischen diesen Algebren sind. In jeder dieser Kategorien gibt es ein bis auf Isomorphie eindeutiges initiales Objekt, das die Semantik der Spezifikation darstellt. Auf semantischer Ebene werden die Auswirkungen der Implementierung auf die initialen Algebren diskutiert.

Informell gesehen, implementiert eine Spezifikation D_1 eine Spezifikation D_0, wenn die Operationen von D_0 abgeleiteten Operationen (im Sinne von [ADJ 77], sh. 3.1.1.) in D_1 zugeordnet werden können, die das durch die Gleichungen ausgedrückte "Verhalten" von D_0 realisieren. Wenn man für die abgeleiteten

Operationen neue Operationssymbole und entsprechende definierende Gleichungen zu D_1 hinzufügt, dann erhält man eine Spezifikation D_2 und es gibt Morphismen sowohl von D_0 nach D_2 als auch von D_1 nach D_2.
Formal ist eine Implementierung von D_0 durch D_1 ein Tripel (D_2,f,t), wobei D_2 eine Spezifikation, $t:D_0 \to D_2$ eine "wahre Einbettung" und $f:D_1 \to D_2$ eine "Σ-Einbettung" ist, die "voll bzgl. t" ist (sh. [Ehr 78]). Das Hauptergebnis von [Ehr 78] betrifft die Komponierbarkeit solcher Implementierungen und besagt für Spezifikationen D_0, D_1, D_2:

> Wenn D_1 D_2 implementiert und wenn D_0 D_1 implementiert, dann implementiert D_0 auch D_2, d.h. es gibt eine "Über-alles"-Implementierung.

Es gibt jedoch kein allgemeines effektives Kompositionsverfahren für Implementierungen. Weiter wird ein praktisches Kompositionsverfahren angegeben, das pushouts benutzt und nicht-systematische Schritte enthält.
Die Frage der Korrektheit wird in [Ehr 78] nicht gestellt.

3.1.3. Guttag, Horowitz und Musser

Das sehr informell gehaltene Papier [GHM 76a] beschäftigt sich anhand von Beispielen mit der Frage, wie algebraische Axiomatisierung den Korrektheitsnachweis von Implementierungen abstrakter Datentypen verbessern kann und mit der Angabe von halb-automatischen Hilfsmitteln, die zur Automatisierung solcher Beweise und zur Erstellung einer direkten Implementierung verwendet werden können. Direkte Implementierungen erlauben begrenzte Tests während der Entwicklung von Software-Systemen.

Eine Implementierung eines Datentyps durch einen anderen ist aufgeteilt in einen

- representation-Teil,

  in dem eine Repräsentation der Werte des zu implementierenden Typs durch (Tupel von) Werte(n) des implementierten Typs angegeben wird, etwa

  SYMT(Stack[Mapping[Identifier,Attributelist]]) → Symboltable,

  und in einen

- programs-Teil,

  in dem "Programme" (d.s. Axiome), welche auf der Repräsentation arbeiten, für die Operationen des zu implementierenden Typs angegeben werden, etwa

  INIT = SYMT(PUSH(NEWSTACK,NEWMAP)).

Dabei wird davon ausgegangen, daß für beide Datentypen algebraische Spezifikationen existieren, die aus einem syntaktischen Teil, der die Operationen des Datentyps und ihre Stelligkeit, und einen semantischen Teil, der die die Operationen beschreibenden Axiome enthält, bestehen.

Die Korrektheit einer Implementierung kann nachgewiesen werden, indem man zeigt, daß jedes Axiom des semantischen Teils der Spezifikation des Datentyps durch die Programme erfüllt wird, etwa für das Axiom LEAVEBLOCK(INIT) = INIT:

```
LEAVEBLOCK(INIT) = LEAVEBLOCK(SYMT(PUSH(NEWSTACK,NEWMAP)))
                 = SYMT(REPLACE(NEWSTACK,NEWMAP))
                 = SYMT(REPLACE(STAK(NEWARRAY,Ø),NEWMAP))
                 = SYMT(STAK(ASSIGN(NEWARRAY,1,NEWMAP),1))
                 = SYMT(PUSH(NEWSTACK,NEWMAP))
                 = INIT.
```

Außerdem wird gezeigt, wie formale Deduktionen, die den Datentyp <u>boolean</u> benutzen, Interpretationen von Gleichheitsoperatoren auf Datentypen und <u>Datentyp-Invarianten</u> bei der Verifikation von Implementierungen verwendet werden können. Dabei ist eine Datentyp-Invariante eine Eigenschaft P, die für alle Werte des Datentyps gilt, etwa

$$P(symtab) = (\exists stk \in Stack)(\ symtab = SYMT(stk)\ )$$

mit $symtab \in Symboltable$.

Schließlich wird noch das Konzept der direkten Implementierungen eingeführt. Eine direkte Implementierung eines Datentyps T ist eine Implementierung, deren reprensentation-Teil eine Teilmenge des syntaktischen Teils der Spezifikation von T ist und deren programs-Teil eine Teilmenge des semantischen Teils der Spezifikation von T ist. So kann zum Beispiel der Datentyp boolean mit den Operationen TRUE, FALSE, AND, OR, NOT, EQV, IMP und XOR durch den Datentyp boolean mit den Operationen TRUE, FALSE, AND und NOT direkt implementiert werden. Häufig werden auch Baumstrukturen als direkte Implementierungen verwendet oder Darstellungen in LISP.

3.1.4. Ehrig, Kreowski und Padawitz

Ehrig, Kreowski und Padawitz verlangen in [EKP 79a] von einem Implementierungskonzept, daß es die "Spracheigenschaft" algebraischer Spezifikationen, welche abstrakte Datentypen syntaktisch definieren und deren Semantik automatisch festlegen, beachtet. Diese Forderung, die von keinem anderen Implementierungskonzept erfüllt wird, soll es ermöglichen, die EKP-Implementierungen als Konzept einer Beschreibungssprache für abstrakte Datentypen aufzufassen. In einem Anforderungskatalog, der hier wörtlich übernommen wird, wird etwas genauer erklärt, wie dies zu verwirklichen ist.

1. Die Implementierung eines abstrakten Datentyps ADT0 durch einen abstrakten Datentyp ADT1 soll syntaktisch als Implementierung ihrer Spezifikationen verwirklicht werden,

$$\text{SPEC1} \xrightarrow{\text{IMPLEMENTATION}} \text{SPEC0}$$

indem die SPEC0-Sorten und -Operationen aus entsprechenden Bestandteilen von SPEC1 zusammengesetzt werden.

2. Semantisch soll dadurch festgelegt sein, wie sich Berechnungen in ADT0 auf Berechnungen in ADT1 zurückführen lassen, wie also die (i-)Semantik von SPEC0 aus der von SPEC1 resultiert:

$$T_{\text{SPEC1}} \xrightarrow{\text{IMPLEMENTATION}} T_{\text{SPEC0}} \quad .$$

3. Dabei soll jeder Wert in ADT0 zusammengesetzt aus ADT1-Daten repräsentiert sein, verschiedene Werte verschieden, jedoch nicht unbedingt eindeutig (d.h. Mehrfach-Repräsentationen sind zugelassen). Auch brauchen nicht alle geeignet zusammengesetzten Daten aus ADT1 zu Daten aus ADT0 zu korrespondieren. Insbesondere dürfen durch das Zusammensetzen neuer Daten aus ADT1-Daten diese selbst nicht zerstört oder verändert werden.

4. Außerdem soll jede Anwendung einer Operation in ADT0 (d.h. jede <u>Berechnung</u>) bis auf Repräsentation von Argumenten und Werten gemäß 3. auf Anwendungen von Operationen in ADT1 zurückgeführt und aus diesen zusammengesetzt werden können.

Es kann jedoch eine genügend allgemeine Syntax einer Implementierung meist keine Semantik induzieren, die 2. bis 4. erfüllt, da diese Forderungen unentscheidbar sind. EKP lösen dieses Problem, indem sie zunächst einen schwachen Implementierungsbegriff angeben, der den syntaktischen Anforderungen genügt, der jedoch meist T_{SPEC1} nicht in ein zu T_{SPEC0} isomorphes, sondern

nur in ein homomorphes überführt. Dann werden semantische Zusatzforderungen angegeben, die eine schwache Implementierung erfüllen muß, um als eigentliche Implementierung zu gelten. Die Implementierungen genügen dann, wie gezeigt wird, auch den Anforderungen 2. bis 4. des Kataloges.
Es folgen die formalen Definitionen aus [EKP 79a].

VORAUSSETZUNG

SPEC0 und SPEC1 seien zwei Spezifikationen, die beide Extension einer gemeinsamen Unterspezifikation SPEC=(S,Σ,E) sind, d.h. sie sind syntaktisch Kombinationen

SPEC0= SPEC + $(S0,\Sigma 0,E0)$ und SPEC1= SPEC + $(S1,\Sigma 1,E1)$,

die die semantische Zusatzforderung erfüllen, daß die SPEC-Anteile ihrer (i-)Semantik mit der (i-)Semantik von SPEC übereinstimmen. □

Dies ist anschaulich damit zu begründen, daß SPEC0 und SPEC1 gemeinsame Teile haben können, die bei der Implementierung natürlich nicht neu realisiert werden müssen. Haben SPEC0 und SPEC1 keine gemeinsamen Teile, so ist SPEC=$(\emptyset,\emptyset,\emptyset)$ die leere Spezifikation.

Definition (Syntax der schwachen Implementierung) (3.2.)

IMPL = $(\Sigma S0,ES0,E\Sigma 0)$ ist eine schwache Implementierung von SPEC0 mit Hilfe von SPEC1, wenn

IMPL1= SPEC1 + $(S0,\Sigma S0,ES0)$ und IMPL2= IMPL1 + $(\emptyset,\Sigma 0,E\Sigma 0)$

Kombinationen sind. Dabei sind $\Sigma S0$ und ES0 die S0-Sorten implementierenden Operationen und Gleichungen und $E\Sigma 0$ die $\Sigma 0$-Operationen implementierenden Gleichungen. Dementsprechend wird der Schritt von SPEC1 zur ersten Implementierungsebene IMPL1 Sorten-Implementierung und der von IMPL1 zur zweiten Implementierungsebene IMPL2 Operationen-Implementierung genannt. □

Definition (Semantik der schwachen Implementierung) (3.4.)

Die Semantik der schwachen Implementierung IMPL=(ΣS0,ES0,EΣ0) von SPEC0 mit Hilfe von SPEC1 ist die aus der Semantik T_{SPEC1} von SPEC1 durch Anwendung der Konstruktionen EXTENSION, RESTRICTION und IDENTIFICATION resultierende SPEC0-Algebra B, wobei die einzelnen Schritte folgendermaßen definiert sind:

1. EXTENSION, aufgeteilt in SORT- und OP-EXTENSION, überführt T_{SPEC1} in die Semantik der Kombination IMPL1 und diese in die von IMPL2:

$$\begin{tikzcd} T_{SPEC1} \arrow[rr, "\text{EXTENSION}"] \arrow[dr, "\text{SORT-EXTENSION}"'] & & T_{IMPL2} \\ & T_{IMPL1} \arrow[ur, "\text{OP-EXTENSION}"'] & \end{tikzcd}$$

.

2. RESTRICTION, aufgeteilt in FORGETTING und REACHABILITY, überführt T_{IMPL2} in seinen SPEC0'-Anteil $T_{IMPL2|SPEC0'}$ mit SPEC0'=SPEC + (S0,Σ0,$\emptyset$) und bildet darin das Bild $B' = h(T_{\Sigma\cup\Sigma0})$ unter der Interpretation der $\Sigma\cup\Sigma$0-Terme $h:T_{\Sigma\cup\Sigma0} \to T_{IMPL2|SPEC0'}$, wobei B' eine SPEC0'-Unteralgebra von $T_{IMPL2|SPEC0'}$ wird:

$$\begin{tikzcd} T_{IMPL2} \arrow[rr, "\text{RESTRICTION}"] \arrow[dr, "\text{FORGETTING}"'] & & B' \\ & T_{IMPL2|SPEC0'} \arrow[ur, "\text{REACHABILITY}"'] & \end{tikzcd}$$

.

3. IDENTIFICATION faktorisiert B' nach der von den Gleichungen E0 erzeugten Kongruenz $\equiv_{E0}$ und liefert als Quotient eine SPEC0-Algebra $B = B'/\equiv_{E0}$:

$$B' \xrightarrow{\text{IDENTIFICATION}} B$$

. □

Die SORT-EXTENSION beschreibt, wie Daten, die zu S0-Sorten gehören, aus SPEC1-Daten zusammengesetzt werden.

Die OP-EXTENSION beschreibt, wie die Σ0-Operationen auf den zusammengesetzten Daten "arbeiten", d.h. wie sie darauf definiert sind.

Im FORGETTING-Schritt werden alle Sorten und Operationen, die nicht zu SPEC0 gehören, "vergessen". Dabei entsteht eine SPEC0'-Unteralgebra von T_{IMPL2}, die aber i.a. die Gleichungen E0 von SPEC0 noch nicht erfüllt.

Im REACHABILITY-Schritt werden dann noch überflüssige Daten entfernt, die zwar zu SPEC0-Sorten gehören, die aber nicht durch SPEC0-Operationen erzeugbar sind, d.h. die keine SPEC0-Daten repräsentieren. Die aus diesem Schritt entstandene Algebra $B' = h(T_{\Sigma \cup \Sigma 0})$, das Bild der Interpretation h, enthält nur noch die von SPEC0 aus in T_{IMPL2} erreichbaren (engl.: reachable) Daten.

Im letzten Schritt, IDENTIFICATION genannt, muß man dafür sorgen, daß man aus B' eine Algebra erhält, die auch die Gleichungen aus E0 erfüllt; denn es gibt in B' zwar geeignete Repräsentationen der SPEC0-Daten und es ist erklärt, wie die SPEC0-Operationen darauf arbeiten, doch erfüllt B' i.a. noch nicht die Gleichungen aus E0. In [EKP 79a] ist dies an einem Beispiel verdeutlicht. Dort werden Mengen von natürlichen Zahlen durch lineare Listen natürlicher Zahlen implementiert. Die Sorte <u>set</u> wird durch eine einfache kopierende Operation c:<u>string</u> → <u>set</u> aus der Sorte <u>string</u> aufgebaut. In der Operationen-Implementierung wird die leere Menge durch die leere Liste dargestellt ($\emptyset = c(\lambda)$) und das Einfügen von Elementen in eine Menge durch das Anhängen eines Eintrags an eine Liste dargestellt ($INSERT(x^1_{nat}, c(x^1_{str})) = c(ADD(x^1_{nat}, x^1_{str}))$). Die schwache Implementierung sieht also so aus:

<u>set(nat)</u>: <u>i-nat</u> +

  sorts: <u>set</u>

  opns: $\emptyset$: → <u>set</u>

INSERT:nat set → set

eqns: E0.1 $INSERT(x^1_{nat},INSERT(x^1_{nat},x^1_{set}))=INSERT(x^1_{nat},x^1_{set})$

E0.2 $INSERT(x^1_n,INSERT(x^2_n,x^1_s))=INSERT(x^2_n,INSERT(x^1_n,x^1_s))$

string(nat): i-nat +

sorts: string

opns: λ: → string

ADD:nat string → string

no eqns

string(nat) impl set(nat) by

sorts impl opns: c:string → set

no sorts impl eqns

opns impl eqns: E∑0.1 $\emptyset=c(\lambda)$

E∑0.2 $INSERT(x^1_n,c(x^1_{st}))=c(ADD(x^1_n,x^1_{st}))$.

Nach RESTRICTION liegt dann eine Algebra B' vor, in der aber noch nicht das durch die Axiome von set(nat) ausgedrückte Verhalten von Mengen realisiert ist: Die Reihenfolge des Einfügens von Elementen in Mengen ist irrelevant, die Reihenfolge des Anhängens von Einträgen an Listen jedoch nicht und außerdem kann eine Liste einen Eintrag, anders als eine Menge ein Element, mehrfach enthalten. Dieses zu erreichen ist nach EKP der Sinn des IDENTIFICATION-Schrittes, der die Algebra B' nach der aus den Gleichungen E0 erzeugten Kongruenz faktorisiert. Nach diesem Schritt liegt also eine SPEC0- bzw. set(nat)-Algebra $B = B'/\equiv_{E0}$ vor, die nach Konstruktion SPEC0 erfüllt. Meines Erachtens nach wird durch den IDENTIFICATION-Schritt etwas erzwungen, was auch auf anderem, natürlicherem Weg erreicht werden kann: Man braucht in dem Beispiel nur die Axiome

$c(ADD(x^1_{nat},ADD(x^1_{nat},x^1_{str})))=c(ADD(x^1_{nat},x^1_{str}))$ und

$c(ADD(x^1_{nat},ADD(x^2_{nat},x^1_{str})))=c(ADD(x^2_{nat},ADD(x^1_{nat},x^1_{str})))$

zu ES0 hinzuzufügen und kann dann auf den IDENTIFICATION -Schritt verzichten. Es kann leicht durch Induktion gezeigt werden, daß bereits nach RESTRICTION eine set(nat)-Algebra vorliegt. Auf diese Weise wird das Verhalten von Mengen in B' nicht erzwungen, sondern sozusagen ebenfalls implementiert.

Der schwache Implementierungsbegriff erfüllt die semantischen Bedingungen 2. bis 4. i.a. nicht. Daher werden semantische Zusatzforderungen gestellt.

Definition (Implementierung) (3.6.)

Eine Implementierung von SPEC0 mit Hilfe von SPEC1 ist eine schwache Implementierung IMPL=(ΣS0,ES0,EΣ0) dieser Spezifikationen, bei der zusätzlich gilt, daß

(i) ihre Semantik B isomorph ist zur Semantik von SPEC0: $B \cong T_{SPEC0}$,

(ii) die erste Implementierungsebene IMPL1=SPEC1+(S0,ΣS0,ES0) eine Extension (i-Erweiterung) von SPEC1 ist und

(iii) die zweite Implementierungsebene IMPL2=IMPL1+($\emptyset$,Σ0,EΣ0) ein Enrichment (i-Anreicherung) von IMPL1 ist. $\square$

Die Punkte (i) bis (iii) lassen sich als Korrektheitskriterien interpretieren:

(ii) besagt, daß die Sorten-Implementierung die Semantik von SPEC1 bewahrt,

(iii) besagt, daß die Operationen-Implementierung die Semantik von IMPL1 bewahrt und

(i) stellt sicher, daß die erzeugte Semantik B auch wirklich zu der von SPEC0, T_{SPEC0}, isomorph ist, d.h. mit ihr übereinstimmt.

EKP zeigen schließlich, daß ihr Implementierungsbegriff den Anforderungen 1. bis 4. des Kataloges genügt.

1. und 2. sind klar nach Definition von Syntax und Semantik der schwachen Implementierung und nach Definition der Implementierung. Zur Vorbereitung von 3. und 4. wird gezeigt, daß es einen Epimorphismus $REP: B' \to T_{SPEC0}$ gibt, der, anschaulich gesagt, die ADT1-Daten den ADT0-Daten zuordnet, die sie repräsentieren. Die Surjektivität von REP besagt, daß jeder SPEC0-Wert — eventuell mehrfach — in IMPL2 repräsentiert ist, und die Rechtseindeutigkeit von REP sichert, daß verschiedene Werte auch verschieden repräsentiert sind.
Die Korrektheitsbedingungen (ii) und (iii) erlauben darauf zu schließen, daß ADT1-Daten nicht zerstört werden und daß, da es zu jedem IMPL1-Term, der keine S0-Sorte verwendet, einen äquivalenten SPEC1-Term gibt (vergl. Satz 4 und Definition 29.(ii)), die Operationen aus $\Sigma S0$, die die S0-Sorten "aufbauen", tatsächlich nur ADT1-Daten (SPEC1-Terme) als Argumente benutzen. Damit ist 3. erfüllt.
Die Homomorphie-Eigenschaft von REP besagt zunächst nur, daß bis auf Datenrepräsentation jede Anwendung einer SPEC0-Operation (jede ADT0-Berechnung) auch in $B' \subseteq T_{IMPL2}$ ausgeführt werden kann:

$$REP(\sigma_{IMPL2}(b_1,\ldots,b_n)) = \sigma_{SPEC0}(a_1,\ldots,a_n)$$

mit $b_1,\ldots,b_n \in B'$ und $a_1,\ldots,a_n \in T_{SPEC0}$. Nach den Bemerkungen zu 3. läßt sich die Berechnung von $\sigma_{IMPL2}(b_1,\ldots,b_n)$ zunächst auf einen IMPL1-Term t' zurückführen, der neben SPEC0-Operationsaufrufen auch Subterme enthält, die keine S0-Sorten verwenden. Diese Subterme können nach den Bemerkungen zu 3. als Berechnungen in ADT1 aufgefaßt werden. Indem man nun schrittweise alle SPEC0-Operationsaufrufe in t' auf Berechnungen in ADT1 zurückführt, erhält man eine Gesamtberechnung von t' bzw. von $\sigma_{IMPL2}(b_1,\ldots,b_n)$, die bis auf Repräsentation nur noch aus Anwendungen von Operationen aus ADT1 besteht. Am Mengen/Listen-

Beispiel sieht dies wie folgt aus:

Die Berechnung von INSERT(n,∅) läßt sich schrittweise auf c(ADD(n,λ)) zurückführen:

INSERT(n,∅) ≡ INSERT(n,c(λ))

≡ c(ADD(n,λ)).

Dabei werden bis auf die Repräsentation c nur Operationsaufrufe (ADD, λ, n) verwendet, die zur <u>string(nat)</u>-Spezifikation gehören.

Damit ist klar, daß auch die vierte Anforderung erfüllt ist.

Die letzten beiden Kapitel von [EKP 79a] beschäftigen sich mit Korrektheitskriterien und Möglichkeiten der Sorten-Implementierung. Die Hauptergebnisse sollen hier noch aufgeführt werden. Zunächst wird der Begriff der RI-Korrektheit (RESTRICTION und IDENTIFICATION) definiert und charakterisiert.

<u>Definition</u> (RI-Korrektheit) (4.1.)

Eine schwache Implementierung IMPL von SPEC0 mit Hilfe von SPEC1 heißt <u>RI-korrekt</u>, wenn ihre Semantik B isomorph ist zur Semantik von SPEC0: $B \cong T_{SPEC0}$. □

<u>Theorem</u> (Charakterisierung der RI-Korrektheit) (4.3.)

Für eine schwache Implementierung IMPL ist jede der folgenden Aussagen äquivalent zur RI-Korrektheit:

1. B ist initiale SPEC0-Algebra
2. die Interpretation $I: T_{SPEC0} \rightarrow B$ ist injektiv
3. es gibt einen Homomorphismus $f': B' \rightarrow T_{SPEC0}$ (der dann automatisch surjektiv ist)
4. alle in IMPL2 äquivalenten $\Sigma \cup \Sigma 0$-Terme t und t' sind auch in SPEC0 äquivalent: $t \equiv_{E \cup E1 \cup ES0 \cup E\Sigma 0} t' \Rightarrow t \equiv_{E \cup E0} t'$.

Beweis: Sh. [EKP 79a], Beweis zu Theorem 4.3.. □

Die vierte Eigenschaft ist eine Art Konsistenzbedingung, die

als Ansatzpunkt für automatische Verifikationsverfahren dienen kann.
Damit ist die Eigenschaft (i) der Implementierung charakterisiert. Die Enrichment-Eigenschaft (iii) der Implementierung wird in [EKP 79a] nicht behandelt; es wird jedoch auf das Papier [EKP 78] verwiesen, wo die Enrichment-Eigenschaft ausführlich untersucht wird. [EKP 78] enthält jedoch noch Fehler, die aber in [EKP 79b] verbessert wurden.

Die Sorten-Implementierung einer schwachen Implementierung, die das folgende Lemma erfüllt, hat bereits die Implementierungseigenschaft (ii).

Lemma (5.1.)

Gilt bei einer schwachen Implementierung IMPL=(ΣS0,ES0,EΣ0) von SPEC0 mit Hilfe von SPEC1

(i) $s \in S0$ für alle $\sigma \in \Sigma S0_{w,s}$

(ii) $s \in S0$ für alle $(L,R) \in ES0_s$,

so ist die erste Implementierungsebene IMPL1=SPEC1+(S0,ΣS0,ES0) eine Extension von SPEC1.

Beweis: Sh. [EKP 79a], Beweis zu Lemma 5.1.. $\square$

Es wird also rein syntaktisch verlangt, daß die Sorten implementierenden neuen Operationen und Gleichungen nur von Sorten $s \in S0$, nicht aber von Sorten $s \in S \cup S1$ sind. Diese Bedingung ist für viele praktische Fälle hinreichend und umfaßt Sorten implementierende Operationen wie

a) kopieren: $c: s_1 \to s$ für $s_1 \in S \cup S1$, $s \in S0$

b) vereinigen: $\left.\begin{matrix} in_1 : s_1 \to s \\ \vdots \\ in_k : s_k \to s \end{matrix}\right\}$ für $s_1,\ldots,s_k \in S \cup S1$, $s \in S0$, $k \in \omega$

c) tupeln: $TUP: s_1 \ldots s_k \to s$ für $s_1,\ldots,s_k \in S \cup S1$, $s \in S0$, $k \in \omega$

d) tabellieren: NIL: $\rightarrow s$, TAB:s $s_1 \ldots s_k \rightarrow s$ $\Big\}$ für $s_1,\ldots,s_k \in S \cup S1$, $s \in S0$

e) verzweigen: EMPTY: $\rightarrow s$, BIN:s s $s_1 \ldots s_k \rightarrow s$ $\Big\}$ für $s_1,\ldots,s_k \in S \cup S1$, $s \in S0$.

Ein Beispiel für eine Implementierung von stacks durch pointer/array-Paare, wobei für die Sorten-Implementierung eine Operation _below_:array nat → stack vom Typ c) benutzt wird, findet sich im Anhang 2 von [EKP 79a].

Dieses Implementierungskonzept erscheint für die Übertragung in die terminale Algebrasemantik besonders geeignet, denn EKP zeigen in [EKP 79a], daß die Implementierungskonzepte in [Ehr 78], [EKP 78], [ADJ 78], [Wan 78], [GN 78] und [LS 77] mehr oder weniger nur Spezialfälle ihres Konzeptes sind.
In einem neuen Papier [EKMP 80] hat die Berliner Gruppe ihr Implementierungskonzept noch einmal überarbeitet und Untersuchungen angestellt, inwieweit eine Implementierung IMPL1 von ADT0 durch ADT1 und eine Implementierung IMPL2 von ADT1 durch ADT2 so komponiert werden können, daß eine Implementierung IMPL3 von ADT0 durch ADT2 entsteht.
Unter den schwachen Implementierungen werden durch semantische Zusatzforderungen <u>starke</u> Implementierungen und unter diesen wiederum <u>persistente</u> Implementierungen gekennzeichnet. Analog dazu wird die schwache, starke und persistente Komposition erklärt. Schwache und persistente Implementierungen sind unter der Komposition abgeschlossen, starke Implementierungen i.a. nicht. Schließlich werden mehrere schwache Implementierungen IMPLi:SPECi => SPEC(i-1), i=1,...,n, zur <u>Compound-Implementierung</u> COMPIMPL=(IMPL1,...,IMPLn) zusammengefaßt, wobei die Komposition IMPL=IMPL1•...•IMPLn wieder eine schwache Implementierung sein muß. COMPIMPL heißt schwach, stark bzw. persistent,

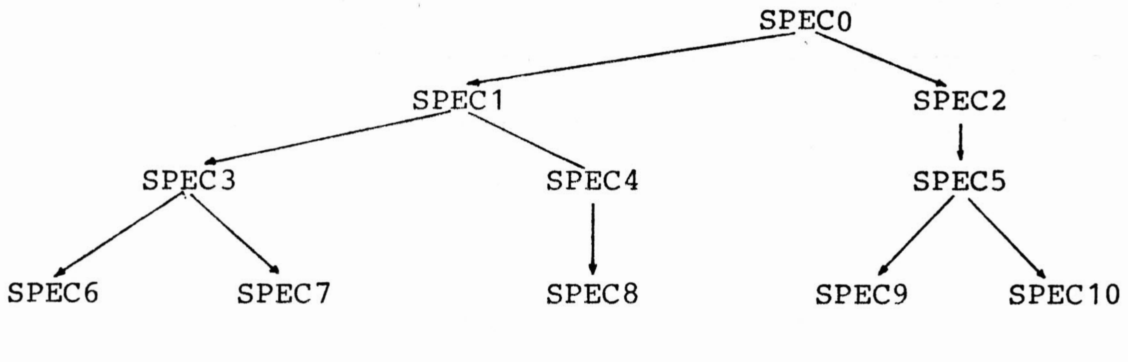
wenn IMPL schwach, stark bzw. persistent ist. Das Papier enthält neben einem großen Beispiel für eine Compound-Implementierung auch noch einen Ansatz für <u>parallele Komposition</u> von Implementierungen, die es ermöglicht, eine "horizontale" Struktur in den schrittweisen Software-Entwicklungsprozess einzubringen. Die Idee dabei ist, daß man nicht nur von einer einzigen grundlegenden Spezifikation SPEC0 wie im Falle der Compound-Implementierungen auszugehen braucht, auf die schließlich alles zurückführbar ist, sondern daß es zwei oder mehrere Grundspezifikationen SPEC0, SPEC0',... usw. für die parallele Implementierung von SPEC1, SPEC1',... usw. gibt. Die sequentielle (•) und die parallele Komposition (+) sind kompatibel. Es gilt: Wenn IMPLi für i=1,...,4 Implementierungen sind, für die

IMPL1•IMPL3 und IMPL2•IMPL4 oder (IMPL1+IMPL2)•(IMPL3+IMPL4) wieder Implementierungen sind, dann gilt

(IMPL1+IMPL2)•(IMPL3+IMPL4)=(IMPL1•IMPL3)+(IMPL2•IMPL4),

wobei beide Seiten Implementierungen sind.

Unter Verwendung beider Kompositionsverfahren ist man in der Lage, <u>algebraische Implementierungsschemata</u> anzugeben, die gegenüber der linearen Struktur der Compound-Implementierungen eine Baumstruktur aufweisen, etwa:



.

Die algebraischen Implementierungsschemata sind den algebraischen Spezifikationsschemata aus [EKW 79] sehr verwandt.
Die Autoren bemerken außerdem, daß ihre Komposition die Probleme, die in [Ehr 78] (sh. 3.1.2.) angesprochen werden, vermei-

det. Es wird nicht nur (wie in [Ehr 78]) festgestellt, daß es eine Implementierung gibt, die aus der Komposition zweier anderer Implementierungen entsteht, sondern es ist syntaktisch und semantisch festgelegt, wie diese "Über-alles"-Implementierung aussieht, während sie in [Ehr 78] konstruiert werden muß.

3.2. Terminale Implementierungen

Wir wollen in diesem Abschnitt das in 3.1.4. vorgestellte Implementierungskonzept, das wir von nun an als I-Implementierung bezeichnen werden, in die terminale Algebrasemantik übertragen. Dies erscheint möglich, da Hornung in [H 79] gezeigt hat, daß es zu fast jedem Begriff der initialen Algebrasemantik eine duale Entsprechung in der terminalen Algebrasemantik gibt, die wir jeweils durch das Präfix i- bzw. t- unterscheiden. Das gilt speziell für die bei den I-Implementierungen benutzten Begriffe wie Kombination, Erweiterung, Anreicherung und Semantik. Es wird darauf verzichtet, die in [EKP 79a] angegebene Motivation der Anforderungen an ein Implementierungskonzept zu wiederholen. Die Anforderungen sind von der Art des Ansatzes – ob initial oder terminal – unabhängig. Wir werden aber sehr wohl untersuchen, inwieweit unser Konzept der T-Implementierungen diesen Anforderungskatalog (sh. 3.1.4.) erfüllt. Obwohl die hier vorgeschlagenen T-Implementierungen in ähnlichen Begriffen wie die I-Implementierungen formuliert werden, ergeben sich bereits zu Beginn unserer Überlegungen einige Unterschiede, die wir hier kurz ansprechen wollen und deren Auswirkungen an späterer Stelle noch ausführlich beschrieben werden.

Da t-Spezifikationen eines Typs gegenüber initial zu interpretierenden Spezifikationen desselben Typs meist – wie wir in 2.3. gesehen haben – zusätzliche Operationen mit Wertebereich <u>dis</u> beinhalten, müssen wir das Konzept der T-Implementierungen etwas anders gestalten als das der I-Implementierungen, wo in der zweiten Implementierungsebene <u>alle</u> Operationen des zu implementierenden Typs angereichert werden, und wo in der ersten Implementierungsebene die Sorten des zu implementierenden Typs

übernommen wurden. Würden wir diese Aufteilung beibehalten, so könnte es vorkommen, daß zwar in der ersten Ebene die Sorten übernommen werden, daß aber keine dis-Kontexte dieser Sorten existieren, da ja die kontexterzeugenden, zu SPEC0 gehörenden Operationen erst in der zweiten Ebene hinzukommen. In dem Fall würden alle Terme dieser Sorten, die aus Sorten implementierenden Operationen gebildet werden, identifiziert werden. In den meisten Fällen ist dann nach Anreicherung der SPEC0-Operationen in der zweiten Ebene die semantische Zusatzforderung, daß die zweite Ebene eine (t-)Anreicherung der ersten Ebene ist, nicht mehr erfüllbar oder die resultierende Semantik stimmt nicht mehr mit der des zu implementierenden Typs überein, denn es müßten auch in der zweiten Ebene alle Terme der obengenannten Sorten identifiziert werden (t-Konsistenz der zweiten Ebene bzgl. der ersten Ebene). Solche (Nicht-)Strukturen, in denen alle Terme identifiziert werden, sind in den wenigsten Fällen von Interesse.
Wir wollen die Aufteilung in zwei Implementierungsebenen beibehalten, jedoch die zusätzlichen Operationen mit Wertebereich dis, die die Kontexte für die Sorten des zu implementierenden Typs erzeugen, zur ersten Ebene bereits hinzunehmen. Da der Begriff "zusätzliche Operationen" nicht allgemein definierbar ist und andererseits Operationen mit Wertebereich dis existieren, die wenig Einfluß auf die t-Semantik haben, also nicht zur ersten Ebene hinzugenommen werden müssen, (Bsp.: IS_NEW:array → dis und ähnliche), wollen wir die Auswahl vom Einzelfall abhängig machen, ebenso wie die Auswahl der diese Operationen beschreibenden Gleichungen, die auch der ersten Ebene angehören sollen.
Auch im terminalen Fall gibt es Gründe, auf einen IDENTIFICATION-Schritt zu verzichten. Bei den Beispielen für t-Spezifi-

kationen, die im Folgenden auftreten, kommen solche "Problem-Gleichungen" wie in set(nat) aus 3.1.4. nicht mehr vor. Die Kommutativitäts- und Idempotenz-Eigenschaften von Operatoren wie INSERT (oder ASSIGN) werden praktisch in den die "zusätzlichen" Operationen beschreibenden Gleichungen "versteckt". Sie sind nicht mehr explizit angegeben wie im initialen Fall, aber implizit noch vorhanden, was man meist durch Induktion zeigen kann. Das Vorhandensein dieser Gleichungen war aber gerade der Grund, den EKP für die Existenz des IDENTIFICATION-Schrittes angaben. Man kann daher auch im terminalen Fall auf den IDENTIFICATION-Schritt verzichten, muß aber darauf achten, daß insbesondere die "zusätzlichen Operationen" korrekt implementiert werden, d.h. daß sie so auf den Datenrepräsentationen definiert werden, daß sich etwa die Kommutativitäts- und Idempotenzeigenschaften von Operatoren wie INSERT oder ASSIGN auch auf die Datenrepräsentation übertragen. Sind alle Operationen korrekt implementiert, so liegt bereits nach RESTRICTION eine Semantik vor, die isomorph ist zu der des implementierten Typs. Es sei noch bemerkt, daß die T-Implementierungen nicht mit den in Definition 37 erwähnten t-Implementationen verwechselt werden sollten. Es wird jedoch in Theorem 1 gezeigt, daß die Semantik einer T-Implementierung auch eine t-Implementation ist.

3.2.1. Das Konzept

GENERALVORAUSSETZUNG für die Abschnitte 3.2.1. bis 3.2.3.

Wir fassen jede t-Spezifikation (S,Σ,E) als t-Kombination $(S_d,\Sigma_d,E_d)+(S',\Sigma',E')$ einer Spezifikation $SPDIS=(S_d,\Sigma_d,E_d)$ für $\underline{\underline{dis}}$ mit $S_d=\{\underline{dis}\}$ auf, d.h. es ist $S=S_d \cup S'$, $\Sigma=\Sigma_d \cup \Sigma'$ und $E=E_d \cup E'$.

SPEC0 und SPEC1 seien t-Spezifikationen, die t-Erweiterung einer gemeinsamen Unterspezifikation $SPEC=(S,\Sigma,E)$ sind, d.h. sie sind syntaktisch t-Kombinationen

$SPEC0 = SPEC + (S0,\Sigma 0,E0)$ und $SPEC1 = SPEC + (S1,\Sigma 1,E1)$,

die die semantische Zusatzforderung erfüllen, daß die SPEC-Anteile ihrer t-Semantik mit der t-Semantik von SPEC übereinstimmen. SPEC, SPEC0 und SPEC1 seien vollständig und konsistent. Es sei $SPEC0' = SPEC + (S0,\Sigma 0,\emptyset)$. $\square$

Die Idee ist, wie im initialen Fall, daß die implementierende und die implementierte (t-)Spezifikation Teile gemeinsam haben können, die nicht erneut zu realisiert werden brauchen. Im Gegensatz zum initialen Fall kann jedoch $SPEC=(\emptyset,\emptyset,\emptyset)$ nicht auftreten, da alle t-Spezifikationen mindestens die Unterspezifikation SPDIS gemeinsam haben.
Es folgen die Definitionen von Syntax und Semantik der schwachen T-Implementierung.

Definition 42 (Syntax der schwachen T-Implementierung)

$IMPL=(\Sigma S0,\Sigma 0D,ES0,E0D,E\Sigma 0)$ ist eine schwache T-Implementierung von SPEC0 mit Hilfe von SPEC1, wenn

$IMPL1 = SPEC1 + (S0,\Sigma S0 \cup \Sigma 0D,ES0 \cup E0D)$ und

$IMPL2 = IMPL1 + (\emptyset,\Sigma 0-\Sigma 0D,E\Sigma 0-E0D)$

vollständige und konsistente t-Kombinationen sind, wobei $E0D \subseteq E\Sigma 0_{dis}$ und $\Sigma 0D \subseteq \Sigma 0DIS=\{\sigma\in\Sigma_{s1\ldots sn,dis} \mid new \wedge (\exists 1\leq j\leq n)(s_j \neq dis)\}$.

$\Sigma S0$ enthält S0-Sorten-implementierende Operationen,

$\Sigma 0D$ enthält S0-Struktur-implementierende Operationen,

ES0 enthält S0-Sorten-implementierende Gleichungen,

E0D enthält S0-Struktur-implementierende Gleichungen und

$E\Sigma 0$ enthält $\Sigma 0$-Operationen-implementierende Gleichungen,

darunter als Teilmenge auch E0D.

Die Schritte von SPEC1 zur ersten (T-)Implementierungsebene IMPL1 heißen entsprechend Sorten&Struktur-Implementierung und von IMPL1 zur zweiten (T-)Implementierungsebene IMPL2 Operationen-Implementierung. □

Definition 43 (Semantik der schwachen T-Implementierung)

Die Semantik der schwachen T-Implementierung $IMPL=(\Sigma S0,\Sigma 0D,ES0,E0D,E\Sigma 0)$ von SPEC0 mit Hilfe von SPEC1 ist die aus der t-Semantik $T_{SPEC1,\sim}$ von SPEC1 durch Anwendung der Konstruktionen EXTENSION und RESTRICTION resultierende SPEC0'-Algebra B', wobei die einzelnen Schritte folgendermaßen definiert sind:

(1) EXTENSION, aufgeteilt in SORT&STRUC- und OP-EXTENSION, überführt $T_{SPEC1,\sim}$ in die t-Semantik der t-Kombination IMPL1 und diese in die von IMPL2.

(2) RESTRICTION, aufgeteilt in FORGETTING und REACHABILITY, überführt $T_{IMPL2,\sim}$ in seinen SPEC0'-Anteil $T_{IMPL2,\sim}|_{\Sigma\cup\Sigma}$ und bildet darin das Bild $B' = h'(T_{\Sigma\cup\Sigma 0})$ der "Interpretation" $h':T_{\Sigma\cup\Sigma 0} \to T_{IMPL2,\sim}|_{\Sigma\cup\Sigma 0}$ der $\Sigma\cup\Sigma 0$-Terme, wobei B' eine SPEC0'-Unteralgebra von $T_{IMPL2,\sim}|_{\Sigma\cup\Sigma 0}$ wird.

$$\begin{tikzcd}
T_{SPEC1,\sim} \arrow[rr, "\text{EXTENSION}"] \arrow[dr, "\text{SORT\&STRUC-EXTENSION}"'] & & T_{IMPL2,\sim} \arrow[rr, "\text{RESTRICTION}"] \arrow[dr, "\text{FORGETTING}"] & & B' \\
& T_{IMPL1,\sim} \arrow[ur, "\text{OP-EXTENSION}"] & & T_{IMPL2,\sim}|_{\Sigma\cup\Sigma 0} \arrow[ur, "\text{REACHBILITY}"] &
\end{tikzcd}$$

□

Unter den schwachen T-Implementierungen werden durch semantische Zusatzbedingungen einige herausgesucht, die die Punkte 2. bis 4. des Anforderungskataloges aus 3.1.4. erfüllen.

Definition 44 (T-Implementierung)

Eine T-Implementierung von SPEC0 mit Hilfe von SPEC1 ist eine schwache T-Implementierung IMPL=(ΣS0,Σ0D,ES0,E0D,EΣ0) dieser t-Spezifikationen, bei der zusätzlich gilt:

(i) ihre Semantik B' ist $\Sigma \cup \Sigma 0$-isomorph zur t-Semantik von SPEC0,

(ii) die erste Implementierungsebene IMPL1 ist eine t-Erweiterung von SPEC1 und

(iii) die zweite Implementierungsebene IMPL2 ist eine t-Anreicherung von IMPL1. □

Diese Bedingungen lassen sich auch als Korrektheitskriterien auffassen. Schwache T-Implementierungen, die nur die Bedingung (i) erfüllen, sollen R-korrekt heißen. Dies ist die terminale Entsprechung für die RI-Korrektheit aus 3.1.4.. Im nächsten Theorem wird die R-Korrektheit charakterisiert.

Definition 45 (R-korrekt)

Eine schwache T-Implementierung IMPL von SPEC0 durch SPEC1 heißt R-korrekt, wenn ihre Semantik B' $\Sigma \cup \Sigma 0$-isomorph zur t-Semantik von SPEC0 ist. □

Theorem 1

Für eine schwache T-Implementierung IMPL von SPEC0 durch SPEC1 sind die folgenden Aussagen äquivalent:

(i) IMPL ist R-korrekt

(ii) B' ist terminale SPEC0-Algebra

(iii) Der terminale $\Sigma \cup \Sigma 0$-Homomorphismus $f' : B' \rightarrow T_{SPEC0,\sim}$

existiert und ist ein $\Sigma\cup\Sigma0$-Monomorphismus.

(iv) Alle in SPEC0 kongruenten $\Sigma\cup\Sigma0$-Terme p und q sind auch in IMPL2 kongruent:

$(\forall s\in S\cup S0)(\forall p,q\in T_{\Sigma\cup\Sigma0,s})(p \sim_{E\cup E0,s} q \Rightarrow p \sim_{EIMPL2,s} q)$

mit $EIMPL2=E\cup E1\cup ES0\cup E\Sigma0$.

Beweis:

Durch Ringschluß.

(iv) => (iii)

Wir zeigen zunächst: $B'\in|t\text{-}Imp_{\Sigma\cup\Sigma0,E\cup E0}|$, d.h.

(a) B' ist $\Sigma\cup\Sigma0$-erzeugt

(b) $B'_{dis} = \{tt_{B'},ff_{B'}\} \wedge tt_{B'}\neq ff_{B'}$

(c) $(\forall s\in S\cup S0)(\forall p,q\in T_{\Sigma\cup\Sigma0,s})(p \sim_{E\cup E0,s} q \Rightarrow p \sim_{B',s} q)$.

ad (a)

B' ist $\Sigma\cup\Sigma0$-erzeugt, wenn B' $\Sigma\cup\Sigma0$-Algebra ist und wenn der initiale $\Sigma\cup\Sigma0$-Homomorphismus $h:T_{\Sigma\cup\Sigma0} \to B'$ ein $\Sigma\cup\Sigma0$-Epimorphismus ist.

B' ist nach Konstruktion bereits eine $\Sigma\cup\Sigma0$-Algebra.

Die "Interpretation" $h':T_{\Sigma\cup\Sigma0} \to T_{IMPL2,\sim}|_{\Sigma\cup\Sigma0}$ ist nach Definition auch ein $\Sigma\cup\Sigma0$-Homomorphismus. Die Einschränkung $e:T_{\Sigma\cup\Sigma0} \to B'$ von h' auf das Bild $B'=h'(T_{\Sigma\cup\Sigma0})$ ist ebenfalls ein $\Sigma\cup\Sigma0$-Homomorphismus und sogar ein $\Sigma\cup\Sigma0$-Epimorphismus. Da $T_{\Sigma\cup\Sigma0}$ eine initiale $\Sigma\cup\Sigma0$-Algebra ist, ist e eindeutig, d.h. es gilt h=e.

ad (b)

Nach Generalvoraussetzung sind alle betrachteten Spezifikationen vollständig und konsistent, d.h. es gilt auch

$tt \not\equiv_{E\cup E0,dis} ff$ und $(\forall t\in T_{\Sigma\cup\Sigma0,dis})(\exists tv\in\{tt,ff\})(t\equiv_{E\cup E0,dis} tv)$.

Der dis-Träger von $T_{SPEC0,\sim}$ besteht also aus genau zwei Klassen. Demnach besteht dann auch der dis-Träger von $T_{IMPL2,\sim}$ aus genau zwei Klassen [tt] und [ff]. Da tt und ff $\Sigma\cup\Sigma0$-Operationen sind, besteht auch der dis-Träger der

SPEC0'-Unteralgebra B' von $T_{IMPL2,\sim}$ aus genau zwei Klassen $tt_{B'}=[tt]$ und $ff_{B'}=[ff]$, q.e.d..

ad (c)

- Die Relation $\sim_{B'}$ ist wie folgt definiert:

  $(\forall s\in S\cup S0)(\forall p,q\in T_{\Sigma\cup\Sigma 0,s})(p \sim_{B',s} q \quad :\Leftrightarrow$
  $(\forall ct\in C_{\Sigma\cup\Sigma 0}(s,dis))(ct[p/x_s] \equiv_{B',dis} ct[q/x_s])))$.

- Die Relation $\equiv_{B'}$ ist wie folgt definiert:

  $(\forall s\in S\cup S0)(\forall p,q\in T_{\Sigma\cup\Sigma 0,s})(p \equiv_{B',s} q \quad :\Leftrightarrow \quad h_s(p)=h_s(q))$,

  wobei h der eindeutige $\Sigma\cup\Sigma 0$-Homomorphismus $h:T_{\Sigma\cup\Sigma 0} \to B'$ ist.

- Also gilt:

  $(\forall s\in S\cup S0)(\forall p,q\in T_{\Sigma\cup\Sigma 0,s})(p \sim_{B',s} q \quad \Leftrightarrow$
  $(\forall ct\in C_{\Sigma\cup\Sigma 0}(s,dis))(h_{dis}(ct[p/x_s])=h_{dis}(ct[q/x_s])))$.

Seien $s\in S\cup S0$ und $p,q\in T_{\Sigma\cup\Sigma 0,s}$ mit $p \sim_{E\cup E0,s} q$. Dann gilt

$p \sim_{E\cup E0,s} q \quad \Leftrightarrow$
$(\forall ct\in C_{\Sigma\cup\Sigma 0}(s,dis))(ct[p/x_s] \equiv_{E\cup E0,dis} ct[q/x_s]) \quad \Leftrightarrow_{(1)}$
$(\forall ct\in C_{\Sigma\cup\Sigma 0}(s,dis))(ct[p/x_s] \sim_{E\cup E0,dis} ct[q/x_s]) \quad \Rightarrow_{(2)}$
$(\forall ct\in C_{\Sigma\cup\Sigma 0}(s,dis))(ct[p/x_s] \sim_{EIMPL2,dis} ct[q/x_s]) \quad \Leftrightarrow_{(3)}$
$(\forall ct\in C_{\Sigma\cup\Sigma 0}(s,dis))(h_{dis}(ct[p/x_s])=h_{dis}(ct[q/x_s])) \quad \Leftrightarrow$
$p \sim_{B',s} q$.

Dabei wurde benutzt:

(1) die Tatsache, daß $\equiv_{E,dis}=\sim_{E,dis}$ für jede t-Spezifikation (S,Σ,E) ist (sh. note nach 2.2.4. in [HR 79]),

(2) Punkt (iv) des Theorems 1 und

(3) B' ist $\Sigma\cup\Sigma 0$-erzeugt, d.h. h ist $\Sigma\cup\Sigma 0$-Epimorphismus; B' ist Unteralgebra von $T_{IMPL2,\sim}$, d.h. es gilt $t \sim_{EIMPL2,dis} t' \;\Leftrightarrow\; [t]=[t'] \;\Leftrightarrow\; h_{dis}(t)=h_{dis}(t')$.

Damit ist (c) gezeigt.

Wir wissen jetzt, daß die Semantik B' von IMPL eine t-Implementation von $T_{SPEC0,\sim}$ ist. Daher existiert der terminale $\Sigma\cup\Sigma 0$-Homomorphismus $f':B' \to T_{SPEC0,\sim}$ und ist eindeutig. Es bestehen die im nebenstehenden Bild dargestellten Beziehungen; nat0 ist der natürliche $\Sigma\cup\Sigma 0$-Epimorphismus, der jedem Term die ihn enthaltende Klasse zuordnet. Da nat0

$$\begin{tikzcd} T_{\Sigma\cup\Sigma 0} \arrow[rr, "h"] \arrow[dr, "nat0"'] & & B' \arrow[dl, "f'"] \\ & T_{SPEC0,\sim} & \end{tikzcd}$$

eindeutig ist und $f'\bullet h$ ebenfalls ein $\Sigma\cup\Sigma 0$-Homomorphismus von $T_{\Sigma\cup\Sigma 0}$ nach $T_{SPEC0,\sim}$ ist, gilt $nat0=f'\bullet h$.

Seien nun $a,b\in B'_s$ mit $s\in S\cup S0$. Da h ein $\Sigma\cup\Sigma 0$-Epimorphismus ist, gilt $(\exists p,q\in T_{\Sigma\cup\Sigma 0,s})(h_s(p)=a \wedge h_s(q)=b)$. Dann gilt

$$\begin{aligned}
f'_s(a)=f'_s(b) \quad &\Leftrightarrow \quad f'_s(h_s(p))=f'_s(h_s(q)) \\
&\Leftrightarrow \quad nat0_s(p)=nat0_s(q) \\
{}_{(1)}&\Leftrightarrow \quad p \sim_{E\cup E0,s} q \\
{}_{(2)}&\Rightarrow \quad p \sim_{EIMPL2,s} q \\
&\Leftrightarrow \quad h_s(p)=h_s(q) \\
&\Leftrightarrow \quad a=b \qquad\qquad .
\end{aligned}$$

Dabei ist $_{(1)}$ Definition von nat0 und $_{(2)}$ ist Punkt (iv) des Theorems 1. Wir haben also gezeigt, daß für alle $s\in S\cup S0$ f'_s injektiv ist. Also ist f' ein $\Sigma\cup\Sigma 0$-Monomorphismus.

(iii) => (i)

Hier bleibt zu zeigen, daß $f':B' \to T_{SPEC0,\sim}$ ein $\Sigma\cup\Sigma 0$-Epimorphismus ist.

Wegen $nat0=f'\bullet h$ ist f' mit nat0 und h ebenfalls ein $\Sigma\cup\Sigma 0$ -Epimorphismus und mit (iii) ist f' nach Lemma 2 auch ein $\Sigma\cup\Sigma 0$-Isomorphismus.

(i) => (ii)

Dieser Schluß gilt nach Korollar 3.

(ii) => (iv)

Die terminale SPEC0-Algebra B' ist $\Sigma\cup\Sigma 0$-isomorph zu der ebenfalls terminalen SPEC0-Algebra $T_{SPEC0,\sim}$, d.h. es gibt einen $\Sigma\cup\Sigma 0$-Isomorphismus $f:B' \to T_{SPEC0,\sim}$. Aus Eindeutigkeitsgründen gilt wieder $nat0=f\bullet h$. Damit gilt

$$(\forall s\in S\cup S0)(\forall p,q\in T_{\Sigma\cup\Sigma 0,s})$$

$$\begin{aligned}(p \sim_{E\cup E0,s} q \quad &\Leftrightarrow \quad nat0_s(p)=nat0_s(q)\\ &\Leftrightarrow \quad f_s(h_s(p))=f_s(h_s(q))\\ {}_{(1)}&\Rightarrow \quad h_s(p)=h_s(q)\\ &\Leftrightarrow \quad p \sim_{EIMPL2,s} q),\end{aligned}$$

wobei bei $_{(1)}$ die Injektivität von f_s ausgenutzt wurde. □

Im Vergleich zu den I-Implementierungen ist die Existenz einer Abstraktionsfunktion, die repräsentierende Daten auf repräsentierte abbildet, viel leichter nachzuweisen, denn sie wird bei der Konstruktion mitgeliefert.

Korollar REP

Sei $IMPL=(\Sigma S0,\Sigma 0D,ES0,E0D,E\Sigma 0)$ eine T-Implementierung von SPEC0 mit Hilfe von SPEC1. Sei B' die Semantik von IMPL, d.h. eine SPEC0'-Unteralgebra von $T_{IMPL2,\sim}$. Dann gibt es genau einen $\Sigma\cup\Sigma 0$-Epimorphismus REP(RESENTATION):B' $\rightarrow T_{SPEC0,\sim}$.

Beweis:

Die Gültigkeit der Behauptung folgt direkt aus Punkt (i) der Definition 44 und aus Lemma 2. Es ist REP=f' mit f' wie im Punkt (iii) von Theorem 1. □

Bemerkung:

Da f' und damit REP sogar ein $\Sigma\cup\Sigma 0$-Isomorphismus ist, kommen Mehrfachrepräsentationen von SPEC0-Daten zwar möglicherweise noch in $T_{IMPL2,\sim}$, aber nicht mehr in B' vor.

Jetzt sind wir in der Lage, das Konzept der T-Implementierungen an dem in 3.1.4. aufgestellten Anforderungskatalog zu messen.

- Punkt 1. ist klar nach Definition der Syntax der schwachen T-Implementierung.

- Punkt 2. ist klar nach Definition der Semantik der schwachen T-Implementierung und nach Definition der T-Implementierung.
- Zu Punkt 3.:

  Die Punkte (ii) und (iii) der Definition der T-Implementierung stellen sicher, daß die SPEC1-Anteile der ersten und der zweiten Implementierungsebene noch mit der t-Semantik von SPEC1 übereinstimmen, d.h. daß sie weder zerstört noch verändert werden.

  Die Existenz des $\Sigma\cup\Sigma 0$-Epimorphismus REP:B' $\rightarrow T_{SPEC0,\sim}$ stellt sicher, daß verschiedene ADT0-Daten auch verschieden repräsentiert sind (Rechtseindeutigkeit von REP). Für Mehrfachrepräsentationen gilt das nach Korollar REP Gesagte. Auch benutzen die ΣS0-Operationen, die die S0-Sorten "aufbauen", tatsächlich nur ADT1-Daten (SPEC1-Terme) als Argumente, denn wir wissen, daß IMPL1 t-vollständig auf SPEC1 ist (Satz 6), d.h. wir können jeden IMPL1-Term mit Sorte aus S$\cup$S1, der als Argument einer ΣS0-Operation auftaucht, durch einen äquivalenten SPEC1-Term ersetzen.
- Zu Punkt 4.:

  Jede Anwendung von SPEC0-Operationen (jede ADT0-Berechnung) kann bis auf Datenrepräsentationen auf Anwendungen von SPEC1 -Operationen zurückgeführt werden.

  Dabei sichert die Homomorphie-Eigenschaft von REP, daß zunächst jede Anwendung einer SPEC0-Operation bis auf Datenrepräsentation auf Anwendungen von IMPL2-Operationen (d.s. SPEC0- und IMPL1-Operationen) zurückgeführt werden kann. Wir können daher schrittweise in einer ADT0-Berechnung die in IMPL2 ausgeführt wird, alle SPEC0-Operationsaufrufe ersetzen, sodaß wir eine Berechnung erhalten, die nur noch IMPL1-Operationsaufrufe enthält. Ein Teil dieser Aufrufe sind Aufrufe von ΣS0-Operationen, d.h. SPEC0-Datenrepräsentationen, die wir nicht weiter ersetzen können. Alle anderen IMPL1-Operationsaufrufe lassen sich jedoch durch SPEC1-Berechnungen ersetzen (vergl. oben zu Punkt 3.; IMPL1 ist t-vollständig auf SPEC1). Damit ist die Anforderung 4. erfüllt, denn wir haben gezeigt, daß sich jede ADT0-Berechnung bis auf Datenrepräsentation auch in ADT1 ausführen läßt.

3.2.2. Möglichkeiten der Sorten&Struktur-Implementierung

Lemma 15

Gilt bei einer schwachen T-Implementierung

$IMPL=(\Sigma S0,\Sigma 0D,ES0,E0D,E\Sigma 0)$ von SPEC0 mit Hilfe von SPEC1

(i) $E0D=\Sigma 0D=\emptyset$

(ii) $s\in S0$ für alle $w\in(S\cup S1\cup S0)^*$ und für alle $\sigma\in\Sigma S0_{w,s}$

(iii) $s\in S0$ für alle $(L,R)\in ES0_s$,

so ist die erste Implementierungsebene

$IMPL1=SPEC1+(S0,\Sigma S0\cup\Sigma 0D,ES0\cup E0D)$ eine t-Erweiterung von SPEC1.

Beweis:

Seien t_1,t_2 zwei in SPEC1 kongruente Terme einer Sorte s aus $S\cup S1$: $t_1 \sim_{E\cup E1,s} t_2$. Nach (i) und (ii) kommen weder in t_1 noch in t_2 neue Operationen vor. Wegen (i) bis (iii) sind die Kontextkategorien $C_{\Sigma\cup\Sigma 1}(s,dis)$ und $C_{\Sigma IMPL1}(s,dis)$ von SPEC1 bzw. IMPL1 für alle $s\in S\cup S1$ gleich. Also gilt auch $t_1 \sim_{EIMPL1,s} t_2$, d.h. IMPL1 ist t-konsistent auf SPEC1. Die Träger von $T_{IMPL1,\sim}$ und $T_{SPEC1,\sim}$ sind für alle $s\in S\cup S1$ gleich, also ist IMPL1 auch t-vollständig auf SPEC1. Nach Satz 6 ist IMPL1 daher eine t-Erweiterung von SPEC1. □

Die Entsprechung von Lemma 15 in [EKP 79a] (Lemma 5.1., sh.

3.1.4.) stellt für viele praktische Fälle von I-Implementierungen ein einfaches syntaktisches Schema der Sorten-Implementierung dar. Unser Entschluß, die "zusätzlichen Operationen" zur ersten Ebene hinzuzunehmen, eliminiert diesen Vorteil. Die interessanten T-Implementierungen mit $E0D \neq \emptyset \neq \Sigma 0D$ werden von Lemma 15 nicht erfaßt. Wir können jedoch eine schwächere Aussage formulieren.

Lemma 16

Gilt bei einer schwachen T-Implementierung

$IMPL=(\Sigma S0,\Sigma 0D,ES0,E0D,E\Sigma 0)$ von SPEC0 mit Hilfe von SPEC1

(i) $s \in S0$ für alle $w \in (S \cup S1 \cup S0)^*$ und für alle $\sigma \in \Sigma S0_{w,s}$,

(ii) $(\forall n \in \omega)(\forall s,s_1,\ldots,s_n \in S \cup S1)(\forall \sigma \in \Sigma_{s_1\ldots s_n,s} \cup \Sigma 1_{s_1\ldots s_n,s})$

$(s \neq \underline{dis} \Rightarrow (\forall i \in \{1,\ldots,n\})(s_i \neq \underline{dis}))$,

dann ist $IMPL1=SPEC1+(S0,\Sigma S0 \cup \Sigma 0D,ES0 \cup E0D)$ sowohl t-vollständig als auch i-vollständig auf SPEC1.

Beweis:

a) $s \in S \cup S1-\{\underline{dis}\}$.

(i) und (ii) stellen sicher, daß $T_{\Sigma IMPL1,s}=T_{\Sigma \cup \Sigma 1,s}$. Es gilt $(\forall p \in T_{\Sigma IMPL1,s})(\exists q \in T_{\Sigma \cup \Sigma 1,s})(p \sim_{EIMPL1,s} q \wedge p \equiv_{EIMPL1,s} q)$ mit p $\underline{is}$ q aufgrund der Reflexivität der beiden Kongruenzen.

b) $s=\underline{dis}$.

Die Gültigkeit der Behauptung folgt, da IMPL1 vollständig ist, aus Korollar 4 und 6 bzw. nur aus Korollar 4. $\square$

In Lemma 16 wird also $E0D \neq \emptyset \neq \Sigma 0D$ zugelassen; jedoch dürfen die SPEC1-Operationen nicht die Sorte $\underline{dis}$ im Rang haben ((ii)), da sonst über die "zusätzlichen Operationen" aus $\Sigma 0D$ neue Terme der SPEC1-Sorten S1 entstehen könnten.

Für die unter Lemma 16 erfaßten Fälle ist dann nur noch die t-Konsistenz von IMPL1 auf SPEC1 zu zeigen, um IMPL1 zu einer t-Erweiterung von SPEC1 zu machen.

Bemerkung:

Da bei dem Beweis von Lemma 16 die Konsistenz von IMPL1 und SPEC1 nicht benötigt wurde, gilt das Lemma auch für nicht -konsistente t-Spezifikationen IMPL1 und SPEC1, die in der in dem Lemma geforderten Beziehung zueinander stehen.

3.2.3. Notation für (schwache) T-Implementierungen

Analog zu der in 2.2. vorgestellten Notation für Spezifikationen und Kombinationen wird in diesem Abschnitt eine Notation für (schwache) T-Implementierungen angegeben.

Sei IMPL=(ΣS0,Σ0D,ES0,E0D,EΣ0) eine (schwache) T-Implementierung von SPEC0 mit Hilfe von SPEC1 und SPEC, SPEC0 und SPEC1 seien wie in der Generalvoraussetzung gefordert.

IMPL wird wie folgt aufgeschrieben:

<Angabe von SPEC, SPEC0 und SPEC1 gemäß 2.2.>

SPEC1 impl SPEC0 by

sorts impl opns:	abr:
$o_1 : w_1 \rightarrow s_{j_1}$	(a_1)
$\vdots$	$\vdots$
$o_n : w_n \rightarrow s_{j_n}$	(a_n)
struc impl opns:	abr:
$o'_1 : w'_1 \rightarrow s'_{j_1}$	(a'_1)
$\vdots$	$\vdots$
$o'_m : w'_m \rightarrow s'_{j_m}$	(a'_m)

sorts impl eqns:

ES0.1 $L_1=R_1$

$\vdots \quad \vdots$

ES0.k $L_k=R_k$

struc impl eqns:

E0D.1 $L_1'=R_1'$

$\vdots \quad \vdots$

E0D.l $L_l'=R_l'$

opns impl eqns:

EΣ0.l+1 $L_1''=R_1''$

$\vdots \quad \vdots$

EΣ0.l+i $L_i''=R_i''$

Dabei werden hinter der Marke

sorts impl eqns: (sorts implementing operations)

die Elemente von ΣS0 und hinter

struc impl opns: (structure implementing operations)

die Elemente von Σ0D so aufgelistet wie die Operationssymbole bei Spezifikationen. Für die Abkürzungen gilt das Entsprechende.

Es werden hinter der Marke

sorts impl eqns: (sorts implementing equations)

die Elemente von ES0, hinter

struc impl eqns: (structure implementing equations)

die Elemente von E0D und hinter

opns impl eqns: (operations implementing equations)

die Elemente von EΣ0-E0D so aufgelistet wie die Axiome bei der Notation für Spezifikationen, wobei wegen E0D$\subseteq$EΣ0 die Nummern unter der letzten Marke nicht wieder mit 1 beginnen, sondern durchgezählt werden.

Ist z.B. ΣS0=$\emptyset$, so wird dies durch no sorts impl opns ausgedrückt. Bei Σ0D,ES0,E0D,EΣ0-E0D=$\emptyset$ gilt das Entsprechende.

3.2.4. Ein Beispiel

Wir wählen als Beispiel die T-Implementierung von Kellern von natürlichen Zahlen (stack(nat)) durch Felder mit natürlichen Zahlen als Indices und Einträgen (array(nat,nat)).

SPEC:

sorts: dis,nat

opns: TT,FF: → dis abr: (tt,ff)

NOT:dis → dis (¬)

AND,_EQV_:dis dis → dis (_∧_,_EQV_)

NULL: → nat (∅)

SUCC:nat → nat (S)

EQN:nat nat → dis

eqns:

E.1 $\neg tt = ff$

E.2 $\neg ff = tt$

E.3 $tt \wedge x_d^1 = x_d^1$

E.4 $ff \wedge x_d^1 = ff$

E.5 $tt\ EQV\ x_d^1 = x_d^1$

E.6 $ff\ EQV\ x_d^1 = \neg x_d^1$

E.7 $EQN(\emptyset,\emptyset) = tt$

E.8 $EQN(\emptyset,S(x_n^1)) = ff$

E.9 $EQN(S(x_n^1),\emptyset) = ff$

E.10 $EQN(S(x_n^1),S(x_n^2)) = EQN(x_n^1,x_n^2)$

SPEC0: SPEC +

sorts: stack

opns: NEWST: → stack abr: (NS)

PUSH:stack nat → stack (PU)

POP:stack → stack (PP)

TOP: stack $\rightarrow$ nat (TO)

EQS: stack stack $\rightarrow$ dis

eqns:

E0.1 $\mathrm{EQS(NEWST,NEWST)} = \mathrm{tt}$

E0.2 $\mathrm{EQS(NEWST,PUSH}(x_s^1,x_n^1)) = \mathrm{ff}$

E0.3 $\mathrm{EQS(PUSH}(x_s^1,x_n^1),\mathrm{NEWST)} = \mathrm{ff}$

E0.4 if $\mathrm{EQN}(x_n^1,x_n^2) \doteq \mathrm{tt}$ then

$$\mathrm{EQS(PUSH}(x_s^1,x_n^1),\mathrm{PUSH}(x_s^2,x_n^2)) = \mathrm{EQS}(x_s^1,x_s^2)$$

E0.5 if $\mathrm{EQN}(x_n^1,x_n^2) \doteq \mathrm{ff}$ then

$$\mathrm{EQS(PUSH}(x_s^1,x_n^1),\mathrm{PUSH}(x_s^2,x_n^2)) = \mathrm{ff}$$

E0.6 $\mathrm{TOP(NEWST)} = \emptyset$

E0.7 $\mathrm{TOP(PUSH}(x_s^1,x_n^1)) = x_n^1$

E0.8 $\mathrm{POP(NEWST)} = \mathrm{NEWST}$

E0.9 $\mathrm{POP(PUSH}(x_s^1,x_n^1)) = x_s^1$

SPEC1: SPEC +

sorts: array

opns:

NEWA: $\rightarrow$ array abr: (NA)

ASSIGN: array nat nat $\rightarrow$ array (AS)

REMOVE: array nat $\rightarrow$ array (RE)

ACCESS: array nat $\rightarrow$ nat (AC)

INDEX: array nat $\rightarrow$ dis (IN)

EQA: array array $\rightarrow$ dis

eqns:

E1.1 $\mathrm{EQA(NA,NA)} = \mathrm{tt}$

E1.2 $\mathrm{EQA(NA,AS}(x_a^1,x_n^1,x_n^2)) = \mathrm{ff}$

E1.3 $\mathrm{EQA(AS}(x_a^1,x_n^1,x_n^2),\mathrm{NA)} = \mathrm{ff}$

E1.4 if $\mathrm{IN(AS}(x_a^1,x_n^1,x_n^2),x_n^3) \wedge \mathrm{IN(AS}(x_a^2,x_n^3,x_n^4),x_n^1) \doteq \mathrm{tt}\,\&$

$\mathrm{EQN(AC(AS}(x_a^1,x_n^1,x_n^2),x_n^3),x_n^4) \wedge \mathrm{EQN(AC(AS}(x_a^2,x_n^3,x_n^4),x_n^1),x_n^2) \doteq \mathrm{tt}$

then

$$\mathrm{EQA(AS}(x_a^1,x_n^1,x_n^2),\mathrm{AS}(x_a^2,x_n^3,x_n^4)) = \mathrm{EQA(RE(RE}(x_a^1,x_n^1),x_n^3),\mathrm{RE(RE}(x_a^2,x_n^3),x_n^1))$$

E1.5 if $IN(AS(x_a^1,x_n^1,x_n^2),x_n^3) \wedge IN(AS(x_a^2,x_n^3,x_n^4),x_n^1) \doteq tt$ &

$EQN(AC(AS(x_a^1,x_n^1,x_n^2),x_n^3),x_n^4) \wedge EQN(AC(AS(x_a^2,x_n^3,x_n^4),x_n^1),x_n^2) \doteq ff$

then

$EQA(AS(x_a^1,x_n^1,x_n^2),AS(x_a^2,x_n^3,x_n^4)) = ff$

E1.6 if $IN(AS(x_a^1,x_n^1,x_n^2),x_n^3) \wedge IN(AS(x_a^2,x_n^3,x_n^4),x_n^1) \doteq ff$ then

$EQA(AS(x_a^1,x_n^1,x_n^2),AS(x_a^2,x_n^3,x_n^4)) = ff$

E1.7 $IN(NA,x_n^1) = ff$

E1.8 if $EQN(x_n^1,x_n^3) \doteq tt$ then

$IN(AS(x_a^1,x_n^1,x_n^2),x_n^3) = tt$

E1.9 if $EQN(x_n^1,x_n^3) \doteq ff$ then

$IN(AS(x_a^1,x_n^1,x_n^2),x_n^3) = IN(x_a^1,x_n^3)$

E1.10 $AC(NA,x_n^1) = \emptyset$

E1.11 if $EQN(x_n^1,x_n^3) \doteq tt$ then

$AC(AS(x_a^1,x_n^1,x_n^2),x_n^3) = x_n^2$

E1.12 if $EQN(x_n^1,x_n^3) \doteq ff$ then

$AC(AS(x_a^1,x_n^1,x_n^2),x_n^3) = AC(x_a^1,x_n^3)$

E1.13 $RE(NA,x_n^1) = NA$

E1.14 if $EQN(x_n^1,x_n^3) \doteq tt$ then

$RE(AS(x_a^1,x_n^1,x_n^2),x_n^3) = RE(x_a^1,x_n^3)$

E1.15 if $EQN(x_n^1,x_n^3) \doteq ff$ then

$RE(AS(x_a^1,x_n^1,x_n^2),x_n^3) = AS(RE(x_a^1,x_n^3),x_n^1,x_n^2)$

SPEC1 impl SPEC0 by

sorts impl opns: abr:

STAK:<u>array</u> <u>nat</u> → <u>stack</u> (ST)

struc impl opns:

EQS:<u>stack</u> <u>stack</u> → <u>dis</u>

no sorts impl eqns

struc impl eqns

E0D.1 $EQS(ST(x_a^1,\emptyset),ST(x_a^2,\emptyset)) = tt$

EOD.2 $\quad EQS(ST(x_a^1,\emptyset),ST(x_a^2,S(x_n^1))) = ff$

EOD.3 $\quad EQS(ST(x_a^1,S(x_n^1)),ST(x_a^2,\emptyset)) = ff$

EOD.4 $\quad$ if $IN(x_a^1,S(x_n^1))\ EQV\ IN(x_a^2,S(x_n^2)) \doteq tt\ \&$

$EQN(AC(x_a^1,S(x_n^1)),AC(x_a^2,S(x_n^2))) \doteq tt$ then

$EQS(ST(x_a^1,S(x_n^1)),ST(x_a^2,S(x_n^2))) = EQS(ST(x_a^1,x_n^1),ST(x_a^2,x_n^2))$

EOD.5 $\quad$ if $IN(x_a^1,S(x_n^1))\ EQV\ IN(x_a^2,S(x_n^2)) \doteq tt\ \&$

$EQN(AC(x_a^1,S(x_n^1)),AC(x_a^2,S(x_n^2))) \doteq ff$ then

$EQS(ST(x_a^1,S(x_n^1)),ST(x_a^2,S(x_n^2))) = ff$

EOD.6 $\quad$ if $IN(x_a^1,S(x_n^1))\ EQV\ IN(x_a^2,S(x_n^2)) \doteq ff$ then

$EQS(ST(x_a^1,S(x_n^1)),ST(x_a^2,S(x_n^2))) = ff$

opns impl eqns:

E∑0.7 $\quad NEWST = STAK(NEWA,\emptyset)$

E∑0.8 $\quad PUSH(STAK(x_a^1,x_n^1),x_n^2) = STAK(ASSIGN(x_a^1,S(x_n^1),x_n^2),S(x_n^1))$

E∑0.9 $\quad POP(STAK(x_a^1,\emptyset)) = STAK(x_a^1,\emptyset)$

E∑0.10 $\quad POP(STAK(x_a^1,S(x_n^1))) = STAK(x_a^1,x_n^1)$

E∑0.11 $\quad TOP(STAK(x_a^1,\emptyset)) = \emptyset$

E∑0.12 $\quad TOP(STAK(x_a^1,S(x_n^1))) = ACCESS(x_a^1,S(x_n^1))$

Es ist also IMPL=(∑S0,∑0D,ES0,E0D,E∑0) mit

∑S0={STAK:<u>array</u> <u>nat</u> → <u>stack</u>}

∑0D={EQS:<u>stack</u> <u>stack</u> → <u>dis</u>}

ES0=∅

E0D={E0D.1,...,E0D.6}

E∑0= E0D∪{E∑0.7,...,E∑0.12}.

Diese Bezeichnungen gelten für das gesamte Kapitel 3.2.4..
Die Sorten <u>dis</u>, <u>nat</u>, <u>stack</u> und <u>array</u> werden jeweils mit ihren Anfangsbuchstaben abgekürzt. Die Spezifikationsbezeichnungen SPEC, SPEC0, SPEC1, IMPL1 und IMPL2 werden SP, SP0, SP1, I1 und I2 abgekürzt. Außerdem gelten folgende Abkürzungen und Identitäten:

$(\forall j\epsilon\omega_0)$ ($\underbrace{S(\ldots S}_{j\text{-mal}}(\emptyset)\ldots) =: S^j(\emptyset)$), speziell $S^0(\emptyset)$ is $\emptyset$.

$(\forall m\epsilon\omega_0)$ (m=0 =>

(ASSIGN(...ASSIGN(NEWA,p_1,q_1),...,p_m,q_m) is NEWA $\wedge$

PUSH(...PUSH(NEWST,r_1),...,r_m) is NEWST)),

wobei die $p_1,\ldots,p_m,q_1,\ldots,q_m$ nat-Terme aus SP1, I1 oder I2 und die $r_1,\ldots,r_m$ nat-Terme aus SP0 oder I2 sind.

3.2.4.1. Erläuterungen zum Beispiel

SPEC ist eine einfache Spezifikation für dis und nat. nat-Terme bestehen entweder nur aus der NULL oder aus endlich vielen Anwendungen von SUCC auf NULL. EQN sorgt dafür, daß die Menge der dis-Kontexte für nat-Terme nicht leer ist. Unter $\sim_{E,nat}$ wird jeder nat-Term nur mit sich selbst identifiziert.

SPEC0 ist eine auf SPEC aufbauende Spezifikation für stack(nat). NEWST steht für den leeren Keller und PUSH fügt zu einem bestehenden Keller ein neues Top-Element hinzu. Darüberhinaus besteht die Möglichkeit, mit TOP das oberste Keller-Element zu lesen bzw. es mit POP zu entfernen. Kanonische stack-Terme sind gerade solche, die weder TOP noch POP enthalten. EQS ist der Gleichheitsoperator und erzeugt auch Kontexte für stack-Terme.

SPEC1 ist eine auf SPEC aufbauende Spezifikation für arrays mit nat-Termen als Indices und als Einträgen. NEWA steht für den leeren array und ASSIGN(a,n_1,n_2) fügt in den bestehenden array a an der Stelle n_1 den Eintrag n_2 hinzu. ACCESS(a,n_1) liefert den an der Stelle n_1 in a befindlichen Eintrag bzw. NULL, falls an der Stelle n_1 kein Eintrag definiert ist. Mit dieser Regelung wird das an dieser Stelle eigentlich benötigte error handling umgangen, um die Übersichtlichkeit der Beweise zu wahren. Es wird jedoch zusätzlich eine Operation INDEX(a,n_1) benötigt, die feststellt, ob an der Stelle n_1 in a ein Eintrag definiert ist oder nicht. Anderenfalls könnte man nicht entscheiden, falls ACCESS angewendet wird, ob an der Stelle n_1 in a kein Eintrag definiert ist oder ob dort gerade der Eintrag NULL definiert ist.

EQA ist der Gleichheitsoperator für arrays.

Die Reihenfolge, in der Einträge an verschiedenen Stellen eines arrays erfolgen, ist irrelevant. Es werden z.B. die Terme

ASSIGN(ASSIGN(NEWA,$S^4(\emptyset)$,$S^{10}(\emptyset)$),$S^7(\emptyset)$,$S^2(\emptyset)$) und

ASSIGN(ASSIGN(NEWA,$S^7(\emptyset)$,$S^2(\emptyset)$),$S^4(\emptyset)$,$S^{10}(\emptyset)$)

unter $\sim_{ESP1,a}$ identifiziert.

Erfolgen mehrere Einträge an derselben Stelle, so ist nur der letzte Eintrag relevant, d.h. die Terme

ASSIGN(ASSIGN(NEWA,$S^6(\emptyset)$,$S^3(\emptyset)$),$S^6(\emptyset)$,$S^5(\emptyset)$) und

ASSIGN(NEWA,$S^6(\emptyset)$,$S^5(\emptyset)$)

werden unter $\sim_{ESP1,a}$ ebenfalls identifiziert.

Mit REMOVE(a,n_1) können schließlich <u>alle</u> an der Stelle n_1 befindlichen Einträge (if any) gelöscht werden.

IMPL1 ist eine (t-)Erweiterung von SPEC1 und dient dazu, die Sorte <u>stack</u> mittels SPEC1 zu implementieren. Die neuen <u>stack</u>-Terme setzen sich aus einem array und einem pointer, der anzeigt, an welcher Position des arrays das Top-Element des Kellers eingetragen ist, zusammen. EQS wird auf den so zusammengesetzten <u>stack</u>-Termen neu definiert. stack-Terme, deren pointer NULL sind, werden unabhängig von der array-Komponente identifiziert und repräsentieren den leeren Keller (sh. auch unter IMPL2). Ansonsten werden solche stack-Terme identifiziert, deren pointer identisch sind und die an allen Stellen mit Index $\leq$ pointer die gleichen Einträge besitzen bzw. an denselben Stellen nicht definierte Einträge besitzen.

IMPL2 ist eine (t-)Anreicherung von IMPL1 und dient dazu, die restlichen Operationen der stack(nat)-Spezifikation SPEC0 zu implementieren. Dies geschieht durch implementierende Gleichungen für jede Operation. Die Struktur wurde bereits in IMPL1 weitgehend festgelegt, denn mit der Existenz von EQS in IMPL1 ist auch die Existenz von dis-Kontexten für stack-Terme in IMPL1 garantiert, womit die Voraussetzungen für eine nicht-triviale $\sum$IMPL1-Kongruenz $\sim_{EI1,s}$ gegeben sind, welche die aus array- und nat -Termen getupelten stack-Terme bereits in Klassen unterteilt. In IMPL2 wird also erklärt, wie die SPEC0-Operationen auf der pointer/array-Repräsentation arbeiten, womit dann auch der Rest der Struktur realisiert ist.

3.2.4.2. Nachweis der T-Implementierungseigenschaften des Beispiels

Um nachzuweisen, daß IMPL eine T-Implementierung von SPEC0 mit Hilfe von SPEC1 ist, muß gezeigt werden:

(1) SP, SP0, SP1, I1 und I2 sind vollständig,

(2) SP, SP0, SP1, I1 und I2 sind konsistent,

(3) SP0 und SP1 sind t-Erweiterungen von SP,

(4) I1 ist t-Erweiterung von SP1,

(5) I2 ist t-Anreicherung von I1 und

(6) IMPL ist R-korrekt.

Dies wird wie folgt getan:

ad (6): Es wird Punkt (iv) des Theorems 1 gezeigt; diese Bedingung ist zur R-Korrektheit äquivalent.

ad (3),(4),(5):

Die Eigenschaft der t-Erweiterung und damit auch der t-Anreicherung läßt sich nach Satz 6 aufspalten in die Eigenschaften der t-Vollständigkeit und der t-Konsistenz, welche im Folgenden nachgewiesen werden.

ad (2): Die Konsistenz läßt sich für Spezifikationen ohne bedingte Axiome wie SPEC oft über den Knuth-Bendix-Algorithmus aus [KB 70] und ähnliche Verfahren (sh. [HO 80], [H 79], [M 78]) zeigen. Hat man die Church-Rosser-Eigenschaft (CRE) mittels des KBA und die Eigenschaft der endlichen Terminierung für die aus der Axiomenmenge von SPEC gebildete Rewrite-Rule-Menge gezeigt, dann zeigt man, daß tt und ff verschiedene irreduzible Terme sind und weiß dann auch, daß tt $\neq_{E,d}$ ff, daß also SPEC konsistent ist. Im Falle der Grundspezifikation SPEC des Beispiels braucht der KBA noch nicht einmal angewendet zu werden, da die Rewrite-Rule-Menge von SPEC superpositionsfrei ist. Für Spezifikationen mit bedingten Axiomen ist der Nachweis der Konsistenz ohne weiteres jedoch nicht möglich, da der KBA noch nicht für Rewrite-Rule-Mengen mit bedingten Regeln erweitert wurde. Diese Erweiterung des KBA ist jedoch zur Zeit an der Uni Bonn im Rahmen einer Diplomarbeit in der Entwicklung. Wir wollen daher auf die Konsistenznachweise verzichten, denn die Konsistenz von SP0, SP1, I1 und I2 ist leicht einsehbar: die Bedingungen von Axiomen mit identischen linken Seiten schließen sich gegenseitig aus und die zugehörigen Rewrite-Rule-Mengen sind dann im erweiterten Sinn "superpositionsfrei".

SP, SP0, SP1, I1 und I2 seien also konsistent.

ad (1): Die Vollständigkeitsnachweise werden durch Induktion über das Gewicht der Terme erbracht, wobei im Induktionsschluß über den Aufbau der Terme, d.h. nach dem äußersten auftretenden Operationssymbol unterschieden wird. Diese Art des Induktionsbeweises wurde aus folgenden Gründen gewählt: Im Induktionsschritt muß sehr oft gezeigt werden, daß eine bestimmte Gleichung auf den zu betrachtenden Term "anwendbar" ist. Enthält die Gleichung Bedingungen, dann muß auch noch gezeigt werden, daß diese "auswertbar" sind, daß die linke Seite einer Bedingung zur rechten i-kongruent ist. Man möchte beispielsweise zeigen, daß der SPEC0-Term $t := EQS(PUSH(p_1,r_1),PUSH(p_2,r_2))$ zu tt oder ff i-kongruent ist. Dazu muß man etwa zeigen, daß die Gleichung E0.5 auf t anwendbar ist, d.h. man muß wissen, daß die Bedingung $EQN(r_1,r_2) \doteq ff$ erfüllt ist, daß also $EQN(r_1,r_2)$ zu ff i-kongruent ist. Da $EQN(r_1,r_2)$ kein Subterm von t ist, kann strukturelle Induktion hier nicht angewendet werden. Man weiß aber, daß $EQN(r_1,r_2)$ ein kleineres Gewicht als t hat. Daher kann im Vollständigkeitsbeweis die Induktionsannahme benutzt werden, um sicherzugehen, daß die Bedingung einer Gleichung erfüllt ist.

Wir brauchen nicht zu zeigen, daß SP, SP0 und SP1 t-Spezifikationen bzw. daß I1 und I2 t-Kombinationen sind, denn dies ist nach Aufschreibung und 2.2. und 3.2.3. klar.

Zunächst zeigen wir, daß SP vollständig ist.

<u>Lemma V.SP</u>

<u>Beh.:</u> $(\forall t \in T_{\Sigma,d})(\exists tv \in \{tt,ff\})(t \equiv_{E,d} tv)$

<u>Bew.:</u>

Durch Induktion über das Gewicht $w(t)$ der Terme $t \in T_{\Sigma,d}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma,d}$ mit $w(t)=1$.

Daraus folgt, daß entweder t <u>is</u> tt oder t <u>is</u> ff gilt.

In beiden Fällen folgt die Gültigkeit der Behauptung aus der Reflexivität der Relation $\equiv_{E,d}$.

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t \in T_{\Sigma,d}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma,d}$ mit $w(t)=n+1$.

Wir unterscheiden nach dem Aufbau von t, d.h. nach dem äußersten in t vorkommenden Operationssymbol. Da $w(t) \geq 2$ ist, gilt t ~~is~~ tt und t ~~is~~ ff. Es bleiben daher vier Fälle zu untersuchen.

iii.a) t <u>is</u> $\neg t'$ mit $t' \in T_{\Sigma,d}$.

Es gilt $w(t') \leq n$ und nach Induktionsannahme ii) gibt es ein $tv' \in \{tt,ff\}$, sodaß $t' \equiv_{E,d} tv'$. Aus Gleichung E.1 bzw. E.2 und der Substitutionseigenschaft (SE) von $\equiv_E$ folgt dann die Gültigkeit der Behauptung für dieses t.

iii.b) t <u>is</u> t_1 AND t_2 mit $t_1, t_2 \in T_{\Sigma,d}$.

Es gilt $w(t_1)+w(t_2) \leq n$ und daher gibt es nach Induktionsannahme ii) Terme $tv_1, tv_2 \in \{tt,ff\}$, sodaß $t_i \equiv_{E,d} tv_i$ für $i=1,2$. Aus der SE von $\equiv_E$ und aus Gleichung E.3 bzw. E.4 folgt dann die Gültigkeit der Behauptung für dieses t.

iii.c) t <u>is</u> t_1 EQV t_2 mit $t_1,t_2 \in T_{\Sigma,d}$.

Dieser Unterfall entspricht dem Unterfall iii.b), wobei jedoch die Gleichungen E.5 und E.6 verwendet werden.

iii.d) t <u>is</u> EQN(t_1,t_2) mit $t_1,t_2 \in T_{\Sigma,n}$.

Wir unterscheiden weiter nach dem Aufbau von t_1 und t_2. Für die Fälle mit t_1 <u>is</u> $\emptyset$, t_2 <u>is</u> $\emptyset$ und t_1 <u>is</u> $\emptyset$, t_2 <u>is</u> $S^k(\emptyset)$ und t_1 <u>is</u> $S^k(\emptyset)$, t_2 <u>is</u> $\emptyset$ mit $k \in \omega$ ergibt sich die Gültigkeit der Behauptung direkt aus den Gleichungen E.7 bis E.9. Es bleibt der Fall

iii.d.4) t_1 <u>is</u> $S^j(\emptyset)$, t_2 <u>is</u> $S^k(\emptyset)$ mit $j,k \in \omega$.

Nach Gleichung E.10 gilt

*1 $EQN(S^j(\emptyset),S^k(\emptyset)) \equiv_{E,d} EQN(S^{j-1}(\emptyset),S^{k-1}(\emptyset))$.

Außerdem ist $w(EQN(S^{j-1}(\emptyset),S^{k-1}(\emptyset))) \leq n$. Nach Induktionsannahme ii) gibt es daher ein $tv \in \{tt,ff\}$, sodaß

*2 $EQN(S^{j-1}(\emptyset),S^{k-1}(\emptyset)) \equiv_{E,d} tv$.

Aus *1 und *2 folgt mittels der Transitivität von $\equiv_{E,d}$ die Gültigkeit der Behauptung für $EQN(S^j(\emptyset),S^k(\emptyset))$.

Damit ist die Induktion geschlossen und die Vollständigkeit von SP gezeigt. $\square$

Bevor wir zeigen, daß SP0 vollständig ist, wollen wir die Existenz von "Normalformen" für <u>stack</u>- und <u>nat</u>-Terme in $T_{\Sigma \cup \Sigma 0}$ nachweisen, d.h. wir zeigen, daß jeder <u>stack</u>- bzw. <u>nat</u>-Term zu einem "kanonischen" <u>stack</u>- bzw. <u>nat</u>-Term i-kongruent ist. Dies trägt zur Vereinfachung des Vollständigkeitsnachweises von SP0 bei, denn die Anzahl der zu betrachtenden Unterfälle in den Induktionsbeweisen wird dadurch verringert.

<u>Lemma S.SP0</u>

<u>Beh.:</u> $(\forall t \in T_{\Sigma\cup\Sigma 0,s})(\exists t' \in T_{\Sigma\cup\Sigma 0,s})(\exists m \in \omega_0)(\exists q_1,\dots,q_m \in T_{\Sigma\cup\Sigma 0,n})$

(1) $t \equiv_{E\cup E0,s} t' \wedge w(t') \leq w(t) \wedge$

2) t' <u>is</u> PUSH(...PUSH(NEWST,q_1),...,q_m))

Bew.:

Durch Induktion über das Gewicht $w(t)$ der Terme $t \in T_{\Sigma \cup \Sigma 0,s}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma \cup \Sigma 0,s}$ mit $w(t)=1$.

Daraus folgt t is NEWST. Also ist $m=0$ und der Rest ist Reflexivität von $\equiv_{E \cup E0,s}$ und $\leq$.

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t \in T_{\Sigma \cup \Sigma 0,s}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma \cup \Sigma 0,s}$ mit $w(t)=n+1$.

Aus $w(t) \geq 2$ folgt t ~~is~~ NEWST und es bleiben zwei Fälle zu unterscheiden.

iii.a) t is PUSH(p,q) mit $p \in T_{\Sigma \cup \Sigma 0,s}$ und $q \in T_{\Sigma \cup \Sigma 0,n}$.

Es gilt $w(p) \leq n$ und nach Induktionsannahme ii) gibt es ein $p' \in T_{\Sigma \cup \Sigma 0,s}$ mit $p \equiv_{E \cup E0,s} p'$ und $w(p') \leq w(p)$ und p' hat die unter 2) der Beh. geforderte Form.

Wegen der SE von $\equiv_{E \cup E0}$ folgt

$$PUSH(p,q) \equiv_{E \cup E0,s} PUSH(p',q).$$

Außerdem ist $w(PUSH(p',q)) \leq w(PUSH(p,q))$, d.h. 1) ist erfüllt und mit $q_{m+1} := q$ hat PUSH(p',q) ebenfalls die unter 2) geforderte Form.

iii.b) t is POP(p) mit $p \in T_{\Sigma \cup \Sigma 0,s}$.

Es gilt $w(p) \leq n$ und nach Induktionsannahme ii) gibt es ein $p' \in T_{\Sigma \cup \Sigma 0,s}$ mit $p \equiv_{E \cup E0,s} p'$ und $w(p) \leq w(p')$ und p' hat die unter 2) geforderte Form.

Dann bleiben zwei Möglichkeiten.

iii.b.1) p' is NEWST, d.h. $m=0$.

Es gilt

$$POP(p) \equiv_{E \cup E0,s} POP(NEWST) \equiv_{E \cup E0,s} NEWST$$

nach SE von $\equiv_{E \cup E0}$ und nach Gleichung E0.8. Außerdem gilt $w(NEWST)=1 \leq w(POP(p))$ und NEWST hat die passende Form.

iii.b.2) p' <u>is</u> $PUSH(\ldots PUSH(NEWST,q_1),\ldots,q_m)$ mit $m \in \omega_0$, $m>0$ und $q_1,\ldots,q_m \in T_{\Sigma \cup \Sigma 0,n}$.

Es gilt

$$\begin{aligned} POP(p) &\equiv_{E \cup E0,s} POP(PUSH(\ldots PUSH(NEWST,q_1),\ldots,q_m)) \\ &\equiv_{E \cup E0,s} PUSH(\ldots PUSH(NEWST,q_1),\ldots,q_{m-1}) \end{aligned}$$

nach SE von $\equiv_{E \cup E0}$ und nach Gleichung E0.9. Der letztgenannte Term hat die unter 2) geforderte Form. Außerdem ist $w(PUSH(\ldots,q_{m-1}))<w(PUSH(\ldots,q_m)) \leq w(p)<w(POP(p))=n+1$.

Damit ist die Induktion geschlossen und Lemma S.SP0 bewiesen. □

<u>Lemma N.SP0</u>

<u>Beh.:</u> $(\forall t \in T_{\Sigma \cup \Sigma 0,n})(\exists t' \in T_{\Sigma \cup \Sigma 0,n})(\exists k \in \omega_0)$

$(\;1)\; t \equiv_{E \cup E0,n} t' \;\wedge\; w(t') \leq w(t) \;\wedge$

$\;\;2)\; t'$ <u>is</u> $S^k(\emptyset) \;)$

<u>Bew.:</u>

Durch Induktion über das Gewicht w(t) der Terme $t \in T_{\Sigma \cup \Sigma 0,n}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma \cup \Sigma 0,n}$ mit $w(t)=1$.

Daraus folgt t <u>is</u> $\emptyset$. Es ist k=0 und der Rest ist Reflexivität von $\equiv_{E \cup E0,n}$ und von $\leq$.

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t \in T_{\Sigma \cup \Sigma 0,n}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma \cup \Sigma 0,n}$ mit $w(t)=n+1$.

Aus $w(t) \geq 2$ folgt t ~~is~~ $\emptyset$. Wir betrachten die Fälle iii.a) t <u>is</u> $S(t')$ mit $t' \in T_{\Sigma \cup \Sigma 0, n}$.

Es gilt $w(t') \leq n$. Nach Induktionsannahme ii) gibt es ein $k \in \omega_0$, sodaß $t' \equiv_{E \cup E0, n} S^k(\emptyset)$ und $w(S^k(\emptyset)) \leq w(t')$.

Dann gilt wegen der SE von $\equiv_{E \cup E0}$ auch

$$S(t') \equiv_{E \cup E0, n} S(S^k(\emptyset)) \text{ \underline{is} } S^{k+1}(\emptyset) \quad .$$

Es ist $w(S^{k+1}(\emptyset)) \leq w(S(t'))$ und $S^{k+1}(\emptyset)$ hat die unter 2) geforderte Form.

iii.b) t <u>is</u> TOP(p) mit $p \in T_{\Sigma \cup \Sigma 0, s}$.

Nach Lemma S.SP0 gibt es ein $p' \in T_{\Sigma \cup \Sigma 0, s}$, sodaß $p \equiv_{E \cup E0, s} p'$ und $w(p') \leq w(p)$ und es gibt $m \in \omega_0$ und $q_1, \ldots, q_m \in T_{\Sigma \cup \Sigma 0, n}$, sodaß p' <u>is</u> PUSH(...PUSH(NEWST,q_1),...,q_m).

Jetzt bleiben zwei Fälle zu betrachten.

iii.b.1) p' <u>is</u> NEWST, d.h. m=0.

Es gilt

$$TOP(p) \equiv_{E \cup E0, n} TOP(NEWST) \equiv_{E \cup E0, n} \emptyset$$

Nach SE von $\equiv_{E \cup E0}$ und nach Gleichung E0.6. Es ist $w(\emptyset) = 1 \leq w(TOP(p))$, d.h. 1) und 2) sind für dieses t erfüllt.

iii.b.2) p' <u>is</u> PUSH(...PUSH(NEWST,q_1),...,q_m) mit $m \in \omega_0$, $m > 0$ und $q_1, \ldots, q_m \in T_{\Sigma \cup \Sigma 0, n}$.

Es gilt

$$\begin{aligned} TOP(p) &\equiv_{E \cup E0, n} TOP(PUSH(\ldots PUSH(NEWST, q_1), \ldots, q_m)) \\ &\equiv_{E \cup E0, n} q_m \end{aligned}$$

nach SE von $\equiv_{E \cup E0}$ und nach Gleichung E0.7. Es gilt außerdem $w(q_m) < w(PUSH(\ldots PUSH(NEWST, q_1), \ldots, q_m)) \leq w(p) \leq n < w(TOP(p))$.

Nach Induktionsannahme ii) gibt es daher ein $t' \in T_{\Sigma \cup \Sigma 0, n}$, sodaß $q_m \equiv_{E \cup E0, n} t'$, $w(t') \leq w(q_m)$ und t' hat die unter 2) geforderte Form. Der Rest ist Transitivität von $\equiv_{E \cup E0, n}$ und $\leq$.

Damit ist die Induktion geschlossen und Lemma N.SP0 gezeigt. □

Lemma V.SP0

Beh.: $(\forall t \in T_{\Sigma \cup \Sigma 0,d})(\exists tv \in \{tt,ff\})(t \equiv_{E \cup E0,d} tv)$

Bew.:

Durch Induktion über das Gewicht $w(t)$ der Terme $t \in T_{\Sigma \cup \Sigma 0,d}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma \cup \Sigma 0,d}$ mit $w(t)=1$.

Daraus folgt, daß entweder t is tt oder t is ff. In beiden Fällen folgt die Gültigkeit der Behauptung aus der Reflexivität von $\equiv_{E \cup E0,d}$.

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t \in T_{\Sigma \cup \Sigma 0,d}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma \cup \Sigma 0,d}$ mit $w(t)=n+1$.

Aus $w(t) \geq 2$ folgt, daß sowohl t ~~is~~ tt als auch t ~~is~~ ff.

Dann unterscheiden wir folgende Fälle.

iii.a) t is $\neg t'$ mit $t' \in T_{\Sigma \cup \Sigma 0,d}$,

iii.b) t is t_1 AND t_2 mit $t_1, t_2 \in T_{\Sigma \cup \Sigma 0,d}$ und

iii.c) t is t_1 EQV t_2 mit $t_1, t_2 \in T_{\Sigma \cup \Sigma 0,d}$

verlaufen völlig analog zu den Punkten iii.a) bis iii.c) aus dem Beweis von Lemma V.SP. Es bleiben die Fälle

iii.d) t is $EQN(t_1,t_2)$ mit $t_1, t_2 \in T_{\Sigma \cup \Sigma 0,n}$.

Nach Lemma N.SP0 gibt es $t_3, t_4 \in T_{\Sigma \cup \Sigma 0,n} \cap T_{\Sigma,n}$, sodaß $t_i \equiv_{E \cup E0,n} t_{i+2}$ für $i=1,2$. Daraus folgt mittels SE von $\equiv_{E \cup E0}$

$$EQN(t_1,t_2) \equiv_{E \cup E0,d} EQN(t_3,t_4),$$

wobei $EQN(t_3,t_4) \in T_{\Sigma \cup \Sigma 0,d} \cap T_{\Sigma,d}$. Nach Lemma V.SP gibt es daher ein $tv \in \{tt,ff\}$, sodaß

$$EQN(t_3,t_4) \equiv_{E,d} tv.$$

Nach Lemma 7 folgt daraus

$$EQN(t_1,t_2) \equiv_{E\cup E0,d} tv.$$

Der Rest ist Transitivität von $\equiv_{E\cup E0,d}$.

iii.e) t <u>is</u> $EQS(t_1,t_2)$ mit $t_1,t_2 \in T_{\Sigma\cup\Sigma 0,s}$.

Nach Lemma S.SP0 gibt es $t_3,t_4 \in T_{\Sigma\cup\Sigma 0,s}$, sodaß

$t_i \equiv_{E\cup E0,s} t_{i+2}$ und $w(t_{i+2}) \leq w(t_i)$ und entweder

t_{i+2} <u>is</u> NEWST oder t_{i+2} <u>is</u> $PUSH(p_i,q_i)$ mit $p_i \in T_{\Sigma\cup\Sigma 0,s}$

und $q_i \in T_{\Sigma\cup\Sigma 0,n}$ für i=1,2. Dabei haben wir die genauere Aussage von Lemma S.SP0 durch Verwendung von p_1 und p_2 etwas abgekürzt, denn es kommt im Folgenden nur darauf an, daß t_3 bzw. t_4 nicht mit POP beginnen.

Die SE von $\equiv_{E\cup E0}$ liefert jetzt

*3 $EQS(t_1,t_2) \equiv_{E\cup E0,d} EQS(t_3,t_4)$.

Für die Fälle t_3 <u>is</u> NEWST, t_4 <u>is</u> NEWST,

t_3 <u>is</u> NEWST, t_4 <u>is</u> $PUSH(p_2,q_2)$ und

t_3 <u>is</u> $PUSH(p_1,q_1)$, t_4 <u>is</u> NEWST

ergibt sich die Gültigkeit der Behauptung aus den Gleichungen E0.1 bis E0.3 und aus *3 zusammen mit der Transitivität von $\equiv_{E\cup E0,d}$. Es bleibt der Fall

iii.e.4) t_3 <u>is</u> $PUSH(p_1,q_1)$, t_4 <u>is</u> $PUSH(p_2,q_2)$.

Es gilt $w(q_1)+w(q_2)<n$ und daher $w(EQN(q_1,q_2)) \leq n$. Nach Induktionsannahme ii) gibt es ein $tv \in \{tt,ff\}$, sodaß

$EQN(q_1,q_2) \equiv_{E\cup E0,d} tv$, d.h. wir können die Bedingungen der Gleichungen E0.4 und E0.5 "auswerten".

iii.e.4.1) tv <u>is</u> tt.

Wir können Gleichung E0.4 anwenden und erhalten

*4 $EQS(PUSH(p_1,q_1),PUSH(p_2,q_2)) \equiv_{E\cup E0,d} EQS(p_1,p_2)$,

wobei $w(EQS(p_1,p_2)) < w(EQS(t_3,t_4)) \leq w(EQS(t_1,t_2)) = n+1$.

Nach Induktionsannahme ii) gibt es daher ein $tv' \in \{tt,ff\}$,

sodaß

$*5$ $EQS(p_1,p_2) \equiv_{E\cup E0,d} tv'$.

Aus $*3$, $*4$ und $*5$ folgt mittels Transitivität von $\equiv_{E\cup E0,d}$ die Gültigkeit der Behauptung für diesen Fall.

iii.e.4.2) tv <u>is</u> ff.

Wir können die Gleichung E0.5 anwenden und erhalten

$*6$ $EQS(PUSH(p_1,q_1),PUSH(p_2,q_2)) \equiv_{E\cup E0,d}$ ff.

Aus $*3$ und $*6$ folgt mittels Transitivität von $\equiv_{E\cup E0,d}$ die Gültigkeit der Behauptung für diesen Fall.

Damit ist die Induktion geschlossen und die Vollständigkeit von SP0 gezeigt. $\square$

Im Beweis des nächsten Lemmas müssen wir im Induktionsschritt bereits wissen, daß zu <u>array</u>-Subtermen von <u>nat</u>-Termen i-kongruente kanonische <u>array</u>-Terme existieren. Da wir Lemma A.SP1 noch nicht zur Verfügung haben, müssen wir dies durch eine weitere Induktion zeigen. Derartige Verschachtelungen von Induktionen werden auch in den weiteren Beweisen benötigt.

<u>Lemma N.SP1</u>

<u>Beh.:</u> $(\forall t\in T_{\Sigma\cup\Sigma 1,n})(\exists t'\in T_{\Sigma\cup\Sigma 1,n})(\exists j\in\omega_0)$

(1) $t \equiv_{E\cup E1,n} t'$ $\wedge$ $w(t')\leq w(t)$ $\wedge$

2) t' <u>is</u> $S^j(\emptyset)$)

<u>Bew.:</u>

Durch Induktion über das Gewicht $w(t)$ der Terme $t\in T_{\Sigma\cup\Sigma 1,n}$.

i) Induktionsanfang.

Sei $t\in T_{\Sigma\cup\Sigma 1,n}$ mit $w(t)=1$.

Daraus folgt t <u>is</u> $\emptyset$ und die Behauptung gilt aufgrund der Reflexivität von $\equiv_{E\cup E0,n}$ und $\leq$. Es ist $j=0$.

ii) Induktionsannahme.

Sei $n\in\omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t \epsilon T_{\Sigma \cup \Sigma 1,n}$ mit $w(t) \le n$.

iii) Induktionsschluß.

Sei $t \epsilon T_{\Sigma \cup \Sigma 1,n}$ mit $w(t)=n+1$.

Es ist $w(t) \ge 2$ und daher t ~~is~~ $\emptyset$.

Es bleiben zwei Fälle zu untersuchen.

iii.a) t <u>is</u> S(t') mit $t' \epsilon T_{\Sigma \cup \Sigma 1,n}$.

Wegen $w(t') \le n$ und nach Induktionsannahme ii) gibt es ein $j \epsilon \omega_0$, sodaß $t' \equiv_{E \cup E1,n} S^j(\emptyset)$ und $w(S^j(\emptyset)) \le w(t')$.

Nach SE von $\equiv_{E \cup E1}$ gilt $S(t') \equiv_{E \cup E1,n} S(S^j(\emptyset))$.

Der letztgenannte Term hat die unter 2) geforderte Form und es gilt $w(S(S^j(\emptyset))) \le w(S(t'))$, d.h. auch 1) ist erfüllt.

iii.b) t <u>is</u> ACCESS(q,r) mit $q \epsilon T_{\Sigma \cup \Sigma 1,a}$ und $r \epsilon T_{\Sigma \cup \Sigma 1,n}$.

Um die Gleichungen E1.10 bis E1.12 anwenden zu können, müssen wir zeigen, daß es zu q einen i-kongruenten <u>array</u>-Term gibt, der nicht mit REMOVE beginnt und ein kleineres oder gleiches Gewicht hat. Wir zeigen daher zunächst

*7 Für alle $q' \epsilon T_{\Sigma \cup \Sigma 1,a}$ mit $w(q') \le w(q)$ gilt

$(\exists q'' \epsilon T_{\Sigma \cup \Sigma 1,a}) (\exists m \epsilon \omega_0) (\exists p_1,\dots,p_m,r_1,\dots,r_m \epsilon T_{\Sigma \cup \Sigma 1,n})$

(3) $q' \equiv_{E \cup E1,a} q''$ $\wedge$ $w(q'') \le w(q')$ $\wedge$

4) q" <u>is</u> ASSIGN(...ASSIGN(NEWA,p_1,r_1),...,p_m,r_m)).

Der Beweis erfolgt durch Induktion über das Gewicht $w(q')$ der Terme $q' \epsilon T_{\Sigma \cup \Sigma 1,a}$ mit $w(q') \le w(q)$.

iii.b.i) Induktionsanfang.

Sei $q' \epsilon T_{\Sigma \cup \Sigma 1,a}$ mit $w(q')=1$.

Daraus folgt q' <u>is</u> NEWA. Es ist m=0 und der Rest ist Reflexivität von $\equiv_{E \cup E1,a}$ und $\le$.

iii.b.ii) Induktionsannahme.

Sei $k \epsilon \omega$ fest mit $k < w(q)$.

Die Behauptung gelte für alle $q' \in T_{\Sigma \cup \Sigma 1, a}$ mit $w(q') \leq k$.

iii.b.iii) Induktionsschluß.

Sei $q' \in T_{\Sigma \cup \Sigma 1, a}$ mit $w(q') = k+1$.

Es ist $w(q') \geq 2$ und daraus folgt q' ~~is~~ NEWA.

Es bleiben folgende Fälle zu betrachten:

iii.b.iii.a) q' <u>is</u> AS(q",x,y) mit $q" \in T_{\Sigma \cup \Sigma 1, a}$ und $x, y \in T_{\Sigma \cup \Sigma 1, n}$.

Wegen $w(q") \leq k$ und Induktionsannahme iii.b.ii) gibt es ein $q"' \in T_{\Sigma \cup \Sigma 1, a}$, sodaß $q" \equiv_{E \cup E1, a} q"'$, $w(q"') \leq w(q")$ und $q"'$ hat die unter 4) geforderte Form. Nach SE von $\equiv_{E \cup E1}$ gilt

$$AS(q",x,y) \equiv_{E \cup E1, a} AS(q"',x,y)$$

und der letztgenannte Term hat ebenfalls die unter 4) geforderte Form. Es ist $w(AS(q"',x,y)) \leq w(AS(q",x,y))$.

iii.b.iii.b) q' <u>is</u> REMOVE(q",x) mit $q" \in T_{\Sigma \cup \Sigma 1, a}$ und $x \in T_{\Sigma \cup \Sigma 1, n}$.

Wegen $w(q") \leq k$ und Induktionsannahme iii.b.ii) gibt es ein $q"' \in T_{\Sigma \cup \Sigma 1, a}$, sodaß $q" \equiv_{E \cup E1, a} q"'$, $w(q"') \leq w(q")$ und $q"'$ hat die unter 4) geforderte Form. Nach SE von $\equiv_{E \cup E1}$ gilt

*8 $RE(q",x) \equiv_{E \cup E1, a} RE(q"',x)$.

Jetzt gibt es zwei Unterfälle.

iii.b.iii.b.1) $q"'$ <u>is</u> NEWA, d.h. $m=0$.

Nach Gleichung E1.13 gilt

*9 $RE(NEWA,x) \equiv_{E \cup E1, a} NEWA$.

Es gilt $w(NEWA) = 1 \leq w(RE(q",x))$ und der Rest ist Transitivität von $\equiv_{E \cup E1, a}$ bzgl. *8 und *9.

iii.b.iii.b.2) $q"'$ <u>is</u> $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$ mit

$m \in \omega_0$, $m>0$ und $p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma \cup \Sigma 1, n}$.

Um die Gleichungen E1.14 bzw. E1.15 anwenden zu können, müssen wir zeigen, daß die Bedingungen derselben "auswertbar" sind:

Es gilt $w(q'')+w(x)\leq k$ und $w(q''')\leq w(q'')$, also auch $w(q''')+w(x)\leq k$ und schließlich $w(p_m)+w(x)\leq k\leq n$. Nach Induktionsannahme ii) gibt es daher $j_1, j_2 \in \omega_0$, sodaß

$$*10 \quad p_m \equiv_{E\cup E1,n} S^{j_1}(\emptyset) \quad , \quad x \equiv_{E\cup E1,n} S^{j_2}(\emptyset) \quad ,$$

$$*11 \quad w(S^{j_1}(\emptyset))\leq w(p_m) \quad , \quad w(S^{j_2}(\emptyset))\leq w(x)$$

und $S^{j_1}(\emptyset), S^{j_2}(\emptyset) \in T_{\Sigma\cup\Sigma 1,n} \cap T_{\Sigma,n}$. Damit ist aber auch $EQN(S^{j_1}(\emptyset), S^{j_2}(\emptyset)) \in T_{\Sigma\cup\Sigma 1,d} \cap T_{\Sigma,d}$ und nach Lemma V.SP gibt es ein $tv \in \{tt, ff\}$, sodaß

$$EQN(S^{j_1}(\emptyset), S^{j_2}(\emptyset)) \equiv_{E,d} tv,$$

woraus nach Lemma 7 folgt

$$*12 \quad EQN(S^{j_1}(\emptyset), S^{j_2}(\emptyset)) \equiv_{E\cup E1,d} tv.$$

Aus *10 folgt nach SE von $\equiv_{E\cup E1}$

$$*13 \quad RE(AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m),x) \equiv_{E\cup E1,a}$$
$$RE(AS(\ldots AS(NA,p_1,r_1),\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)).$$

Wegen *12 können wir jetzt die Gleichungen E1.14 und E1.15 anwenden.

iii.b.iii.b.2.1) tv <u>is</u> tt.

Nach Gleichung E1.14 gilt

$$*14 \quad RE(AS(\ldots AS(NA,p_1,r_1),\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)) \equiv_{E\cup E1,a}$$
$$RE(AS(\ldots AS(NA,p_1,r_1),\ldots,p_{m-1},r_{m-1}),S^{j_2}(\emptyset)).$$

Es gilt wegen *11

$$w(RE(AS(\ldots,r_{m-1}),S^{j_2}(\emptyset)))<w(RE(AS(\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)))$$
$$\leq w(RE(q'',x)) = k+1.$$

Nach Induktionsannahme iii.b.ii) gibt es daher ein $q^* \in T_{\Sigma\cup\Sigma 1,a}$, sodaß

$$*15 \quad RE(AS(\ldots AS(NA,p_1,r_1),\ldots,p_{m-1},r_{m-1}),S^{j_2}(\emptyset)) \equiv_{E\cup E1,a} q^*$$

$$*16 \quad w(q^*)\leq w(RE(AS(\ldots,r_{m-1}),S^{j_2}(\emptyset)))$$

und q^* hat die passende Form. Dann folgt die Gültigkeit der Behauptung für diesen Fall aus der Transitivität von $\equiv_{E\cup E1,a}$

zusammen mit *8, *13, *14 und *15 und aus der Transitivität von $\leq$.

iii.b.iii.b.2.2) tv <u>is</u> ff.

Nach Gleichung E1.15 gilt

$$*17 \quad RE(AS(\ldots AS(NA,p_1,r_1),\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)) \equiv_{E\cup E1,a}$$

$$AS(RE(AS(\ldots AS(NA,p_1,r_1),\ldots,p_{m-1},r_{m-1}),S^{j_2}(\emptyset)),S^{j_1}(\emptyset),r_m).$$

Es gilt wegen *11 wieder $w(RE(AS(\ldots,r_{m-1}),S^{j_2}(\emptyset)))\leq k$ und nach Induktionsannahme iii.b.ii) gibt es ein $q^* \epsilon T_{\Sigma\cup\Sigma1,a}$, sodaß

$$*18 \quad RE(AS(\ldots AS(NA,p_1,r_1),\ldots,p_{m-1},r_{m-1}),S^{j_2}(\emptyset)) \equiv_{E\cup E1,a} q^*$$

$$*19 \quad w(q^*)\leq w(RE(AS(\ldots,r_{m-1}),S^{j_2}(\emptyset)))$$

und q^* hat die unter 4) geforderte Form. Aus der SE von $\equiv_{E\cup E1}$ folgt mit *17 und *18 auch

$$*20 \quad RE(AS(\ldots AS(NA,p_1,r_1),\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)) \equiv_{E\cup E1,a}$$

$$AS(q^*,S^{j_1}(\emptyset),r_m),$$

wobei $AS(q^*,S^{j_1}(\emptyset),r_m)$ ebenfalls die unter 4) geforderte Form besitzt. Es gilt $w(AS(q^*,S^{j_1}(\emptyset),r_m))\leq w(RE(q",x))$ und der Rest ist Transitivität von $\equiv_{E\cup E1,a}$ zusammen mit *8, *13 und *20.

Damit ist die Induktion unter iii.b) geschlossen.

Nach *7 wissen wir jetzt, daß es zu dem Subterm q von ACCESS(q,r) ein $q' \epsilon T_{\Sigma\cup\Sigma1,a}$ gibt, sodaß $q \equiv_{E\cup E1,a} q'$, $w(q')\leq w(q)$ und entweder q' <u>is</u> NEWA oder es gibt ein $m \epsilon \omega_0$, $m>0$, und $p_1,\ldots,p_m,r_1,\ldots,r_m \epsilon T_{\Sigma\cup\Sigma1,n}$, sodaß

q' <u>is</u> $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$.

iii.b.1) q' <u>is</u> NEWA, d.h. m=0.

Nach SE von $\equiv_{E\cup E1}$ und nach Gleichung E1.10 gilt

$$ACCESS(q,r) \equiv_{E\cup E1,n} ACCESS(NEWA,r) \equiv_{E\cup E1,n} \emptyset.$$

Es gilt $w(\emptyset)=1\leq w(ACCESS(q,r))$.

iii.b.2) q is $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m',r_m)$'.

Zunächst zeigen wir, daß die Bedingungen der Gleichungen E1.11 und E1.12 auswertbar sind und betrachten dann die Unterfälle, die sich durch Anwendung der Gleichungen ergeben.

Es gilt $w(q)+w(r)\leq n$ und $w(p_m)\leq w(q)$. Nach Induktionsannahme ii) gibt es daher $j_1,j_2\in\omega_0$, sodaß

*21 $p_m \equiv_{E\cup E1,n} S^{j_1}(\emptyset)$, $r \equiv_{E\cup E1,n} S^{j_2}(\emptyset)$,

*22 $w(S^{j_1}(\emptyset))\leq w(p_m)$, $w(S^{j_2}(\emptyset))\leq w(r)$

und $S^{j_1}(\emptyset),S^{j_2}(\emptyset)\in T_{\Sigma\cup\Sigma 1,n}\cap T_{\Sigma,n}$. Damit ist aber auch $EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset))\in T_{\Sigma\cup\Sigma 1,d}\cap T_{\Sigma,d}$. Nach Lemma V.SP gibt es daher ein $tv\in\{tt,ff\}$, sodaß

$$EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset)) \equiv_{E,d} tv.$$

Nach Lemma 7 folgt daraus

*23 $EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset)) \equiv_{E\cup E1,d} tv.$

Aus *21 folgt nach SE von $\equiv_{E\cup E1}$

*24 $AC(q,r) \equiv_{E\cup E1,n} AC(AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m),r)$

$\equiv_{E\cup E1,n} AC(AS(\ldots AS(NA,p_1,r_1),\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset))$.

Wegen *23 können wir jetzt die Gleichung E1.11 bzw. E1.12 anwenden.

iii.b.2.1) tv is tt.

Nach Gleichung E1.11 gilt

*25 $AC(AS(\ldots AS(NA,p_1,r_1),\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)) \equiv_{E\cup E1,n} r_m.$

Es gilt $w(r_m)<w(q')\leq w(q)<n+1$. Daher gibt es nach Induktionsannahme ii) ein $r'\in T_{\Sigma\cup\Sigma 1,n}$, sodaß

*26 $r_m \equiv_{E\cup E1,n} r'$,

$w(r')\leq w(r_m)$ und r' hat die unter 2) geforderte Form. Es gilt $w(r')\leq w(ACCESS(q,r))$ und der Rest ist Transitivität von $\equiv_{E\cup E1,n}$ zusammen mit *24, *25 und *26.

iii.b.2.2) tv <u>is</u> ff.

Nach Gleichung E1.12 gilt

$*27\quad AC(AS(\ldots AS(NA,p_1,r_1),\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)) \equiv_{E\cup E1,n}$

$AC(AS(\ldots AS(NA,p_1,r_1),\ldots,p_{m-1},r_{m-1}),S^{j_2}(\emptyset)).$

Es gilt

$*28\quad w(AC(AS(\ldots,r_{m-1}),S^{j_2}(\emptyset)))<w(AC(AS(\ldots,S^{j_1}(\emptyset),r_m),S^{j_2}(\emptyset)))$

$\leq w(AC(AS(\ldots,p_m,r_m),r)) = n+1.$

Nach Induktionsannahme ii) gibt es daher ein $r' \in T_{\Sigma\cup\Sigma1,n}$, sodaß

$*29\quad AC(AS(\ldots AS(NA,p_1,r_1),\ldots,p_{m-1},r_{m-1}),S^{j_2}(\emptyset)) \equiv_{E\cup E1,n} r'$

$*30\quad w(r')\leq w(AC(AS(\ldots,r_{m-1}),S^{j_2}(\emptyset)))$

und r' hat die unter 2) geforderte Form. Der Rest ist Transitivität von $\equiv_{E\cup E1,n}$ zusammen mit *24, *27 und *29 und von $\leq$ zusammen mit *28 und *30.

Damit ist die Induktion geschlossen und Lemma N.SP1 gezeigt. □

<u>Lemma A.SP1</u>

<u>Beh.:</u> $(\forall t\in T_{\Sigma\cup\Sigma1,a})(\exists t'\in T_{\Sigma\cup\Sigma1,a})(\exists m\in\omega_0)$

$(\exists p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma\cup\Sigma1,n})$

(1) $t \equiv_{E\cup E1,a} t' \;\wedge\; w(t')\leq w(t) \;\wedge$

2) t' <u>is</u> $ASSIGN(\ldots ASSIGN(NEWA,p_1,r_1),\ldots,p_m,r_m)$)

<u>Bew.:</u>

Durch Induktion über das Gewicht w(t) der Terme $t\in T_{\Sigma\cup\Sigma1,a}$.

i) Induktionsanfang.

Sei $t\in T_{\Sigma\cup\Sigma1,a}$ mit $w(t)=1$.

Daraus folgt t <u>is</u> NEWA. Der Rest ist Reflexivität von $\leq$ und von $\equiv_{E\cup E1,a}$. Es ist $m=0$.

ii) Induktionsannahme.

Sei $n\in\omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t\in T_{\Sigma\cup\Sigma1,a}$ mit $w(t)\leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma \cup \Sigma 1,a}$ mit $w(t)=n+1$.

Da $w(t) \geq 2$ ist, gilt t ~~is~~ NEWA. Es bleiben zwei Fälle.

iii.a) t is ASSIGN(q,x,y) mit $q \in T_{\Sigma \cup \Sigma 1,a}$ und $x,y \in T_{\Sigma \cup \Sigma 1,n}$.

Es ist $w(q) \leq n$. Nach Induktionsannahme ii) gibt es ein $q' \in T_{\Sigma \cup \Sigma 1,a}$, sodaß $q \equiv_{E \cup E1,a} q'$, $w(q') \leq w(q)$ und q' hat die unter 2) geforderte Form. Es gilt auch $w(ASSIGN(q',x,y)) \leq w(ASSIGN(q,x,y))$ und nach SE von $\equiv_{E \cup E1}$

$$ASSIGN(q,x,y) \equiv_{E \cup E1,a} ASSIGN(q',x,y)$$

und der letztgenannte Term hat ebenfalls die unter 2) geforderte Form mit $p_{m+1} := x$ und $r_{m+1} := y$.

iii.b) t is REMOVE(q,x) mit $q \in T_{\Sigma \cup \Sigma 1,a}$ und $x \in T_{\Sigma \cup \Sigma 1,n}$.

Es ist $w(q) \leq n$. Nach Induktionsannahme ii) gibt es ein $q' \in T_{\Sigma \cup \Sigma 1,a}$, sodaß $q \equiv_{E \cup E1,a} q'$, $w(q') \leq w(q)$ und es gibt $m \in \omega_0$, $p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma \cup \Sigma 1,n}$, sodaß q' is $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$.

iii.b.1) q' is NEWA, d.h. $m=0$.

Nach SE von $\equiv_{E \cup E1}$ und nach Gleichung E1.13 gilt

$$REMOVE(q,x) \equiv_{E \cup E1,a} REMOVE(NEWA,x) \equiv_{E \cup E1,a} NEWA.$$

Es gilt $w(NEWA)=1 \leq w(REMOVE(q,x))$, d.h. 1) und 2) gelten für dieses t.

iii.b.2) q' is $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$ und $m>0$.

Um die Gleichungen E1.14 bzw. E1.15 anwenden zu können, müssen wir zeigen, daß die Bedingungen derselben auswertbar sind.

Nach Lemma N.SP1 gibt es $j,k \in \omega_0$, sodaß

*31 $p_m \equiv_{E \cup E1,n} S^j(\emptyset)$, $x \equiv_{E \cup E1,n} S^k(\emptyset)$,

*32 $w(S^j(\emptyset)) \leq w(p_m)$, $w(S^k(\emptyset)) \leq w(x)$

und $S^j(\emptyset), S^k(\emptyset) \in T_{\Sigma \cup \Sigma 1,n} \cap T_{\Sigma,n}$. Damit gilt aber auch

$EQN(S^j(\emptyset),S^k(\emptyset))\in T_{\Sigma\cup\Sigma 1,d}\cap T_{\Sigma,d}$. Nach Lemma V.SP gibt es ein $tv\in\{tt,ff\}$, sodaß

$$EQN(S^j(\emptyset),S^k(\emptyset)) \equiv_{E,d} tv,$$

woraus nach Lemma 7 folgt

$$*33 \quad EQN(S^j(\emptyset),S^k(\emptyset)) \equiv_{E\cup E1,d} tv.$$

Nach SE von $\equiv_{E\cup E1}$ gilt zunächst mit *31

$$\begin{aligned} *34 \quad RE(q,x) &\equiv_{E\cup E1,a} RE(q,S^k(\emptyset)) \\ &\equiv_{E\cup E1,a} RE(AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m),S^k(\emptyset)) \\ &\equiv_{E\cup E1,a} RE(AS(..AS(NA,p_1,r_1),..,S^j(\emptyset),r_m),S^k(\emptyset)). \end{aligned}$$

Wegen *33 können wir Gleichung E1.14 bzw. E1.15 anwenden.

iii.b.2.1) tv <u>is</u> tt.

Nach Gleichung E1.14 gilt

$$\begin{aligned} *35 \quad & RE(AS(..AS(NA,p_1,r_1),..,S^j(\emptyset),r_m),S^k(\emptyset)) \equiv_{E\cup E1,a} \\ & RE(AS(..AS(NA,p_1,r_1),..,p_{m-1},r_{m-1}),S^k(\emptyset)). \end{aligned}$$

Es gilt wegen *32

$$\begin{aligned} w(RE(AS(\ldots,r_{m-1}),S^k(\emptyset)))&<w(RE(AS(\ldots,r_m),S^k(\emptyset))) \\ &\leq w(RE(q,x)) = n+1. \end{aligned}$$

Nach Induktionsannahme ii) gibt es daher ein $q''\in T_{\Sigma\cup\Sigma 1,a}$, sodaß

$$*36 \quad RE(AS(..AS(NA,p_1,r_1),..,p_{m-1},r_{m-1}),S^k(\emptyset)) \equiv_{E\cup E1,a} q'',$$

$$*37 \quad w(q'')\leq w(RE(AS(\ldots,r_{m-1}),S^k(\emptyset)))$$

und q" hat die unter 2) geforderte Form. Die Gültigkeit der Behauptung folgt jetzt für diesen Fall aus der Transitivität von $\equiv_{E\cup E1,a}$ zusammen mit *34, *35 und *36.

iii.b.2.2) tv <u>is</u> ff.

Nach Gleichung E1.15 gilt

$$\begin{aligned} *38 \quad & RE(AS(..AS(NA,p_1,r_1),..,S^j(\emptyset),r_m),S^k(\emptyset)) \equiv_{E\cup E1,a} \\ & AS(RE(AS(..AS(NA,p_1,r_1),..,p_{m-1},r_{m-1}),S^k(\emptyset)),S^j(\emptyset),r_m). \end{aligned}$$

Es gilt wegen *32

$$w(RE(AS(\ldots,r_{m-1}),S^k(\emptyset)))<w(RE(AS(\ldots,r_m),S^k(\emptyset)))$$

$\leq w(RE(q,x)) = n+1.$

Nach Induktionsannahme ii) gibt es daher ein $q'' \in T_{\Sigma\cup\Sigma 1,a}$, sodaß

*39 $RE(AS(\ldots AS(NA,p_1,r_1),\ldots,p_{m-1},r_{m-1}),S^k(\emptyset)) \equiv_{E\cup E1,a} q''$,

*40 $w(q'') \leq w(RE(AS(\ldots,r_{m-1}),S^k(\emptyset)))$

und q" hat die passende Form. Damit gilt auch

$w(AS(q'',S^j(\emptyset),r_m)) \leq w(AS(RE(AS(\ldots,r_{m-1}),S^k(\emptyset)),S^j(\emptyset),r_m))$

und nach SE von $\equiv_{E\cup E1}$

*41 $AS(RE(AS(\ldots,r_{m-1}),S^k(\emptyset)),S^j(\emptyset),r_m)) \equiv_{E\cup E1,a}$
$AS(q'',S^j(\emptyset),r_m)$,

wobei der letztgenannte Term wieder die unter 2) geforderte Form besitzt. Es gilt $w(AS(q'',S^j(\emptyset),r_m)) \leq w(RE(q,x))$ und der Rest ist Transitivität von $\equiv_{E\cup E1,a}$ zusammen mit *34, *38 und *41.

Damit ist die Induktion geschlossen und Lemma A.SP1 gezeigt. □

Für weitere Beweise ist es hilfreich zu wissen, daß es zu jedem <u>array</u>-Term, der mit REMOVE beginnt, einen i-kongruenten Term mit echt kleinerem Gewicht gibt.

<u>Korollar REM.SP1</u>

<u>Beh.:</u> $(\forall q \in T_{\Sigma\cup\Sigma 1,a})(\forall r \in T_{\Sigma\cup\Sigma 1,n})(\exists q' \in T_{\Sigma\cup\Sigma 1,a})$
$(\ REMOVE(q,r) \equiv_{E\cup E1,a} q' \ \wedge\ w(q') \leq w(q)\ )$

<u>Bew.:</u>

Durch Induktion über das Gewicht w(q) der Terme $q \in T_{\Sigma\cup\Sigma 1,a}$.

i) Induktionsanfang.

Sei $q \in T_{\Sigma\cup\Sigma 1,a}$ mit $w(q)=1$, d.h. q <u>is</u> NEWA. Sei $r \in T_{\Sigma\cup\Sigma 1,n}$.
Dann gilt nach Gleichung E1.13

$$REMOVE(NEWA,r) \equiv_{E\cup E1,a} NEWA.$$

ii) Induktionsannahme.

Sei $n\in\omega$ beliebig, aber fest.

Die Behauptung gelte für alle $q\in T_{\Sigma\cup\Sigma 1,a}$ mit $w(q)\leq n$.

iii) Induktionsschluß.

Sei $q\in T_{\Sigma\cup\Sigma 1,a}$ mit $w(q)=n+1$. Sei $r\in T_{\Sigma\cup\Sigma 1,n}$.
Nach Lemma A.SP1 gibt es ein $q'\in T_{\Sigma\cup\Sigma 1,a}$, sodaß $q \equiv_{E\cup E1,a} q'$ und $w(q')\leq w(q)$ und entweder q' <u>is</u> NEWA oder es gibt $m\in\omega_0$, $m>0$, $p_1,\ldots,p_m,r_1,\ldots,r_m\in T_{\Sigma\cup\Sigma 1,n}$, sodaß
q' <u>is</u> AS(...AS(NA,p_1,r_1),...,p_m,r_m).

iii.a) q' <u>is</u> NEWA.

Nach SE von $\equiv_{E\cup E1}$ und nach Gleichung E1.13 gilt

$$\text{REMOVE}(q,r) \equiv_{E\cup E1,a} \text{REMOVE}(\text{NEWA},r) \equiv_{E\cup E1,a} \text{NEWA}$$

und es ist $w(\text{NEWA})=1\leq w(q)$.

iii.b) q' <u>is</u> AS(...AS(NA,p_1,r_1),...,p_m,r_m) mit $m>0$.

Nach Lemma N.SP1 gibt es $j,k\in\omega_0$, sodaß

*42 $p_m \equiv_{E\cup E1,n} S^j(\emptyset)$, $r \equiv_{E\cup E1,n} S^k(\emptyset)$,

*43 $w(S^j(\emptyset))\leq w(p_m)$, $w(S^k(\emptyset))\leq w(r)$

und $S^j(\emptyset),S^k(\emptyset)\in T_{\Sigma\cup\Sigma 1,n}\cap T_{\Sigma,n}$. Damit ist aber auch $\text{EQN}(S^j(\emptyset),S^k(\emptyset))\in T_{\Sigma\cup\Sigma 1,d}\cap T_{\Sigma,d}$ und nach Lemma V.SP gibt es ein $tv\in\{tt,ff\}$, sodaß

$$\text{EQN}(S^j(\emptyset),S^k(\emptyset)) \equiv_{E,d} tv,$$

woraus nach Lemma 7

*44 $\text{EQN}(S^j(\emptyset),S^k(\emptyset)) \equiv_{E\cup E1,d} tv$ folgt.

Nach SE von $\equiv_{E\cup E1}$ folgt aus *42

*45 $\text{RE}(q,r) \equiv_{E\cup E1,a} \text{RE}(\text{AS}(..\text{AS}(\text{NA},p_1,r_1),..,p_m,r_m),r)$
$\equiv_{E\cup E1,a} \text{RE}(\text{AS}(..\text{AS}(\text{NA},p_1,r_1),..,S^j(\emptyset),r_m),S^k(\emptyset))$.

Wegen *44 können wir jetzt Gleichung E1.14 bzw. E1.15 anwenden.

iii.b.1) tv <u>is</u> tt.

Nach Gleichung E1.14 gilt

$$*46 \quad RE(AS(..AS(NA,p_1,r_1),..,S^j(\emptyset),r_m),S^k(\emptyset)) \equiv_{E\cup E1,a} RE(AS(..AS(NA,p_1,r_1),..,p_{m-1},r_{m-1}),S^k(\emptyset)).$$

Es ist $w(AS(...,r_{m-1}))<w(AS(...r_m))\leq n+1$. Nach Induktionsannahme ii) gibt es ein $q''\in T_{\Sigma\cup\Sigma 1,a}$, sodaß

$$*47 \quad RE(AS(..AS(NA,p_1,r_1),..,p_{m-1},r_{m-1}),S^k(\emptyset)) \equiv_{E\cup E1,a} q''$$

und $w(q'')\leq w(AS(...,r_{m-1}))<w(AS(...,r_m))\leq w(q)$. Der Rest ist Transitivität von $\equiv_{E\cup E1,a}$ zusammen mit *45 bis *47.

iii.b.2) tv <u>is</u> ff.

Nach Gleichung E1.15 gilt

$$*48 \quad RE(AS(..AS(NA,p_1,r_1),..,S^j(\emptyset),r_m),S^k(\emptyset)) \equiv_{E\cup E1,a} AS(RE(AS(..AS(NA,p_1,r_1),..,p_{m-1},r_{m-1}),S^k(\emptyset)),S^j(\emptyset),r_m).$$

Es ist $w(AS(...,r_{m-1}))<w(AS(...,r_m))\leq n+1$. Nach Induktionsannahme ii) gibt es daher ein $q''\in T_{\Sigma\cup\Sigma 1,a}$, sodaß

$$*49 \quad RE(AS(..AS(NA,p_1,r_1),..,p_{m-1},r_{m-1}),S^k(\emptyset)) \equiv_{E\cup E1,a} q'' \text{ und}$$

$$*50 \quad w(q'')\leq w(AS(...AS(NA,p_1,r_1),...,p_{m-1},r_{m-1})).$$

Nach SE von $\equiv_{E\cup E1}$ gilt mit *49 auch

$$*51 \quad AS(RE(AS(...,r_{m-1}),S^k(\emptyset)),S^j(\emptyset),r_m) \equiv_{E\cup E1,a} AS(q'',S^j(\emptyset),r_m)$$

und mit *50 auch

$$w(AS(q'',S^j(\emptyset),r_m))\leq w(AS(...AS(NA,p_1,r_1),...,S^j(\emptyset),r_m)) \leq w(AS(...AS(NA,p_1,r_1),...,p_m,r_m))\leq w(q).$$

Also ist $AS(q'',S^j(\emptyset),r_m)$ der gesuchte Term.

Damit ist die Induktion geschlossen. □

<u>Lemma V.SP1</u>

<u>Beh.:</u> $(\forall t\in T_{\Sigma\cup\Sigma 1,d})(\exists tv\in\{tt,ff\})(t \equiv_{E\cup E1,d} tv)$

<u>Bew.:</u>

Durch Induktion über das Gewicht w(t) der Terme $t\in T_{\Sigma\cup\Sigma 1,d}$.

i) Induktionsanfang,

ii) Induktionsannahme und die Fälle iii.a) bis iii.d) des Induktionsschlusses verlaufen völlig analog zu den Punkten i), ii) und iii.a) bis iii.d) des Beweises von Lemma V.SP0. Es ist lediglich $\Sigma 0$, E0 und N.SP0 durch $\Sigma 1$, E1 und N.SP1 zu ersetzen. Es bleiben die Fälle iii.e) und iii.f).

iii.e) t <u>is</u> EQA(t_1,t_2) mit $t_1,t_2 \in T_{\Sigma \cup \Sigma 1,a}$ und $w(t)=n+1$.

Nach Lemma A.SP1 gibt es $t_3,t_4 \in T_{\Sigma \cup \Sigma 1,a}$, sodaß

*52 $t_i \equiv_{E \cup E1,a} t_{i+2}$,

*53 $w(t_{i+2}) \le w(t_i)$

und entweder t_{i+2} <u>is</u> NEWA oder t_{i+2} <u>is</u> AS(q_i,p_i,r_i) mit $q_i \in T_{\Sigma \cup \Sigma 1,a}$ und $p_i,r_i \in T_{\Sigma \cup \Sigma 1,n}$ für i=1,2. Hierbei wurde die Aussage von Lemma A.SP1 etwas abgekürzt, da es i.f. nur darauf ankommt, daß t_3 und t_4 nicht mit REMOVE beginnen.

Nach SE von $\equiv_{E \cup E1}$ folgt aus *52

*54 EQA$(t_1,t_2) \equiv_{E \cup E1,d}$ EQA(t_3,t_4).

Wir unterscheiden jetzt nach dem Aufbau von t_3 und t_4.

Für die Fälle t_3 <u>is</u> NEWA , t_4 <u>is</u> NEWA ,

t_3 <u>is</u> NEWA , t_4 <u>is</u> AS(q_2,p_2,r_2) und

t_3 <u>is</u> AS(q_1,p_1,r_1) , t_4 <u>is</u> NEWA

ergibt sich die Gültigkeit der Behauptung aus den Gleichungen E1.1 bis E1.3 und zusammen mit *54 aus der Transitivität von $\equiv_{E \cup E1,d}$. Es bleibt der Fall

iii.e.4) t_3 <u>is</u> AS(q_1,p_1,r_1), t_4 <u>is</u> AS(q_2,p_2,r_2).

Zunächst zeigen wir, daß die Bedingungen der Gleichungen E1.4 bis E1.6 auswertbar sind. Man rechnet leicht nach, daß für i=1,2

$w(\text{INDEX}(\text{AS}(q_i,p_i,r_i),p_{3-i}))<n+1$ und

$w(\text{EQN}(\text{ACCESS}(\text{AS}(q_i,p_i,r_i),p_{3-i}),r_{3-i}))<n+1$ gilt.

Nach Induktionsannahme ii) gibt es also $tv_1,\dots,tv_4 \in \{tt,ff\}$, sodaß für i=1,2 gilt

$$\mathrm{INDEX}(\mathrm{AS}(q_i,p_i,r_i),p_{3-i}) \equiv_{E\cup E1,d} tv_i \quad \text{und}$$

$$\mathrm{EQN}(\mathrm{ACCESS}(\mathrm{AS}(q_i,p_i,r_i),p_{3-i}),r_{3-i}) \equiv_{E\cup E1,d} tv_{i+2}.$$

Nach SE von $\equiv_{E\cup E1}$ und nach Gleichung E.3 bzw. E.4 gilt dann, daß es $tv_5, tv_6 \in \{tt,ff\}$ gibt, sodaß

$\mathrm{IN}(\mathrm{AS}(q_1,p_1,r_1),p_2)$ AND $\mathrm{IN}(\mathrm{AS}(q_2,p_2,r_2),p_1) \equiv_{E\cup E1,d} tv_5$ und

$\mathrm{EQN}(\mathrm{AC}(\mathrm{AS}(q_1,p_1,r_1),p_2),r_2) \wedge \mathrm{EQN}(\mathrm{AC}(\mathrm{AS}(q_2,p_2,r_2),p_1),r_1) \equiv_{E\cup E1,d} tv_6.$

Jetzt gibt es drei Möglichkeiten.

iii.e.4.1) tv_5 is tt , tv_6 is tt.

Wir können Gleichung E1.4 anwenden und erhalten

*55 $\mathrm{EQA}(\mathrm{AS}(q_1,p_1,r_1),\mathrm{AS}(q_2,p_2,r_2)) \equiv_{E\cup E1,d}$

$\mathrm{EQA}(\mathrm{RE}(\mathrm{RE}(q_1,p_1),p_2),\mathrm{RE}(\mathrm{RE}(q_2,p_2),p_1)).$

Nach Korollar REM.SP1 gibt es $q_1', q_2' \in T_{\Sigma\cup\Sigma 1,a}$, sodaß für i=1,2

*56 $\mathrm{RE}(\mathrm{RE}(q_i,p_i),p_{3-i}) \equiv_{E\cup E1,a} q_i'$ und

$w(q_i') \le w(\mathrm{RE}(q_i,p_i))$ gilt.

Nach SE von $\equiv_{E\cup E1}$ folgt aus *56

*57 $\mathrm{EQA}(\mathrm{RE}(\mathrm{RE}(q_1,p_1),p_2),\mathrm{RE}(\mathrm{RE}(q_2,p_2),p_1)) \equiv_{E\cup E1,d} \mathrm{EQA}(q_1',q_2')$

und mit $w(q_i') \le w(\mathrm{RE}(q_i,p_i)) < w(\mathrm{AS}(q_i,p_i,r_i))$ für i=1,2 gilt auch $w(\mathrm{EQA}(q_1',q_2')) < w(\mathrm{EQA}(\mathrm{AS}(q_1,p_1,r_1),\mathrm{AS}(q_2,p_2,r_2))) = n+1$.

Nach Induktionsannahme ii) gibt es daher ein $tv \in \{tt,ff\}$, sodaß

*58 $\mathrm{EQA}(q_1',q_2') \equiv_{E\cup E1,d} tv.$

Die Gültigkeit der Behauptung folgt für diesen Unterfall dann aus der Transitivität von $\equiv_{E\cup E1,d}$ zusammen mit *54, *55, *57 und *58.

iii.e.4.2) tv_5 is tt , tv_6 is ff.

Wir können Gleichung E1.5 anwenden und erhalten

*59 $\mathrm{EQA}(\mathrm{AS}(q_1,p_1,r_1),\mathrm{AS}(q_2,p_2,r_2)) \equiv_{E\cup E1,d}$ ff.

iii.e.4.3) tv_5 is ff.

Wir können Gleichung E1.6 anwenden und erhalten

*60 $EQA(AS(q_1,p_1,r_1),AS(q_2,p_2,r_2)) \equiv_{E\cup E1,d}$ ff.

Die Gültigkeit der Behauptung folgt für iii.e.4.2) bzw. iii.e.4.3) aus der Transitivität von $\equiv_{E\cup E1,d}$ zusammen mit *54 und *59 bzw. *60.

iii.f) t <u>is</u> INDEX(q,r) mit $q\epsilon T_{\Sigma\cup\Sigma1,a}$ und $r\epsilon T_{\Sigma\cup\Sigma1,n}$.

Nach Lemma A.SP1 gibt es ein $q'\epsilon T_{\Sigma\cup\Sigma1,a}$, sodaß $q \equiv_{E\cup E1,a} q'$, $w(q')\leq w(q)$ und entweder q' <u>is</u> NEWA oder q' <u>is</u> AS(q",p",r") mit $q"\epsilon T_{\Sigma\cup\Sigma1,a}$ und $p",r"\epsilon T_{\Sigma\cup\Sigma1,n}$. Wie unter iii.e) kommt es nur darauf an, daß q' nicht mit REMOVE beginnt.

Wir unterscheiden nach dam Aufbau von q'.

iii.f.1) q' <u>is</u> NEWA.

Nach SE von $\equiv_{E\cup E1}$ und nach Gleichung E1.7 gilt

$$INDEX(q,r) \equiv_{E\cup E1,d} INDEX(NEWA,r) \equiv_{E\cup E1,d} ff.$$

iii.f.2) q' <u>is</u> AS(q",p",r").

Nach SE von $\equiv_{E\cup E1}$ gilt zunächst

*61 $INDEX(q,r) \equiv_{E\cup E1,d} INDEX(AS(q",p",r"),r)$.

Es ist $w(p")+w(r)<n$ und daher $w(EQN(p",r))\leq n$. Nach Induktionsannahme ii) gibt es daher ein $tv\epsilon\{tt,ff\}$, sodaß $EQN(p",r) \equiv_{E\cup E1,d} tv$. Dann gibt es zwei Möglichkeiten.

iii.f.2.1) tv <u>is</u> tt.

Wir können Gleichung E1.8 anwenden und erhalten

*62 $INDEX(AS(q",p",r"),r) \equiv_{E\cup E1,d}$ tt.

Der Rest ist Transitivität von $\equiv_{E\cup E1,d}$ zusammen mit *61 und *62.

iii.f.2.2) tv <u>is</u> ff.

Wir können Gleichung E1.9 anwenden und erhalten

*63 $INDEX(AS(q",p",r"),r) \equiv_{E\cup E1,d} INDEX(q",r)$

und es gilt $w(IN(q",r))\leq n$. Nach Induktionsannahme ii) gibt es ein $tv' \in \{tt,ff\}$, sodaß

*64 $\quad INDEX(q'',r) \equiv_{E \cup E1,d} tv'$.

Der Rest ist Transitivität von $\equiv_{E \cup E1,d}$ zusammen mit *61, *63 und *64.

Damit ist die Induktion geschlossen und es ist gezeigt, daß SP1 vollständig ist. □

Mit dem bisher Gezeigten können wir leicht Aussagen über die t-Vollständigkeit von SP0 bzw. SP1 auf SP machen.

Lemma TV.SP0

Beh.: $(\forall s \in \{\underline{dis},\underline{nat}\})(\forall t \in T_{\Sigma \cup \Sigma 0,s})(\exists t' \in T_{\Sigma,s})(t \sim_{E \cup E0,s} t')$

Bew.:

Durch Fallunterscheidung über die Sorten $s \in S$.

a) $s=\underline{dis}$.

Da SP0 und SP vollständig sind, ist SP0 nach Korollar 4 i-vollständig auf SP bzgl. dis und nach Korollar 6 daher auch t-vollständig auf SP bzgl. dis.

b) $s=\underline{nat}$.

Da $T_{\Sigma,n}$ ja gerade die Menge $\{S^k(\emptyset) \mid k \in \omega_0\}$ ist, haben wir mit Lemma N.SP0 gezeigt, daß

$(\forall t \in T_{\Sigma \cup \Sigma 0,n})(\exists t' \in T_{\Sigma,n})(t \equiv_{E \cup E0,n} t')$ gilt,

d.h. daß SP0 i-vollständig auf SP bzgl. nat ist. Nach Korollar 6 ist SP0 daher auch t-vollständig auf SP bzgl. nat. □

Lemma TV.SP1

Beh.: $(\forall s \in \{\underline{dis},\underline{nat}\})(\forall t \in T_{\Sigma \cup \Sigma 1,s})(\exists t' \in T_{\Sigma,s})(t \sim_{E \cup E1,s} t')$

Bew.:

Der Beweis verläuft völlig analog zum Beweis von Lemma TV.SP0. □

Jetzt wird gezeigt, daß auch in I1 jeder nat- und array-Term i-kongruent zu einem kanonischen nat- bzw. array-Term ist und daß I1 vollständig ist.

Lemma N.I1

Beh.: $(\forall t \in T_{\Sigma I1,n})(\exists t' \in T_{\Sigma I1,n})(\exists j \in \omega_0)$

$$(\quad 1)\quad t \equiv_{EI1,n} t' \quad \wedge \quad w(t') \leq w(t) \quad \wedge$$

$$2)\quad t' \ \underline{is}\ S^j(\emptyset) \quad)$$

Bew.:

Nach Definition von I1 gilt $T_{\Sigma I1,a} = T_{\Sigma \cup \Sigma 1,a}$ und $T_{\Sigma I1,n} = T_{\Sigma \cup \Sigma 1,n}$. Lemma N.SP1 liefert damit

$$(\forall t \in T_{\Sigma I1,n})(\exists t' \in T_{\Sigma I1,n})(\exists j \in \omega_0)$$

$$(\quad 1')\quad t \equiv_{E \cup E1,n} t' \quad \wedge \quad w(t') \leq w(t) \quad \wedge$$

$$2')\quad t' \ \underline{is}\ S^j(\emptyset) \quad),$$

woraus nach Lemma 7, also wegen $t \equiv_{E \cup E1,n} t' \Rightarrow t \equiv_{EI1,n} t'$, die Gültigkeit der Behauptung folgt. □

Lemma A.I1

Beh.: $(\forall t \in T_{\Sigma I1,a})(\exists t' \in T_{\Sigma I1,a})(\exists m \in \omega_0)(\exists p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma I1,n})$

$$(\quad 1)\quad t \equiv_{EI1,a} t' \quad \wedge \quad w(t') \leq w(t) \quad \wedge$$

$$2)\quad t' \ \underline{is}\ \mathrm{ASSIGN}(\ldots\mathrm{ASSIGN}(\mathrm{NEWA},p_1,r_1),\ldots,p_m,r_m) \quad)$$

Bew.:

Die Gültigkeit der Behauptung folgt wegen $T_{\Sigma I1,a} = T_{\Sigma \cup \Sigma 1,a}$ und $T_{\Sigma I1,n} = T_{\Sigma \cup \Sigma 1,n}$ aus Lemma A.SP1 und Lemma 7. □

Lemma V.I1

Beh.: $(\forall t \in T_{\Sigma I1,d})(\exists tv \in \{tt,ff\})(t \equiv_{EI1,d} tv)$

Bew.:

Durch Induktion über das Gewicht w(t) der Terme $t \in T_{\Sigma I1,d}$.

i) Induktionsanfang,

ii) Induktionsannahme und die Punkte iii.a) bis iii.c) verlaufen völlig analog zu den Punkten i), ii) und iii.a) bis iii.c) des Beweises von Lemma V.SP.

iii.d) t <u>is</u> $EQN(t_1,t_2)$ mit $t_1,t_2 \in T_{\Sigma I1,n}$.

Wegen $T_{\Sigma I1,n} = T_{\Sigma\cup\Sigma 1,n}$ ist $EQN(t_1,t_2) \in T_{\Sigma I1,d} \cap T_{\Sigma\cup\Sigma 1,d}$. Die Gültigkeit der Behauptung folgt daher aus Lemma V.SP1 und aus Lemma 7.

iii.e) t <u>is</u> $EQA(t_1,t_2)$ mit $t_1,t_2 \in T_{\Sigma I1,a}$.

Dieser Fall verläuft analog zu iii.d).

iii.f) t <u>is</u> $INDEX(q,r)$ mit $q \in T_{\Sigma I1,a}$ und $r \in T_{\Sigma I1,n}$.

Wegen der Gleichheit der <u>nat</u>- bzw. <u>array</u>-Träger von $T_{\Sigma I1}$ und $T_{\Sigma\cup\Sigma 1}$ ist bereits $INDEX(q,r) \in T_{\Sigma I1,d} \cap T_{\Sigma\cup\Sigma 1,d}$.

Nach Lemma V.SP1 gibt es daher ein $tv \in \{tt,ff\}$, sodaß

$INDEX(q,r) \equiv_{E\cup E1,d} tv$ gilt, woraus nach Lemma 7

$INDEX(q,r) \equiv_{EI1,d} tv$ folgt.

iii.g) t <u>is</u> $EQS(ST(q_1,r_1),ST(q_2,r_2))$ mit $q_1,q_2 \in T_{\Sigma I1,a}$ und $r_1,r_2 \in T_{\Sigma I1,n}$.

Nach Lemma N.I1 gibt es $j,k \in \omega_0$, sodaß

*65 $r_1 \equiv_{EI1,n} S^j(\emptyset)$, $r_2 \equiv_{EI1,n} S^k(\emptyset)$,

$w(S^j(\emptyset)) \leq w(r_1)$, $w(S^k(\emptyset)) \leq w(r_2)$.

Nach SE von $\equiv_{EI1}$ folgt aus *65

*66 $EQS(ST(q_1,r_1),ST(q_2,r_2)) \equiv_{EI1,d}$
$EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset)))$.

Wir unterscheiden nach j und k.

iii.g.1) $S^j(\emptyset)$ <u>is</u> $\emptyset$, $S^k(\emptyset)$ <u>is</u> $\emptyset$, d.h. $j=k=0$.

Dann gilt nach Gleichung E0D.1

*67 $EQS(ST(q_1,\emptyset),ST(q_2,\emptyset)) \equiv_{EI1,d} tt$.

iii.g.2) $S^j(\emptyset)$ <u>is</u> $\emptyset$, $S^k(\emptyset)$ <u>~~is~~</u> $\emptyset$, d.h. $j=0 \wedge k>0$.

Nach Gleichung E0D.2 gilt

*68 $EQS(ST(q_1,\emptyset),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d} ff$.

iii.g.3) $S^j(\emptyset)$ $\underline{\cancel{is}}$ $\emptyset$, $S^k(\emptyset)$ $\underline{is}$ $\emptyset$, d.h. $j>0 \wedge k=0$.

Dann gilt nach Gleichung E0D.3

$*69$ $EQS(ST(q_1,S^j(\emptyset)),ST(q_2,\emptyset)) \equiv_{EI1,d}$ ff.

Für die Fälle iii.g.1) bis iii.g.3) folgt die Gültigkeit der Behauptung aus der Transitivität von $\equiv_{EI1,d}$ zusammen mit $*66$ und $*67$ bis $*69$.

iii.g.4) $S^j(\emptyset)$ $\underline{\cancel{is}}$ $\emptyset$, $S^k(\emptyset)$ $\underline{\cancel{is}}$ $\emptyset$, d.h. $j,k>0$.

Die Bedingungen von E0D.4 bis E0D.6 sind auswertbar, denn es gilt

$p_1:=EQN(AC(q_1,S^j(\emptyset)),AC(q_2,S^k(\emptyset)))\in T_{\Sigma I1,d} \cap T_{\Sigma\cup\Sigma 1,d}$ und

$p_2:=IN(q_1,S^j(\emptyset)),p_3:=IN(q_2,S^k(\emptyset))\in T_{\Sigma I1,d} \cap T_{\Sigma\cup\Sigma 1,d}$.

Nach Lemma V.SP1 und Lemma 7 gibt es daher

$tv_1,tv_2,tv_3\in\{tt,ff\}$, sodaß für $i=1,2,3$ $p_i \equiv_{EI1,d} tv_i$.

Dann gibt es auch ein $tv_4\in\{tt,ff\}$, sodaß wir nach SE von $\equiv_{EI1}$ und durch Anwendung der Gleichung E.5 bzw. E.6

$$p_2 \text{ EQV } p_3 \equiv_{EI1,d} tv_4$$

erhalten. Jetzt bleiben drei Möglichkeiten.

iii.g.4.1) tv_4 $\underline{is}$ tt , tv_1 $\underline{is}$ tt.

Wir können Gleichung E0D.4 anwenden und erhalten

$*70$ $EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d}$

$EQS(ST(q_1,S^{j-1}(\emptyset)),ST(q_2,S^{k-1}(\emptyset)))$.

Es gilt

$w(EQS(q_1,S^{j-1}(\emptyset)),ST(q_2,S^{k-1}(\emptyset))))< w(EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset))))$

$\leq w(EQS(ST(q_1,r_1),ST(q_2,r_2))) = n+1.$

Nach Induktionsannahme ii) gibt es also ein $tv\in\{tt,ff\}$, sodaß

$*71$ $EQS(ST(q_1,S^{j-1}(\emptyset)),ST(q_2,S^{k-1}(\emptyset))) \equiv_{EI1,d} tv$.

Der Rest ist Transitivität von $\equiv_{EI1,d}$ zusammen mit $*66$, $*70$ und $*71$.

iii.g.4.2) tv_4 <u>is</u> tt , tv_1 <u>is</u> ff.

Wir können Gleichung EOD.5 anwenden und erhalten

*72 $EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d}$ ff.

iii.g.4.3) tv_4 <u>is</u> ff.

Wir können Gleichung EOD.6 anwenden und erhalten

*73 $EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d}$ ff.

Für die letzten beiden Unterfälle folgt die Gültigkeit der Behauptung aus der Transitivität von $\equiv_{EI1,d}$ zusammen mit *66 und *72 bzw. *73.

Damit ist die Induktion geschlossen und es ist gezeigt, daß I1 vollständig ist. □

Die erste Implementierungsebene I1 erfüllt die syntaktischen Kriterien (i) und (ii) von Lemma 16 und ist daher t-vollständig auf SP1.

<u>Lemma TV.I1</u>

<u>Beh.:</u> $(\forall s \in \{\underline{dis},\underline{nat},\underline{array}\})(\forall t \in T_{\Sigma I1,s})(\exists t' \in T_{\Sigma \cup \Sigma 1,s})(t \sim_{EI1,s} t')$

<u>Bew.:</u>

Die Gültigkeit der Behauptung folgt direkt aus Lemma 16. □

Auch in I2 ist jeder <u>nat</u>-Term i-kongruent zu einem kanonischen <u>nat</u>-Term.

<u>Lemma N.I2</u>

<u>Beh.:</u> $(\forall t \in T_{\Sigma I2,n})(\exists t' \in T_{\Sigma I2,n})(\exists j \in \omega_0)$

$(\quad 1)\ t \equiv_{EI2,n} t' \quad \wedge \quad w(t') \leq w(t) \quad \wedge$

$\quad 2)\ t'\ \underline{is}\ S^j(\emptyset) \quad)$

<u>Bew.:</u>

Durch Induktion über das Gewicht w(t) der Terme $t \in T_{\Sigma I2,n}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma I2,n}$ mit $w(t)=1$, d.h. t is $\emptyset$.

Die Gültigkeit der Behauptung folgt dann aus den Reflexivitätseigenschaften der Relationen $\equiv_{EI2,n}$ und $\leq$.

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t \in T_{\Sigma I2,n}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma I2,n}$ mit $w(t)=n+1$.

Es gilt $w(t) \geq 2$ und daher auch t ~~is~~ $\emptyset$. Wir unterscheiden drei Fälle.

iii.a) t is $S(t')$ mit $t' \in T_{\Sigma I2,n}$.

Dieser Fall verläuft analog zum Fall iii.a) aus dem Beweis von Lemma N.SP1.

iii.b) t is ACCESS(q,r) mit $q \in T_{\Sigma I2,a}$ und $r \in T_{\Sigma I2,n}$.

Wir wollen zeigen, daß es zu t einen $\equiv_{EI2}$-kongruenten nat-Term aus I1 gibt, auf den wir dann die Lemmata N.I1 und 7 anwenden können. Wir zeigen daher zunächst

*74 Für alle $q' \in T_{\Sigma I2,a}$ mit $w(q') \leq w(q)$ gilt

$$(\exists q'' \in T_{\Sigma I2,a})(\exists m \in \omega_0)(\exists p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n})$$
$$(\;3)\; q' \equiv_{EI2,a} q'' \;\wedge\; w(q'') \leq w(q') \;\wedge$$
$$4)\; q'' \text{ is } AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)\;).$$

Die gegenüber der ähnlichen Aussage im Lemma A.SP1 etwas veränderte Formulierung, nämlich die Forderung, daß die p_i und r_i bereits aus $T_{\Sigma I1}$ sein sollen, erfüllt einen doppelten Zweck. Zum einen bleibt uns unter iii.b.iii.b) eine weitere Induktion erspart und zum anderen wird sichergestellt, daß unter iii.c.2.2) die Bedingungen der Gleichungen E1.11 und E1.12 auswertbar sind.

Der Beweis erfolgt durch Induktion über das Gewicht $w(q')$ der Terme $q' \in T_{\Sigma I2,a}$ mit $w(q') \leq w(q)$.

iii.b.i) Induktionsanfang.

Sei $q' \in T_{\Sigma I2,a}$ mit $w(q')=1$.

Daraus folgt, daß q' is NEWA. Es ist $m=0$ und der Rest ist Reflexivität von $\equiv_{EI2,a}$ und $\leq$.

iii.b.ii) Induktionsannahme.

Sei $k \in \omega$ fest mit $k<w(q)$.

Die Behauptung gelte für alle $q' \in T_{\Sigma I2,a}$ mit $w(q') \leq k$.

iii.b.iii) Induktionsschluß.

Sei $q' \in T_{\Sigma I2,a}$ mit $w(q')=k+1$.

Aus $w(q') \geq 2$ folgt q' ~~is~~ NEWA.

Wir unterscheiden nach dem Aufbau von q' zwei Fälle.

iii.b.iii.a) q' is ASSIGN(q_1,p_1,r_1) mit $q_1 \in T_{\Sigma I2,a}$, $p_1,r_1 \in T_{\Sigma I2,n}$.

Es gilt $w(q_1) \leq k$. Nach Induktionsannahme iii.b.ii) gibt es ein $q_2 \in T_{\Sigma I2,a}$, sodaß

*75 $\quad q_1 \equiv_{EI2,a} q_2 \quad$ und $\quad w(q_2) \leq w(q_1)$

und q_2 hat die unter 4) geforderte Form.

Es gilt $w(p_1), w(r_1) \leq k \leq n$ und nach Induktionsannahme ii) gibt es $p_2,r_2 \in T_{\Sigma I2,n}$, sodaß

*76 $\quad p_1 \equiv_{EI2,n} p_2 \quad , \quad r_1 \equiv_{EI2,n} r_2 \quad ,$

$\quad\quad w(p_2) \leq w(p_1) \quad , \quad w(r_2) \leq w(r_1)$

und p_2 und r_2 haben die unter 2) geforderte Form, d.h. es gilt $p_2,r_2 \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$.

Mit *75 und *76 gilt nach SE von $\equiv_{EI2}$ auch

$\quad$ ASSIGN$(q_1,p_1,r_1) \equiv_{EI2,a}$ ASSIGN(q_2,p_2,r_2),

$\quad$ $w(AS(q_2,p_2,r_2)) \leq w(AS(q_1,p_1,r_1))$ und

$AS(q_2,p_2,r_2)$ hat die unter 4) geforderte Form.

iii.b.iii.b) q' <u>is</u> RE(q",r") mit $q"\epsilon T_{\Sigma I2,a}$ und $r"\epsilon T_{\Sigma I2,n}$.

Es ist $w(q")\leq k$ und nach Induktionsannahme iii.b.ii) gibt es ein $q"'\epsilon T_{\Sigma I2,a}$, ein $m\epsilon\omega_0$ und $p_1,\ldots,p_m,r_1,\ldots,r_m\epsilon T_{\Sigma I2,n}\cap T_{\Sigma I1,n}$, sodaß

*77 $q" \equiv_{EI2,a} q"'$,

*78 $w(q"')\leq w(q")$ und

*79 q"' <u>is</u> $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)\epsilon T_{\Sigma I2,a}\cap T_{\Sigma I1,a}$.

Es ist $w(r")\leq k\leq n$ und nach Induktionsannahme ii) gibt es ein $r'\epsilon T_{\Sigma I2,n}$ und ein $j\epsilon\omega_0$, sodaß

*80 $r" \equiv_{EI2,n} r'$,

*81 $w(r')\leq w(r")$ und

*82 r' <u>is</u> $S^j(\emptyset)\ \epsilon\ T_{\Sigma I2,n}\cap T_{\Sigma I1,n}$.

Nach SE von $\equiv_{EI2}$ gilt mit *77 und *80 auch

*83 $RE(q",r") \equiv_{EI2,a} RE(q"',r')$.

Mit *78 und *81 gilt

*84 $w(RE(q"',r'))\leq w(RE(q",r"))$

und wegen *79 und *82 gilt $RE(q"',r')\epsilon T_{\Sigma I2,a}\cap T_{\Sigma I1,a}$.

Nach Lemma A.I1 gibt es daher ein $q^*\epsilon T_{\Sigma I1,a}$, $l\epsilon\omega_0$ und $p_1^*,\ldots,p_l^*,r_1^*,\ldots,r_l^*\epsilon T_{\Sigma I1,n}$, sodaß

*85 $RE(q"',r') \equiv_{EI1,a} q^*$,

*86 $w(q^*)\leq w(RE(q"',r'))$ und

*87 q^* <u>is</u> $AS(\ldots AS(NA,p_1^*,r_1^*),\ldots,p_l^*,r_l^*)$.

Nach Lemma 7 folgt aus *85

*88 $RE(q"',r') \equiv_{EI2,a} q^*$.

q^* hat die unter 4) geforderte Form und der Rest ist Transitivität von $\equiv_{EI2,a}$ zusammen mit *83 und *88 und von $\leq$ zusammen mit *84 und *86.

Damit ist die Induktion unter iii.b) geschlossen.

Wir wissen jetzt nach *74, daß es zu dem Subterm q von ACCESS(q,r) ein $q'\epsilon T_{\Sigma I2,a}\cap T_{\Sigma I1,a}$ gibt, sodaß

$q \equiv_{EI2,a} q'$ und $w(q') \leq w(q)$.

Wegen $w(r) \leq n$ gibt es nach Induktionsannahme ii) ein

$r' \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$, sodaß

$r \equiv_{EI2,n} r'$ und $w(r') \leq w(r)$.

Daher gilt nach SE von $\equiv_{EI2}$ auch

*89 $ACCESS(q,r) \equiv_{EI2,n} ACCESS(q',r')$

und es ist $ACCESS(q',r') \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$.

Nach Lemma N.I1 gibt es daher ein $r'' \in T_{\Sigma I1,n}$, sodaß

*90 $ACCESS(q',r') \equiv_{EI1,n} r''$,

$w(r'') \leq w(ACCESS(q',r'))$ und

r'' hat die unter 2) geforderte Form. Nach Lemma 7 folgt aus *90

*91 $ACCESS(q',r') \equiv_{EI2,n} r''$.

Es gilt $w(r'') \leq w(ACCESS(q,r))$ und der Rest ist Transitivität von $\equiv_{EI2,n}$ mit *89 und *91.

iii.c) t <u>is</u> TOP(p) mit $p \in T_{\Sigma I2,s}$.

Um die Gleichungen EΣ0.11 bzw. EΣ0.12 anwenden zu können, müssen wir zunächst die folgende Hilfsbehauptung zeigen, deren zusätzliche Forderung 6.2) im Unterfall iii.c.2.2) benötigt wird:

*92 Für alle $p' \in T_{\Sigma I2,s}$ mit $w(p') \leq w(p)$ gilt:

$(\exists q \in T_{\Sigma I2,a} \cap T_{\Sigma I1,a})(\exists r \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n})(\exists m \in \omega_0)$

$(\exists p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n})$

(5) $p' \equiv_{EI2,s} STAK(q,r) \wedge w(r) \leq w(p') \wedge$

6.1) q <u>is</u> $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m) \wedge$

6.2) $(\forall i \in \{1,\ldots,m\})(\ w(r_i) \leq w(p')\ )\quad)$.

Der Beweis erfolgt durch Induktion über das Gewicht $w(p')$ der Terme $p' \in T_{\Sigma I2,s}$ mit $w(p') \leq w(p)$.

iii.c.i) Induktionsanfang.

Sei $p' \in T_{\Sigma I2,s}$ mit $w(p')=1$, d.h. p' <u>is</u> NEWST.

Nach Gleichung $E\Sigma 0.7$ gilt

$$\text{NEWST} \equiv_{EI2,s} \text{STAK}(\text{NEWA},\emptyset),$$

wobei STAK(NEWA,$\emptyset$) die Forderungen 5), 6.1) und 6.2) erfüllt.

iii.c.ii) Induktionsannahme.

Sei $k \in \omega$ fest mit $k<w(p)$.

Die Behauptung gelte für alle $p' \in T_{\Sigma I2,s}$ mit $w(p') \leq k$.

iii.c.iii) Induktionsschluß.

Sei $p' \in T_{\Sigma I2,s}$ mit $w(p')=k+1$.

Es gilt $w(p') \geq 2$ und daher p' ~~is~~ NEWST.

Es bleiben drei Fälle zu betrachten.

iii.c.iii.a) p' <u>is</u> POP(p") mit $p'' \in T_{\Sigma I2,s}$.

Es gilt $w(p'') \leq k$. Nach Induktionsannahme iii.c.ii) gibt es Terme $q \in T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ und $r \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$, sodaß

*93 $\quad p'' \equiv_{EI2,s} \text{STAK}(q,r)$,

$\quad w(r) \leq w(p'') \leq k \leq n$ und

q erfüllt die Forderungen 6.1) und 6.2). Nach Induktionsannahme ii) gibt es nun ein $r' \in T_{\Sigma I2,n}$, sodaß

*94 $\quad r \equiv_{EI2,n} r'$, $w(r') \leq w(r)$ und

entweder r' <u>is</u> $\emptyset$ oder es gibt ein $j \in \omega_0$, $j>0$, sodaß r' <u>is</u> $S^j(\emptyset)$. Danach unterscheiden wir zwei Fälle.

iii.c.iii.a.1) r' <u>is</u> $\emptyset$.

Nach SE von $\equiv_{EI2}$ folgt aus *93 und *94

$$\begin{aligned} \text{POP}(p'') &\equiv_{EI2,s} \text{POP}(\text{STAK}(q,r)) \\ &\equiv_{EI2,s} \text{POP}(\text{STAK}(q,\emptyset)) \\ &\equiv_{EI2,s} \text{STAK}(q,\emptyset), \end{aligned}$$

wobei die letzte i-Kongruenz nach Gleichung $E\Sigma 0.9$ gilt.

q und $\emptyset$ erfüllen die Forderungen 5) und 6.1) und 6.2).

iii.c.iii.a.2) r' <u>is</u> $S^j(\emptyset)$ mit $j\epsilon\omega_0$, $j>0$.

Nach SE von $\equiv_{EI2}$, *93, *94 und Gleichung EΣ0.10 gilt

$$POP(p") \equiv_{EI2,s} POP(STAK(q,S^j(\emptyset)))$$
$$\equiv_{EI2,s} STAK(q,S^{j-1}(\emptyset)).$$

Die Terme $S^{j-1}(\emptyset)$ und q erfüllen die Forderungen 5) bis 6.2).

iii.c.iii.b) p' <u>is</u> PUSH(p",r) mit $p"\epsilon T_{\Sigma I2,s}$ und $r\epsilon T_{\Sigma I2,n}$.

Es ist $w(p")\leq k$. Daher gibt es nach Induktionsannahme iii.c.ii) Terme $q'\epsilon T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ und $r'\epsilon T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$, sodaß

*95 $\quad p" \equiv_{EI2,s} STAK(q',r')$

und q' und r' erfüllen auch die restlichen Forderungen unter 5), 6.1) und 6.2).

Es gilt $w(r)<w(p')\leq n+1$. Nach Induktionsannahme ii) gibt es ein $r"\epsilon T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$, sodaß

*96 $\quad r \equiv_{EI2,n} r"$ und $w(r")\leq w(r)$.

Nach SE von $\equiv_{EI2}$ folgt aus *95 und *96

$$PUSH(p",r) \equiv_{EI2,s} PUSH(ST(q',r'),r")$$
$$\equiv_{EI2,s} ST(AS(q',S(r'),r"),S(r')),$$

wobei die letzte i-Kongruenz nach Gleichung EΣ0.8 gilt.

Wegen $S(r'),r"\epsilon T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$ und $q'\epsilon T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ gilt auch $AS(q',S(r'),r")\epsilon T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ und mit q' hat auch AS(q',S(r'),r") die unter 6.1) geforderte Form.

Es gilt $w(r')\leq w(p")$ und daher auch $w(S(r'))\leq w(PUSH(p",r))$, d.h. S(r') erfüllt 5).

Nach *96 gilt auch $w(r")\leq w(PUSH(p",r))$ und für q' <u>is</u> $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$ gilt nach Voraussetzung $(\forall i\epsilon\{1,\ldots,m\})(w(r_i)\leq w(p"))$. Daher gilt auch $(\forall i\epsilon\{1,\ldots,m,m+1\})(w(r_i)\leq w(PUSH(p",r)))$, wobei $r_{m+1}:=r"$.

Also erfüllen die Subterme $r_1,\ldots,r_{m+1}$ von $AS(q',S(r'),r_{m+1})$ auch 6.2).

iii.c.iii.c) p' is $ST(q,r)$ mit $q \in T_{\Sigma I2,a}$ und $r \in T_{\Sigma I2,n}$.
Wir können analog zu iii.b) zeigen, daß

*97 Für alle $q^* \in T_{\Sigma I2,a}$ mit $w(q^*) \leq w(q)$ gilt

$$(\exists q' \in T_{\Sigma I2,a})(\exists m \in \omega_0)(\exists p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n})$$

$$(\ 7)\quad q^* \equiv_{EI2,a} q' \ \wedge\ w(q') \leq w(q^*)$$

$$8)\quad q' \text{ is } AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)\ )$$

gilt; daher wird der Beweis hier unterlassen.

Es ist $w(q) \leq w(q)$. Sei also $q' \in T_{\Sigma I2,a}$ der nach *97 existierende array-Term mit $q \equiv_{EI2,a} q'$, $w(q') \leq w(q)$ und der unter 8) geforderten Form. Es ist $w(r) < w(p') = k+1 \leq n+1$. Nach Induktionsannahme ii) gibt es daher ein $j \in \omega_0$, sodaß

$$r \equiv_{EI2,n} S^j(\emptyset) \quad \wedge \quad w(S^j(\emptyset)) \leq w(r)$$

und es ist $S^j(\emptyset) \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$. Nach SE von $\equiv_{EI2}$ gilt

$$ST(q,r) \equiv_{EI2,s} ST(q',S^j(\emptyset)).$$

Es ist $w(S^j(\emptyset)) \leq w(r) \leq w(ST(q,r))$, d.h. $S^j(\emptyset)$ erfüllt 5).
Für $m=0$, d.h. q' is NEWA, sind 6.1) und 6.2) automatisch erfüllt. Für $m>0$, d.h. q' is $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$, gilt

$$(\forall i \in \{1,\ldots,m\})(w(r_i) \leq w(q') \leq w(q) \leq w(ST(q,r)));$$

also ist auch für diesen Fall 6.1) und 6.2) erfüllt.

Damit ist die Induktion unter iii.c) geschlossen.

Wir wissen jetzt, daß es zu dem Subterm p von TOP(p) Terme $q \in T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ und $r \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$ gibt, sodaß

$$p \equiv_{EI2,s} ST(q,r),$$

woraus nach SE von $\equiv_{EI2}$ folgt

*98 $TOP(p) \equiv_{EI2,n} TOP(ST(q,r))$.

Es ist $w(r) \leq w(p) \leq n$. Nach Induktionsannahme ii) ist entweder $r \equiv_{EI2,n} \emptyset$ oder es gibt ein $j \in \omega_0$, $j>0$, sodaß

$r \equiv_{EI2,n} S^j(\emptyset)$ und $w(S^j(\emptyset)) \leq w(r)$.

iii.c.1) $r \equiv_{EI2,n} \emptyset$.

Nach SE von $\equiv_{EI2}$ und Gleichung EΣ0.11 gilt

*99 $TOP(ST(q,r)) \equiv_{EI2,n} TOP(ST(q,\emptyset)) \equiv_{EI2,n} \emptyset$.

Es ist $w(\emptyset) \leq w(TOP(p))$ und der Rest ist Transitivität von $\equiv_{EI2,n}$ zusammen mit *98 und *99.

iii.c.2) $r \equiv_{EI2,n} S^j(\emptyset)$.

Nach SE von $\equiv_{EI2}$ und Gleichung EΣ0.12 gilt

*100 $TOP(ST(q,r)) \equiv_{EI2,n} TOP(ST(q,S^j(\emptyset)))$
$\equiv_{EI2,n} ACCESS(q,S^j(\emptyset))$.

Jetzt gibt es folgendes Problem: Wir müssen zu $ACCESS(q,S^j(\emptyset))$ einen i-kongruenten nat-Term finden, dessen Gewicht kleiner oder gleich $w(TOP(p))$ ist. Dies ist deshalb problematisch, weil nicht immer $w(ST(q,S^j(\emptyset))) \leq w(p)$ gilt und wir daher nicht den nach Lemma N.I1 und Lemma 7 existierenden, zu $ACCESS(q,S^j(\emptyset))$ i-kongruenten Term r' nehmen können, für den $w(r') \leq w(ACCESS(q,S^j(\emptyset))) < w(TOP(ST(q,S^j(\emptyset))))$
$\leq w(TOP(ST(q,r)))$ gilt.

Wir unterscheiden zunächst nach dem Aufbau des Terms q, der ja die Forderungen 5), 6.1) und 6.2) erfüllt.

iii.c.2.1) q is NEWA, d.h. m=0.

Es gilt nach Gleichung E1.10

*101 $ACCESS(NEWA,S^j(\emptyset)) \equiv_{EI2,n} \emptyset$.

$\emptyset$ hat die unter 2) geforderte Form, es ist $w(\emptyset) \leq w(TOP(p))$ und wegen der Transitivität von $\equiv_{EI2,n}$ zusammen mit *98, *100 und *101 ist auch der Rest von 1) erfüllt.

iii.c.2.2) q is $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$ mit $m \in \omega_0$, $m>0$ und $p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$ und $(\forall i \in \{1,\ldots,m\})(w(r_i) \leq w(p))$.

Wir zeigen die folgende Hilfsaussage.

*102 Für alle $q' \epsilon T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ mit
q' is $AS(\ldots,AS(NA,p'_1,r'_1),\ldots,p'_k,r'_k)$, $k\epsilon\omega$ und
$p'_1,\ldots,p'_k,r'_1,\ldots,r'_k \epsilon T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$ gilt:
$(\forall r' \epsilon T_{\Sigma I2,n} \cap T_{\Sigma I1,n})$
(9) $ACCESS(q',r') \equiv_{EI2,n} \emptyset$ $\oplus$
10) $(\exists l \epsilon \{1,\ldots,k\})(ACCESS(q',r') \equiv_{EI2,n} r'_l)$).

Wir zeigen also, daß die Anwendung von ACCESS auf einen array in kanonischer Form entweder die $\emptyset$ liefert – falls an der zugegriffenen Stelle kein Eintrag definiert ist – oder genau den Eintrag r'_l an der zugegriffenen Stelle p'_l liefert. Zusammen mit 6.2) heißt das, daß wir in jedem Fall einen zu $ACCESS(q,S^j(\emptyset))$ i-kongruenten nat-Term finden, dessen Gewicht kleiner oder gleich $w(TOP(p))$ ist.

Der Beweis erfolgt durch Induktion über $k\epsilon\omega$.

iii.c.2.2.i) Induktionsanfang.

Sei k=1, d.h. q' is $AS(NA,p'_1,r'_1)$ mit $p'_1,r'_1 \epsilon T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$.
Sei $r' \epsilon T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$. Dann ist $EQN(p'_1,r') \epsilon T_{\Sigma I2,d} \cap T_{\Sigma I1,d}$ und nach Lemma V.I1 und Lemma 7 gibt es ein $tv \epsilon \{tt,ff\}$, sodaß
$$EQN(p'_1,r') \equiv_{EI2,d} tv.$$

iii.c.2.2.i.a) tv is tt.

Wir können Gleichung E1.11 anwenden und erhalten
$$ACCESS(AS(NA,p'_1,r'_1),r') \equiv_{EI2,n} r'_1,$$
d.h. 10) ist erfüllt mit l=1.

iii.c.2.2.i.b) tv is ff.

Wir können Gleichung E1.12 anwenden und erhalten
$$ACCESS(AS(NA,p'_1,r'_1),r') \equiv_{EI2,n} AC(NA,r') \equiv_{EI2,n} \emptyset,$$
wobei die letzte Kongruenz nach Gleichung E1.10 gilt.

iii.c.2.2.ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $k\in\omega$.

iii.c.2.2.iii) Induktionsschluß.

Sei $q' \in T_{\sum I2,a} \cap T_{\sum I1,a}$

q' is $AS(\ldots AS(NA,p_1',r_1'),\ldots,p_{k+1}',r_{k+1}')$ und

$p_1',\ldots,p_k',p_{k+1}',r_1',\ldots,r_k',r_{k+1}' \in T_{\sum I2,n} \cap T_{\sum I1,n}$.

Sei $r' \in T_{\sum I2,n} \cap T_{\sum I1,n}$.

Dann ist $EQN(p_{k+1}',r') \in T_{\sum I2,d} \cap T_{\sum I1,d}$ und nach Lemma V.I1 und Lemma 7 gibt es ein $tv \in \{tt,ff\}$, sodaß

$$EQN(p_{k+1}',r') \equiv_{EI2,d} tv,$$

d.h. die Bedingungen von Gleichung E1.11 und E1.12 sind auswertbar. Wir unterscheiden daher zwei Fälle.

iii.c.2.2.iii.a) tv is tt.

Wir können Gleichung E1.11 anwenden und erhalten

$$AC(AS(\ldots AS(NA,p_1',r_1'),\ldots,p_{k+1}',r_{k+1}'),r') \equiv_{EI2,n} r_{k+1}',$$

d.h. 10) ist erfüllt mit $l=k+1$.

iii.c.2.2.iii.b) tv is ff.

Wir können Gleichung E1.12 anwenden und erhalten

$$AC(AS(\ldots AS(NA,p_1',r_1'),\ldots,p_{k+1}',r_{k+1}'),r') \equiv_{EI2,n}$$
$$AC(AS(\ldots AS(NA,p_1',r_1'),\ldots,p_k',r_k'),r').$$

Der letztgenannte Term erfüllt wegen Induktionsannahme iii.c.2.2.ii) die Forderung 9) bzw. 10) und wegen der Transitivität von $\equiv_{EI2,n}$ erfüllt $AC(AS(\ldots,r_{k+1}'),r')$ sie auch.

Damit ist die Induktion unter iii.c.2.2) geschlossen.

Wegen *102 bleiben für unseren Term $AS(\ldots,p_m,r_m)$ zwei Fälle zu untersuchen.

iii.c.2.2.1) Für den Term gilt 9), d.h. es gilt

*103 $\quad AC(AS(..AS(NA,p_1,r_1),\ldots,p_m,r_m),S^j(\emptyset)) \equiv_{EI2,n} \emptyset.$

Es ist $w(\emptyset) \leq w(TOP(p))$, $\emptyset$ hat die unter 2) geforderte Form und wegen der Transitivität von $\equiv_{EI2,n}$ zusammen mit *98, *100 und *103 ist auch der Rest von 1) erfüllt.

iii.c.2.2.2) Für den Term gilt 10), d.h. es gilt

$(\exists l \in \{1,\ldots,m\})$

(*104 $AC(AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m),S^j(\emptyset)) \equiv_{EI2,n} r_l)$.

Wegen der Transitivität von $\equiv_{EI2,n}$ zusammen mit *98, *100 und *104 gilt also

*105 $TOP(p) \equiv_{EI2,n} r_l$.

Nach Voraussetzung des Unterfalls iii.c.2.2) gilt

*106 $w(r_l) \leq w(p) \leq n \leq w(TOP(p))$,

denn der Term q <u>is</u> $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$ erfüllt die Forderung 6.2).

Nach Induktionsannahme ii) gibt es daher ein $k \in \omega_0$, sodaß

*107 $r_l \equiv_{EI2,n} S^k(\emptyset)$ und

*108 $w(S^k(\emptyset)) \leq w(r_l)$.

Die Gültigkeit der Behauptung folgt jetzt aus der Transitivität von $\equiv_{EI2,n}$ bzw. $\leq$ zusammen mit *105 und *107 bzw. *106 und *108.

Damit ist die Induktion geschlossen und Lemma N.I2 gezeigt. $\square$

<u>Lemma A.I2</u>

<u>Beh.:</u> $(\forall t \in T_{\Sigma I2,a})(\exists t' \in T_{\Sigma I2,a})(\exists m \in \omega_0)$

$(\exists p_1,\ldots,p_m,r_1,\ldots,r_m \in T_{\Sigma I2,n})$

(1) $t \equiv_{EI2,a} t' \quad \wedge \quad w(t') \leq w(t) \quad \wedge$

2) t' <u>is</u> $AS(\ldots AS(NA,p_1,r_1),\ldots,p_m,r_m)$)

<u>Bew.:</u>

Der Beweis verläuft völlig analog zum Beweis von Lemma A.SP1 und seine Durchführung wird daher hier unterlassen. $\square$

Jetzt zeigen wir noch, daß jeder stack-Term in I2 i-kongruent zu einem pointer/array-Paar ST(q,r) ist.

Lemma S.I2

Beh.: $(\forall t \in T_{\Sigma I2,s})(\exists q \in T_{\Sigma I2,a})(\exists r \in T_{\Sigma I2,n})(t \equiv_{EI2,s} ST(q,r))$

Bew.:

Durch Induktion über das Gewicht w(t) der Terme $t \in T_{\Sigma I2,s}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma I2,s}$ mit $w(t)=1$.

Es ist t is NEWST und nach Gleichung EΣ0.7 gilt

$$NEWST \equiv_{EI2,s} ST(NEWA,\emptyset).$$

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.

Die Behauptung gelte für alle $t \in T_{\Sigma I2,s}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma I2,s}$ mit $w(t)=n+1$.

Da t wegen $w(t) \geq 2$ nicht identisch NEWST sein kann, bleiben drei zu untersuchende Fälle übrig.

iii.a) t is PUSH(p,r') mit $p \in T_{\Sigma I2,s}$ und $r' \in T_{\Sigma I2,n}$.

Es gilt $w(p) \leq n$ und nach Induktionsannahme ii) gibt es daher ein $q \in T_{\Sigma I2,a}$ und ein $r \in T_{\Sigma I2,n}$, sodaß

$$p \equiv_{EI2,s} ST(q,r).$$

Nach SE von $\equiv_{EI2}$ und nach Gleichung EΣ0.8 folgt daraus

$$\begin{aligned} PUSH(p,r') &\equiv_{EI2,s} PUSH(ST(q,r),r') \\ &\equiv_{EI2,s} ST(AS(q,S(r),r'),S(r)). \end{aligned}$$

iii.b) t is POP(p) mit $p \in T_{\Sigma I2,s}$.

Es ist $w(p) \leq n$ und nach Induktionsannahme ii) gibt es daher ein $q \in T_{\Sigma I2,a}$ und ein $r \in T_{\Sigma I2,n}$, sodaß

*109 $\quad p \equiv_{EI2,s} ST(q,r).$

Nach Lemma N.I2 gibt es ein $r' \in T_{\Sigma I2,n}$ und ein $j \in \omega_0$, sodaß

*110 $r \equiv_{EI2,n} r'$

und $w(r') \leq w(r)$ und r' __is__ $S^j(\emptyset)$. Wir unterscheiden nach $j \in \omega_0$ zwei Fälle.

iii.b.1) r' __is__ $\emptyset$, d.h. $j=0$.

Nach SE von $\equiv_{EI2}$ folgt aus *109 und *110

$$POP(p) \equiv_{EI2,s} POP(ST(q,\emptyset)) \equiv_{EI2,s} ST(q,\emptyset),$$

wobei die letzte Kongruenz nach Gleichung EΣ0.9 gilt.

iii.b.2) r' __is__ $S^j(\emptyset)$ ~~is~~ $\emptyset$, d.h. $j>0$.

Nach SE von $\equiv_{EI2}$ folgt aus *109 und *110

$$POP(p) \equiv_{EI2,s} POP(ST(q,S^j(\emptyset))) \equiv_{EI2,s} ST(q,S^{j-1}(\emptyset)),$$

wobei die letzte Kongruenz nach Gleichung EΣ0.10 gilt.

iii.c) t __is__ $ST(q,r)$ mit $q \in T_{\Sigma I2,a}$ und $r \in T_{\Sigma I2,n}$.

Die Gültigkeit der Behauptung folgt für diesen Fall aus der Reflexivität von $\equiv_{EI2,s}$.

Damit ist die Induktion geschlossen. $\square$

__Lemma V.I2__

__Beh.:__ $(\forall t \in T_{\Sigma I2,d})(\exists tv \in \{tt,ff\})(t \equiv_{EI2,d} tv)$

__Bew.:__

Durch Induktion über das Gewicht $w(t)$ der Terme $t \in T_{\Sigma I2,d}$.

i) Induktionsanfang,

ii) Induktionsannahme und die Schritte

iii.a) bis iii.c) des Induktionsschlusses verlaufen völlig analog zu den Punkten i), ii) und iii.a) bis iii.c) des Beweises von Lemma V.SP. Sei also $t \in T_{\Sigma I2,d}$ mit $w(t)=n+1$. Dann bleiben folgende vier Fälle übrig.

iii.d) t is $EQN(t_1,t_2)$ mit $t_1,t_2 \in T_{\Sigma I2,n}$.

Dieser Fall verläuft völlig analog zum Fall iii.d) des Beweises von Lemma V.SP0.

iii.e) t is $EQA(t_1,t_2)$ mit $t_1,t_2 \in T_{\Sigma I2,a}$.

Nach Lemma A.I2 gibt es $q_1,q_2 \in T_{\Sigma I2,a}$ und $m_1,m_2 \in \omega_0$ und $p^1_1,\ldots,p^1_{m_1},r^1_1,\ldots,r^1_{m_1},p^2_1,\ldots,p^2_{m_2},r^2_1,\ldots,r^2_{m_2} \in T_{\Sigma I2,n}$, sodaß

*111 $t_i \equiv_{EI2,a} q_i$ und

q_i is $AS(\ldots AS(NA,p^i_1,r^i_1),\ldots,p^i_{m_i},r^i_{m_i})$ für $i=1,2$.

Nach Lemma N.I2 gibt es zu den p^i_j und r^i_j für $i=1,2$ und $j=1,\ldots,m_i$ $\equiv_{EI2}$-kongruente kanonische nat-Terme, die ja sowohl in $T_{\Sigma I2,n}$ als auch in $T_{\Sigma I1,n}$ enthalten sind. Ersetzt man die kanonischen nat-Terme für die p^i_j und r^i_j in den q_i, dann erhält man array-Terme q'_1,q'_2, die auch schon in $T_{\Sigma I1,a}$ enthalten sind. Also folgt zusammen mit der SE von $\equiv_{EI2}$ aus Lemma N.I2, daß es $q'_1,q'_2 \in T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ gibt, sodaß

*112 $q_i \equiv_{EI2,a} q'_i$ für $i=1,2$.

Transitivität von $\equiv_{EI2,a}$ und SE von $\equiv_{EI2}$ liefern mit *111 und *112

*113 $EQA(t_1,t_2) \equiv_{EI2,d} EQA(q'_1,q'_2)$,

wobei $EQA(q'_1,q'_2) \in T_{\Sigma I2,d} \cap T_{\Sigma I1,d}$. Nach Lemma V.I1 und Lemma 7 gibt es daher ein $tv \in \{tt,ff\}$, sodaß

*114 $EQA(q'_1,q'_2) \equiv_{EI2,d} tv$.

Der Rest ist Transitivität von $\equiv_{EI2,d}$ zusammen mit *113 und *114.

iii.f) t is $INDEX(q,r)$ mit $q \in T_{\Sigma I2,a}$ und $r \in T_{\Sigma I2,n}$.

Nach Lemma A.I2 und Lemma N.I2 gibt es, aus den gleichen Gründen wie unter iii.e), ein $q' \in T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ und ein $r' \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$, sodaß

$q \equiv_{EI2,a} q'$ und $r \equiv_{EI2,n} r'$ bzw.

*115 $INDEX(q,r) \equiv_{EI2,d} INDEX(q',r')$.

Es ist $INDEX(q',r') \in T_{\Sigma I2,d} \cap T_{\Sigma I1,d}$ und nach Lemma V.I1 und Lemma 7 gibt es ein $tv \in \{tt,ff\}$, sodaß

$*116 \quad INDEX(q',r') \equiv_{EI2,d} tv.$

Der Rest ist Transitivität von $\equiv_{EI2,d}$ mit $*115$ und $*116$.

iii.g) t <u>is</u> $EQS(t_1,t_2)$ mit $t_1,t_2 \in T_{\Sigma I2,s}$.

Nach Lemma S.I2, A.I2 und N.I2 gibt es Terme

$q_1,q_2 \in T_{\Sigma I2,a} \cap T_{\Sigma I1,a}$ und $r_1,r_2 \in T_{\Sigma I2,n} \cap T_{\Sigma I1,n}$, sodaß

$$t_i \equiv_{EI2,s} ST(q_i,r_i)$$

für $i=1,2$. Nach SE von $\equiv_{EI2}$ gilt auch

$*117 \quad EQS(t_1,t_2) \equiv_{EI2,d} EQS(ST(q_1,r_1),ST(q_2,r_2)),$

wobei der letztgenannte Term aus $T_{\Sigma I2,d} \cap T_{\Sigma I1,d}$ ist. Lemma V.I1 und 7 sichern die Existenz eines $tv \in \{tt,ff\}$ zu, sodaß

$*118 \quad EQS(ST(q_1,r_1),ST(q_2,r_2)) \equiv_{EI2,d} tv.$

Der Rest ist Transitivität von $\equiv_{EI2,d}$ mit $*117$ und $*118$.

Damit ist die Induktion geschlossen. Es sei erwähnt, daß die Induktionsannahme unter iii.a) bis iii.c) benutzt wurde. □

<u>Lemma TV.I2</u>

<u>Beh.:</u> $(\forall s \in \{\underline{dis},\underline{nat},\underline{stack},\underline{array}\})$

$$(\forall t \in T_{\Sigma I2,s})(\exists t' \in T_{\Sigma I1,s})(t \sim_{EI2,s} t')$$

<u>Bew.:</u>

Durch Fallunterscheidung über die Sorten $s \in S \cup S0 \cup S1$.

a) s=<u>dis</u>.

Die Gültigkeit der Behauptung folgt für diesen Fall aus der Vollständigkeit von I2 und aus den Korollaren 4 und 6.

b) s=<u>nat</u>.

Folgt aus Lemma N.I2 und Korollar 6.

c) s=<u>stack</u>.

Folgt aus Lemma S.I2, A.I2 und N.I2 und aus Korollar 6.

d) s=<u>array</u>.

Folgt aus Lemma A.I2, Lemma N.I2 und aus Korollar 6.

Damit ist gezeigt, daß I2 t-vollständig auf I1 ist. $\square$

Zur Vorbereitung der t-Konsistenznachweise werden drei Aussagen über die Operation EQN jeweils für einige der fünf beteiligten t-Spezifikationen SP,...,I2 gemacht. Es wird, anschaulich gesagt, gezeigt, daß mit EQN ein Gleichheitsprädikat für die <u>nat</u>-Terme dieser Spezifikationen spezifiziert wird.
Zuerst wird gezeigt, daß jeder kanonische <u>nat</u>-Term unter EQN zu keinem anderen kanonischen <u>nat</u>-Term außer sich selbst gleich ist.

<u>Lemma K.NAT.SP</u>

<u>Beh.:</u> 1) $(\forall j\epsilon\omega_0)(EQN(S^j(\emptyset),S^j(\emptyset)) \equiv_{E,d} tt)$

2) $(\forall j,k\epsilon\omega_0)(j\neq k \Rightarrow EQN(S^j(\emptyset),S^k(\emptyset)) \equiv_{E,d} ff)$

<u>Bew.:</u>

ad 1) Durch Induktion über $j\epsilon\omega_0$.

1.i) Induktionsanfang.

Sei $j\epsilon\omega_0$, $j=0$.

Dann gilt $S^j(\emptyset)$ <u>is</u> $\emptyset$ und nach Gleichung E.7 gilt

$$EQN(\emptyset,\emptyset) \equiv_{E,d} tt.$$

1.ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $j\epsilon\omega_0$.

1.iii) Induktionsschluß.

Es gilt $j+1>0$ und daher $S^{j+1}(\emptyset)$ $\cancel{\text{is}}$ $\emptyset$.

Nach Gleichung E.10 und Induktionsannahme 1.ii) gilt

$$EQN(S^{j+1}(\emptyset),S^{j+1}(\emptyset)) \equiv_{E,d} EQN(S^j(\emptyset),S^j(\emptyset)) \equiv_{E,d} tt.$$

Damit ist die Induktion unter ad 1) geschlossen.

ad 2) Durch Induktion über $j \in \omega_0$.

2.i) Induktionsanfang.

Sei $j \in \omega_0$, $j=0$.

Dann gilt $S^j(\emptyset)$ is $\emptyset$. Sei $k \in \omega_0$ mit $j \neq k$. Es gilt $k>0$ und daher $S^k(\emptyset)$ ~~is~~ $\emptyset$. Nach Gleichung E.8 folgt daraus

$$EQN(\emptyset, S^k(\emptyset)) \equiv_{E,d} ff.$$

2.ii) Induktionsannahme.

Die Behauptung gelte für ein $j \in \omega_0$ und alle $k \in \omega_0$.

2.iii) Induktionsschluß.

Sei $k \in \omega_0$, $j+1 \neq k$.

Es gilt $S^{j+1}(\emptyset)$ ~~is~~ $\emptyset$ und wir unterscheiden nach k.

2.iii.a) $k=0$.

D.h. $S^k(\emptyset)$ is $\emptyset$. Nach Gleichung E.9 gilt

$$EQN(S^{j+1}(\emptyset), \emptyset) \equiv_{E,d} ff.$$

2.iii.b) $k>0$.

D.h. $S^k(\emptyset)$ ~~is~~ $\emptyset$. Dann gilt

$$EQN(S^{j+1}(\emptyset), S^k(\emptyset)) \equiv_{E,d} EQN(S^j(\emptyset), S^{k-1}(\emptyset)) \equiv_{E,d} ff$$

nach Gleichung E.10 und Induktionsannahme 2.ii).

Damit ist die Induktion unter ad 2) geschlossen. □

Lemma K.NAT.SP0

Beh.: 1) $(\forall j \in \omega_0)(EQN(S^j(\emptyset), S^j(\emptyset)) \equiv_{E \cup E0,d} tt)$

2) $(\forall j,k \in \omega_0)(j \neq k \Rightarrow EQN(S^j(\emptyset), S^k(\emptyset)) \equiv_{E \cup E0,d} ff)$

Bew.:

Die Gültigkeit der Behauptung folgt aus Lemma K.NAT.SP und Lemma 7. □

Lemma K.NAT.SP1, K.NAT.I1 und K.NAT.I2 sollen die gleichen

Aussagen wie Lemma K.NAT.SP für die Relationen $\equiv_{E \cup E1}$, $\equiv_{EI1}$ und $\equiv_{EI2}$ enthalten. Die Beweise verlaufen völlig analog zu dem Beweis von Lemma K.NAT.SP0.

Korollar 7

Beh.: $\sim_{E,n} = \{(t,t) \mid t \in T_{\Sigma,n}\}$

Bew.:

Alle Terme $t \in T_{\Sigma,n}$ lassen sich darstellen als t is $S^j(\emptyset)$ mit $j \in \omega_0$. $S^j(\emptyset), S^k(\emptyset) \in T_{\Sigma,n}$ sind genau dann verschieden, wenn $j,k \in \omega_0$ mit $j \neq k$.

Wir zeigen, daß es zu zwei Termen $S^j(\emptyset), S^k(\emptyset) \in T_{\Sigma,n}$ mit $j,k \in \omega_0$ und $j \neq k$ immer einen Kontext $ct \in C_\Sigma(n,d)$ gibt, sodaß

$$ct[S^j(\emptyset)/x_n^1] \not\equiv_{E,d} ct[S^k(\emptyset)/x_n^1].$$

Seien also $S^j(\emptyset), S^k(\emptyset) \in T_{\Sigma,n}$ mit $j,k \in \omega_0$ und $j \neq k$.

Dann gilt nach Lemma K.NAT.SP

$$EQN(S^j(\emptyset), S^j(\emptyset)) \equiv_{E,n} tt \quad \text{und}$$

$$EQN(S^j(\emptyset), S^k(\emptyset)) \equiv_{E,n} ff.$$

Der gesuchte Kontext ist $EQN(S^j(\emptyset), x_n^1) \in C_\Sigma(n,d)$ und es gilt

$$EQN(S^j(\emptyset), x_n^1)[S^j(\emptyset)/x_n^1] \not\equiv_{E,d} EQN(S^j(\emptyset), x_n^1)[S^k(\emptyset)/x_n^1],$$

da SP nach Voraussetzung konsistent ist. Es gilt also

$$(\forall t_1, t_2 \in T_{\Sigma,n})(t_1 \ \cancel{\text{is}}\ t_2 \Rightarrow (t_1,t_2) \notin \sim_{E,n}).$$

Da $\sim_{E,n}$ reflexiv ist, gilt

$$(\forall t \in T_{\Sigma,n})((t,t) \in \sim_{E,n}), \text{ q.e.d. } \square$$

Korollar 8

Beh.: SP ist kategorisch.

Bew.:

Es gilt $\equiv_{E,d} = \sim_{E,d}$ per definitionem.

Da E keine Axiome der Sorte nat enthält, $\equiv_{E,n}$ aber reflexiv ist, gilt $\equiv_{E,n} = \{(t,t) \mid t \in T_{\Sigma,n}\}$ und nach Korollar 7 also $\equiv_{E,n} = \sim_{E,n}$. Wir haben gezeigt: $(\forall s \in S)(\equiv_{E,s} = \sim_{E,s})$, q.e.d.. $\square$

Korollar 9

Beh.: $(\forall t_1,t_2 \in T_{\Sigma,n})(t_1 \equiv_{E,n} t_2 \iff EQN(t_1,t_2) \equiv_{E,d} tt)$

Bew.:

Die Gültigkeit der Behauptung folgt aus Korollar 7, Korollar 8 und Lemma K.NAT.SP sowie aus der Konsistenz von SP. $\square$

Korollar 10

Beh.: $(\forall t_1,t_2 \in T_{\Sigma\cup\Sigma 0,n})(t_1 \equiv_{E\cup E0,n} t_2 \iff EQN(t_1,t_2) \equiv_{E\cup E0,d} tt)$

"=>" Seien $t_1,t_2 \in T_{\Sigma\cup\Sigma 0,n}$ mit $t_1 \equiv_{E\cup E0,n} t_2$.

Dann gilt nach SE von $\equiv_{E\cup E0}$

*119 $EQN(t_1,t_2) \equiv_{E\cup E0,d} EQN(t_1,t_1)$.

Nach Lemma N.SP0 gibt es ein $j \in \omega_0$, sodaß $t_1 \equiv_{E\cup E0,n} S^j(\emptyset)$

und nach SE von $\equiv_{E\cup E0}$ und Lemma K.NAT.SP0 folgt daraus

*120 $EQN(t_1,t_1) \equiv_{E\cup E0,d} EQN(S^j(\emptyset),S^j(\emptyset)) \equiv_{E\cup E0,d} tt$.

Der Rest ist Transitivität von $\equiv_{E\cup E0,d}$ mit *119 und *120.

"<=" Seien $t_1,t_2 \in T_{\Sigma\cup\Sigma 0,n}$ und es gelte $EQN(t_1,t_2) \equiv_{E\cup E0,d} tt$.

Nach Lemma N.SP0 gibt es $j,k \in \omega_0$, sodaß

*121 $t_1 \equiv_{E\cup E0,n} S^j(\emptyset)$ und $t_2 \equiv_{E\cup E0,n} S^k(\emptyset)$.

Nach SE von $\equiv_{E\cup E0}$ gilt

$$EQN(S^j(\emptyset),S^k(\emptyset)) \equiv_{E\cup E0,d} tt.$$

Wäre $j \neq k$, dann gälte nach Lemma K.NAT.SP0

$$tt \equiv_{E\cup E0,d} EQN(S^j(\emptyset),S^k(\emptyset)) \equiv_{E\cup E0,d} ff,$$

was der Konsistenz von SP0 widerspricht.

Also muß $j=k$ sein und daher gilt mit *121 auch

$$t_1 \equiv_{E\cup E0,n} S^j(\emptyset) \ \underline{is}\ S^k(\emptyset) \equiv_{E\cup E0,n} t_2,$$ q.e.d.. $\square$

Korollar 11

Beh.: $(\forall t_1,t_2 \in T_{\Sigma\cup\Sigma 1,n})(t_1 \equiv_{E\cup E1,n} t_2 \iff EQN(t_1,t_2) \equiv_{E\cup E1,d} tt)$ $\square$

Korollar 12

Beh.: $(\forall t_1,t_2 \in T_{\Sigma I1,n})(t_1 \equiv_{EI1,n} t_2 \iff EQN(t_1,t_2) \equiv_{EI1,d} tt)$ $\square$

Korollar 13

Beh.: $(\forall t_1,t_2 \in T_{\Sigma I2,n})(t_1 \equiv_{EI2,n} t_2 \iff EQN(t_1,t_2) \equiv_{EI2,d} tt)$ □

Die Beweise der Korollare 11, 12 und 13 verlaufen völlig analog zu dem Beweis von Korollar 10.

Korollar EQ.NAT.SP

Beh.: $(\forall t_1,t_2,t_3 \in T_{\Sigma,n})$

(1) $EQN(t_1,t_1) \equiv_{E,d} tt \wedge$

2) $EQN(t_1,t_2) \equiv_{E,d} EQN(t_2,t_1) \wedge$

3) $(EQN(t_1,t_2) \equiv_{E,d} tt \wedge EQN(t_2,t_3) \equiv_{E,d} tt) \Rightarrow$

$EQN(t_1,t_3) \equiv_{E,d} tt$)

Bew.:

ad 1) Die Gültigkeit der Behauptung 1) folgt aus der Reflexivität von $\equiv_{E,n}$ und aus Korollar 9.

ad 2) Seien $t_1,t_2 \in T_{\Sigma,n}$.

2.a) Es gelte $EQN(t_1,t_2) \equiv_{E,d} tt$.
Nach Korollar 9 gilt dann $t_1 \equiv_{E,n} t_2$ und die SE von $\equiv_E$ und Punkt 1) dieses Korollars liefern

$$EQN(t_2,t_1) \equiv_{E,d} EQN(t_1,t_1) \equiv_{E,d} tt \equiv_{E,d} EQN(t_1,t_2).$$

2.b) Es gelte $EQN(t_1,t_2) \equiv_{E,d} ff$.
Wegen der Konsistenz von SP und nach Korollar 9 folgt daraus $t_1 \not\equiv_{E,n} t_2$. Also gibt es $j,k \in \omega_0$, $j \neq k$, sodaß

$$t_1 \equiv_{E,n} S^j(\emptyset) \quad \text{und} \quad t_2 \equiv_{E,n} S^k(\emptyset).$$

Nach Korollar K.NAT.SP gilt

$$EQN(S^k(\emptyset),S^j(\emptyset)) \equiv_{E,d} ff$$

und daraus folgt nach SE von $\equiv_E$ und nach Voraussetzung

$$EQN(t_2,t_1) \equiv_{E,d} ff \equiv_{E,d} EQN(t_1,t_2).$$

2.c) Seien $t_1,t_2,t_3 \in T_{\Sigma,n}$. Es gelte

$EQN(t_1,t_2) \equiv_{E,d} tt$ und $EQN(t_2,t_3) \equiv_{E,d} tt$.

Nach Korollar 9 gilt

$$t_1 \equiv_{E,n} t_2 \quad \text{und} \quad t_2 \equiv_{E,n} t_3,$$

woraus nach SE von $\equiv_E$ und Punkt 1) dieses Korollars

$$EQN(t_1,t_3) \equiv_{E,d} EQN(t_2,t_2) \equiv_{E,d} tt \qquad \text{folgt.} \ \Box$$

<u>Korollar EQ.NAT.I1</u>

<u>Beh.:</u> $(\forall t_1,t_2,t_3 \in T_{\Sigma I1,n})$

(1) $EQN(t_1,t_1) \equiv_{EI1,d} tt \wedge$

2) $EQN(t_1,t_2) \equiv_{EI1,d} EQN(t_2,t_1) \wedge$

3) $(EQN(t_1,t_2) \equiv_{EI1,d} tt \wedge EQN(t_2,t_3) \equiv_{EI1,d} tt) \Rightarrow$

$EQN(t_1,t_3) \equiv_{EI1,d} tt$)

<u>Bew.:</u>

ad 1) Die Gültigkeit der Behauptung 1) folgt aus der Reflexivität von $\equiv_{EI1,n}$ und aus Korollar 12.

ad 2) Seien $t_1,t_2 \in T_{\Sigma I1,n}$.

Nach Lemma N.I1 gibt es $j_1,j_2 \in \omega_0$, sodaß $t_i \equiv_{EI1,n} S^{j_i}(\emptyset)$ für i=1,2. Nach SE von $\equiv_{EI1}$ gilt

*122 $EQN(t_1,t_2) \equiv_{EI1,d} EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset))$.

Es ist $EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset)) \in T_{\Sigma I1,d} \cap T_{\Sigma,d}$ und nach Punkt 2) von Korollar EQ.NAT.SP gilt

$$EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset)) \equiv_{E,d} EQN(S^{j_2}(\emptyset),S^{j_1}(\emptyset)),$$

woraus nach Lemma 7 folgt

*123 $EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset)) \equiv_{EI1,d} EQN(S^{j_2}(\emptyset),S^{j_1}(\emptyset))$.

Nach SE von $\equiv_{EI1}$ gilt

*124 $EQN(S^{j_2}(\emptyset),S^{j_1}(\emptyset)) \equiv_{EI1,d} EQN(t_2,t_1)$.

Der Rest ist Transitivität von $\equiv_{EI1,d}$ zusammen mit *122, *123 und *124.

ad 3) Die Gültigkeit von Punkt 3) der Behauptung folgt analog zum Beweis von 3) des Korollars EQ.NAT.SP aus Korollar 12, SE von $\equiv_{EI1}$ und aus Punkt 1) von EQ.NAT.I1. $\square$

Korollar EQ.NAT.I2

Beh.: $(\forall t_1,t_2,t_3 \in T_{\Sigma I2,n})$

$$(\; 1) \; EQN(t_1,t_1) \equiv_{EI2,d} tt \; \wedge$$

$$2) \; EQN(t_1,t_2) \equiv_{EI2,d} EQN(t_2,t_1) \; \wedge$$

$$3) \; (EQN(t_1,t_2) \equiv_{EI2,d} tt \wedge EQN(t_2,t_3) \equiv_{EI2,d} tt) \Rightarrow$$

$$EQN(t_1,t_3) \equiv_{EI2,d} tt \;)$$

Bew.:

Völlig analog zum Beweis von Korollar EQ.NAT.I1. $\square$

Wir zeigen jetzt, daß SP0 und SP1 t-konsistent auf SP sind und daß sie auch t-Erweiterungen von SP sind. Danach bereiten wir die restlichen t-Konsistenznachweise weiter vor.

Lemma TK.SP0

Beh.: $(\forall s \in \{\underline{dis},\underline{nat}\})(\forall t_1,t_2 \in T_{\Sigma,s})(t_1 \sim_{E,s} t_2 \;\Rightarrow\; t_1 \sim_{E \cup E0,s} t_2)$

Bew.:

Durch Fallunterscheidung über die Sorten $s \in S=\{\underline{dis},\underline{nat}\}$.

a) SP0 ist t-konsistent auf SP bzgl. dis nach Korollar 5.

b) $s=\underline{nat}$.

Nach Korollar 7 gilt

$$(\forall t_1,t_2 \in T_{\Sigma,n})(t_1 \sim_{E,n} t_2 \;\Leftrightarrow\; t_1 \;\underline{is}\; t_2)$$

und wegen der Reflexivität von $\sim_{E \cup E0,n}$ und $T_{\Sigma,n} \subseteq T_{\Sigma \cup \Sigma 0,n}$

gilt $(\forall t \in T_{\Sigma,n})(t \sim_{E,n} t \;\Rightarrow\; t \sim_{E \cup E0,n} t)$. $\square$

Lemma TK.SP1

Beh.: $(\forall s \in \{\underline{dis},\underline{nat}\})(\forall t_1,t_2 \in T_{\Sigma,s})(t_1 \sim_{E,s} t_2 \;\Rightarrow\; t_1 \sim_{E \cup E1,s} t_2)$

Bew.:

Völlig analog zum Beweis von Lemma TK.SP0. $\square$

Korollar T.ERW.SP0

Beh.: SP0 ist t-Erweiterung von SP.

Bew.:

SP und SP0 sind konsistente t-Spezifikationen nach Voraussetzung und vollständige t-Spezifikationen nach Lemma V.SP und Lemma V.SP0. Außerdem ist SP0 t-konsistent und t-vollständig auf SP nach Lemma TV.SP0 und TK.SP0. Nach Satz 6 ist SP0 daher t-Erweiterung von SP. $\square$

Korollar T.ERW.SP1

Beh.: SP1 ist t-Erweiterung von SP.

Bew.:

Völlig analog zum Beweis von Korollar T.ERW.SP0. $\square$

Wir haben gezeigt, daß SP0 und SP1 die Generalvoraussetzung erfüllen. Nach Definition 42 und nach 3.2.2. ist IMPL=(ΣS0,Σ0D,ES0,E0D,EΣ0) eine schwache T-Implementierung von SP0 mit Hilfe von SP1, da I1 und I2 vollständig und konsistent sind.
Um zu zeigen, daß IMPL auch eine T-Implementierung ist, muß noch bewiesen werden, daß I1 auf SP1 und I2 auf I1 t-konsistent ist und daß IMPL R-korrekt ist. Wir bereiten zunächst die restlichen t-Konsistenznachweise vor.

Lemma EQV.I1

Beh.: $(\forall t_1, t_2 \in T_{\Sigma I1,d})(t_1 \equiv_{EI1,d} t_2 \iff t_1 \text{ EQV } t_2 \equiv_{EI1,d} tt)$

Bew.:

Seien $t_1, t_2 \in T_{\Sigma I1,d}$. Nach Lemma V.I1 gibt es $tv_1, tv_2 \in \{tt,ff\}$, sodaß für i=1,2 $\quad t_i \equiv_{EI1,d} tv_i \quad$ gilt.

"=>"

Sei $t_1 \equiv_{EI1,d} t_2$. Dann gilt auch $tv_1 \equiv_{EI1,d} tv_2$ und aufgrund der Konsistenz von I1 muß tv_1 <u>is</u> $tv_2=:tv$ gelten. Es gilt

$$t_1 \text{ EQV } t_2 \equiv_{EI1,d} tv \text{ EQV } tv \equiv_{EI1,d} tt$$

für beide möglichen Belegungen von tv nach Gleichung E.5 bzw. E.6 und E.2 und der Transitivität von $\equiv_{EI1,d}$.

"<="

Sei t_1 EQV $t_2 \equiv_{EI1,d}$ tt. Dann gilt nach SE von $\equiv_{EI1}$ auch
*125 tv_1 EQV $tv_2 \equiv_{EI1,d}$ tt.

a) Sei tv_1 <u>is</u> tt.
Wäre jetzt tv_2 <u>is</u> ff, dann erhielten wir mit Gleichung E.5 tv_1 EQV $tv_2 \equiv_{EI1,d}$ ff; dies ergibt mit *125 einen Widerspruch zur Konsistenz von I1. Also gilt tv_2 <u>is</u> tt <u>is</u> tv_1 und damit $t_1 \equiv_{EI1,d} t_2$ nach Reflexivität von $\equiv_{EI1,d}$ und SE von $\equiv_{EI1}$.

b) Sei tv_1 <u>is</u> ff.
Dieser Fall verläuft analog zu a). □

Die nächsten Beweise sind mit Untersuchungen von Kontexten verbunden, für die uns noch einige Begriffe fehlen. Wir wollen zum Beispiel Terme benennen, die mit einem bestimmten Operationssymbol beginnen.

<u>Definition 46</u> (σ-wurzelnd,Wurzel)

Sei SPEC=(S,Σ,E) eine Spezifikation. Ein Term $t\in T_{\Sigma(X),s}$ mit $s\in S$ heißt <u>σ-wurzelnd</u>, wenn gilt
$(\exists n\in\omega)(\exists s_1,\ldots,s_n\in S)(\exists(t_1,\ldots,t_n)\in T_{\Sigma(X),s_1}\times\ldots\times T_{\Sigma(X),s_n})$
$(\sigma\in\Sigma_{s_1\ldots s_n,s} \wedge t$ <u>is</u> $\sigma(t_1,\ldots,t_n))$.
Man sagt: σ ist die <u>Wurzel</u> von t, t hat die Wurzel σ. □

Als nächstes definieren wir die "contexts of interest" und zeigen, daß man zur Festlegung der t-Kongruenzrelation nicht alle Kontexte, sondern nur die "contexts of interest" betrachten muß.

<u>Definition 47</u> (COI(Σ,s))

Es sei SPEC'=(S',Σ',E') eine t-Spezifikation mit S'-{<u>dis</u>}$\neq\emptyset$.

Sei

$\Sigma'_{DIS}:=\{\sigma \mid (\exists n\in\omega)(\exists s_1,\ldots,s_n\in S)((\exists i\in\{1,\ldots,n\})(s_i\neq \underline{dis}) \wedge$
$\sigma\in\Sigma'_{s_1\ldots s_n,s})\}$

und für alle s$\in$S'-{<u>dis</u>} sei

COI(Σ',s):={ct$\in C_{\Sigma'}$(s,dis)|ct hat eine Wurzel $\sigma\in\Sigma'_{DIS}$}. □

<u>Lemma 17</u>

Sei SPEC'=(S',Σ',E') eine t-Spezifikation mit S'-{<u>dis</u>}$\neq\emptyset$.

Dann gilt

$(\forall s\in S'-\{\underline{dis}\})(\forall p,q\in T_{\Sigma',s})$

$(((\forall ct\in COI(\Sigma',s))(ct[p/x^1_s] \equiv_{E',d} ct[q/x^1_s])) \Rightarrow$

$((\forall\tau\in C_{\Sigma'}(s,dis))(\tau[p/x^1_s] \equiv_{E',d} \tau[q/x^1_s])))$.

<u>Bew.:</u>

Sei s$\in$S'-{<u>dis</u>}, seien $p,q\in T_{\Sigma',s}$, es gelte

*126 $(\forall ct\in COI(\Sigma',s))(ct[p/x^1_s] \equiv_{E',d} ct[q/x^1_s])$

und es sei $\tau\in C_{\Sigma'}(s,dis)$. Wir unterscheiden zwei Fälle.

a) $\tau\in COI(\Sigma',s)$.

Gilt nach Voraussetzung.

b) $\tau\notin COI(\Sigma',s)$.

Dann hat τ eine Wurzel $\sigma\in\Sigma'_{w,dis}$ mit w$\in${<u>dis</u>}$^+$.

Wegen s$\neq$<u>dis</u> muß τ einen Subterm ct$\in$COI(Σ',s) haben.

Nach *126 gilt

$$ct[p/x^1_s] \equiv_{E',d} ct[q/x^1_s]$$

und die SE von $\equiv_{E'}$ liefert dann sofort

$$\tau[p/x^1_s] \ \underline{is}\ \tau[ct[p/x^1_s]/ct] \equiv_{E',d} \tau[ct[q/x^1_s]/ct] \ \underline{is}\ \tau[q/x^1_s].$$

□

Beim t-Konsistenznachweis brauchen wir die "alten" Kontexte der zu erweiternden t-Spezifikation, die ja auch Teilmenge der "neuen" Kontexte sind, nicht mehr zu berücksichtigen, denn für zwei "alte" Terme t und t' mit $t \sim_{E'} t'$ gilt ja bereits

$ct[t/x] \equiv_{E',d} ct[t'/x]$

für alle "alten" Kontexte ct und nach Lemma 7 folgt daraus

$ct[t/x] \equiv_{E'',d} ct[t'/x]$.

Formal wird dies im nächsten Lemma festgehalten.

Lemma 18

Seien $SPEC'=(S',\Sigma',E')$ und $SPEC''=(S'',\Sigma'',E'')$ t-Spezifikationen mit

(1) $S' \subseteq S'' \wedge (S'-\{\underline{dis}\}) \neq \emptyset$

(2) $(\forall w \in S^*)(\forall s \in S)(\Sigma'_{w,s} \subseteq \Sigma''_{w,s})$

(3) $E' \subseteq E''$.

Wenn für alle $p,q \in T_{\Sigma',s}$ mit $s \in S'-\{\underline{dis}\}$ und $p \sim_{E',s} q$ gilt

*127 $(\forall ct \in COI(\Sigma'',s)-C_{\Sigma'}(s,dis))(ct[p/x^1_s] \equiv_{E'',d} ct[q/x^1_s])$,

dann ist SPEC" t-konsistent auf SPEC' bzgl. der Sorte s.

Bew.:

Seien $p,q \in T_{\Sigma',s}$ mit $s \in S'-\{\underline{dis}\}$ und mit $p \sim_{E',s} q$, d.h. es gilt $(\forall ct \in C_{\Sigma'}(s,dis))(ct[p/x^1_s] \equiv_{E',d} ct[q/x^1_s])$.

Nach Lemma 7 gilt auch

$(\forall ct \in C_{\Sigma'}(s,dis))(ct[p/x^1_s] \equiv_{E'',d} ct[q/x^1_s])$,

was zusammen mit *127 ergibt

$(\forall \tau \in COI(\Sigma',s))\ (\tau[p/x^1_s] \equiv_{E'',d} \tau[q/x^1_s])$.

Nach Lemma 17 folgt daraus

$(\forall \tau \in C_{\Sigma''}(s,dis))(\tau[p/x^1_s] \equiv_{E'',d} \tau[q/x^1_s])$,

was äquivalent ist zu $p \sim_{E'',s} q$.

Unter der Voraussetzung, daß *127 gilt, haben wir gezeigt

$(\forall p,q \in T_{\Sigma',s})(p \sim_{E',s} q \Rightarrow p \sim_{E'',s} q)$,

d.h. SPEC" ist t-konsistent auf SPEC' bzgl. $s \in S'-\{\underline{dis}\}$. □

Jetzt zeigen wir, daß I1 t-konsistent auf SP1 ist.

Lemma TK.I1

Beh.: $(\forall s\in\{\underline{dis},\underline{nat},\underline{array}\})(\forall t,t'\in T_{\Sigma\cup\Sigma 1,s})$

$(t \sim_{E\cup E1,s} t' \Rightarrow t \sim_{EI1,s} t')$

Bew.:

Durch Fallunterscheidung über die Sorten $s\in S\cup S1$.

a) $s=\underline{dis}$.

I1 ist t-konsistent auf SP1 bzgl. dis nach Korollar 5.

b) $s=\underline{nat}$.

Wir zeigen zunächst $\equiv_{E\cup E1,n} = \sim_{E\cup E1,n}$, d.h. SP1 ist kategorisch bzgl. nat.

b.1) $\equiv_{E\cup E1,n} \subseteq \sim_{E\cup E1,n}$ gilt nach Lemma 9.

b.2) Wir zeigen $\sim_{E\cup E1,n} \subseteq \equiv_{E\cup E1,n}$, d.h.

$(\forall t_1,t_2\in T_{\Sigma\cup\Sigma 1,n})(t_1 \sim_{E\cup E1,n} t_2 \Rightarrow t_1 \equiv_{E\cup E1,n} t_2)$.

Seien also $t_1,t_2\in T_{\Sigma\cup\Sigma 1,n}$ und es gelte $t_1 \sim_{E\cup E1,n} t_2$.

Nach Lemma N.SP1 gibt es $j_1,j_2\in\omega_0$, sodaß für i=1,2

*128 $t_i \equiv_{E\cup E1,n} S^{j_i}(\emptyset)$.

Wäre jetzt $j_1\neq j_2$, dann gälte nach SE von $\equiv_{E\cup E1}$, *128 und Lemma K.NAT.SP1

$EQN(t_1,t_2) \equiv_{E\cup E1,d} EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset)) \equiv_{E\cup E1,d} ff$ und

$EQN(t_1,t_1) \equiv_{E\cup E1,d} EQN(S^{j_1}(\emptyset),S^{j_1}(\emptyset)) \equiv_{E\cup E1,d} tt$.

Da SP1 konsistent ist, gälte auch

$EQN(t_1,t_1) \not\equiv_{E\cup E1,d} EQN(t_1,t_2)$.

Damit hätten wir einen Kontext $EQN(t_1,x_n^1)$ gefunden, sodaß

$EQN(t_1,x_n^1)[t_1/x_n^1] \not\equiv_{E\cup E1,d} EQN(t_1,x_n^1)[t_2/x_n^1]$.

Dies widerspricht der Voraussetzung $t_1 \sim_{E\cup E1,n} t_2$.

Also muß $j_1=j_2$ sein.

Aus *128 folgt mit der Transitivität und Symmetrie von

$\equiv_{E\cup E1,n}$: $\quad t_1 \equiv_{E\cup E1,n} t_2$, $\quad$ q.e.d.

Es gilt also $\sim_{E\cup E1,n} = \equiv_{E\cup E1,n}$. Nach Lemma 7 gilt $\equiv_{E\cup E1,n} \subseteq \equiv_{EI1,n}$ und nach Lemma 9 gilt $\equiv_{EI1,n} \subseteq \sim_{EI1,n}$.

Zusammenfassend heißt das: Es gilt $\sim_{E\cup E1,n} \subseteq \sim_{EI1,n}$, was nichts anderes bedeutet als

$$(\forall t,t' \in T_{\Sigma\cup\Sigma1,n})(t \sim_{E\cup E1,n} t' \Rightarrow t \sim_{EI1,n} t'),$$

d.h. I1 ist t-konsistent auf SP1 bzgl. <u>nat</u>.

c) s=<u>array</u>.

Wir wollen zeigen, daß die Voraussetzungen von Lemma 18 erfüllt sind. Dazu bestimmen wir die "contexts of interest". Es ist

$CA := COI(\Sigma I1,a) - C_{\Sigma\cup\Sigma1}(a,d) =$

$\{EQS(ST(t_1,t_2),ST(t_3,t_4)) \mid t_1,t_3 \in T_{\Sigma I1(X),a} \wedge t_2,t_4 \in T_{\Sigma I1(X),n}$
$\wedge\ (\exists^1 i \in \{1,2,3,4\})(var(t_i)=\{x_a^1\} \wedge (\forall j \in \{1,2,3,4\})(i \neq j \Rightarrow var(t_j)=\emptyset))\}$.

Wir zerlegen CA disjunkt in vier Teilmengen indem wir definieren:

Für i=1,2,3,4 sei $\quad Ci := \{ct \in CA \mid var(t_i)=\{x_a^1\}\}$.

Jetzt sind wir in der Lage, eine Fallunterscheidung über die Kontexte zu machen, in denen x_a^1 in einem bestimmten Subterm t_i, i=1,2,3,4, auftritt.

Wir zeigen

*129 $\quad (\forall p,q \in T_{\Sigma\cup\Sigma1,a})(p \sim_{E\cup E1,a} q \Rightarrow$
$\quad\quad ((\forall i \in \{1,2,3,4\})(\forall ct \in Ci)(ct[p/x_a^1] \equiv_{EI1,d} ct[q/x_a^1])))$

durch Fallunterscheidung über $i \in \{1,2,3,4\}$.

c.1) i=2.

Sei $ct \in C2$, d.h. ct <u>is</u> $EQS(ST(t_1,t_2),ST(t_3,t_4))$ und x_a^1 ist Subterm vom Term t_2 der Sorte <u>nat</u>.

Wegen $T_{\Sigma I1,n}=T_{\Sigma\cup\Sigma1,n}$ und SE von $\sim_{E\cup E1}$ gilt

$^{*}130 \quad (\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow t_2[p/x_a^1] \sim_{E\cup E1,n} t_2[q/x_a^1])$.

In b) dieses Beweises hatten wir gezeigt, daß

$\equiv_{E\cup E1,n} = \sim_{E\cup E1,n}$. Daher gilt mit $^{*}130$ auch

$$(\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow t_2[p/x_a^1] \equiv_{E\cup E1,n} t_2[q/x_a^1]),$$

woraus nach Lemma 7 folgt

$$(\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow t_2[p/x_a^1] \equiv_{EI1,n} t_2[q/x_a^1]).$$

Die SE von $\equiv_{EI1}$ liefert

$$(\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow$$
$$ct[t_2[p/x_a^1]/t_2] \equiv_{EI1,d} ct[t_2[q/x_a^1]/t_2]),$$

was äquivalent ist zu

$$(\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow ct[p/x_a^1] \equiv_{EI1,d} ct[q/x_a^1]).$$

c.2) i=4.

Dieser Fall verläuft symmetrisch zu c.1).

c.3) i=1.

Sei $ct \in C1$, d.h. ct <u>is</u> EQS(ST(t_1,t_2),ST(t_3,t_4)), x_a^1 ist Subterm von t_1 und t_1 ist von der Sorte <u>array</u>.

Wegen $T_{\Sigma I1,a} = T_{\Sigma\cup\Sigma 1,a}$ und SE von $\sim_{E\cup E1}$ gilt

$$(\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow t_1[p/x_a^1] \sim_{E\cup E1,a} t_1[q/x_a^1]).$$

Dann ist es aber gleichwertig, ob wir

$$(\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow$$
$$EQS(ST(t_1,t_2),ST(t_3,t_4))[p/x_a^1] \equiv_{EI1,d} EQS(ST(t_1,t_2),ST(t_3,t_4))[q/x_a^1])$$

oder ob wir

$$(\forall p,q \in T_{\Sigma\cup\Sigma 1,a})(t_1[p/x_a^1] \sim_{E\cup E1,a} t_1[q/x_a^1] \Rightarrow$$
$$EQS(ST(x_a^2,t_2),ST(t_3,t_4))[t_1[p/x_a^1]/x_a^2] \equiv_{EI1,d}$$
$$EQS(ST(x_a^2,t_2),ST(t_3,t_4))[t_1[q/x_a^1]/x_a^2])$$

zeigen, denn wegen

EQS(ST(t_1,t_2),ST(t_3,t_4))[t'/x_a^1] <u>is</u> EQS(ST(x_a^2,t_2),ST(t_3,t_4))[$t_1[t'/x_a^1]/x_a^2$]

mit t' <u>is</u> p,q folgt die mittlere Aussage aus den beiden anderen. Da sich jeder <u>array</u>-Term $t \in T_{\Sigma\cup\Sigma 1,a}$ darstellen läßt als $t_1[t'/x_a^1]$ mit $t_1 \in T_{\Sigma\cup\Sigma 1(X),a}$ und $t' \in T_{\Sigma\cup\Sigma 1,a}$ kann man in der letzten Aussage $t_1[p/x_a^1]$ und $t_1[q/x_a^1]$ wieder durch p und q ersetzen.

Außerdem gibt es nach Lemma N.I1 $j,k\epsilon\omega_0$, sodaß zusammen mit der SE von $\equiv_{EI1}$ für alle $t\epsilon T_{\Sigma\cup\Sigma 1,a}$ gilt

$EQS(ST(x_a^1,t_2),ST(t_3,t_4))[t/x_a^1] \equiv_{EI1,d} EQS(ST(x_a^1,S^j(\emptyset)),ST(t_3,S^k(\emptyset)))[t/x_a^1]$.

Wir können daher für diesen Unterfall die Menge C1 der zu betrachtenden "contexts of interest" einschränken auf

$CON:=\{EQS(ST(x_a^1,S^j(\emptyset)),ST(t\ ,\ S^k(\emptyset)))\mid j,k\epsilon\omega_0 \wedge t\epsilon T_{\Sigma I1,a}\}$.

Wir zeigen

*131 $(\forall p,q\epsilon T_{\Sigma\cup\Sigma 1,a})(p \sim_{E\cup E1,a} q \Rightarrow$
$(\forall ct\epsilon CON)(ct[p/x_a^1] \equiv_{EI1,d} ct[q/x_a^1]))$

durch Induktion über $j\epsilon\omega_0$.

c.3.i) Induktionsanfang.

Sei $EQS(ST(x_a^1,S^j(\emptyset)),ST(t,S^k(\emptyset)))\epsilon CON$ mit $j=0$, d.h. $S^j(\emptyset)$ <u>is</u> $\emptyset$. Wir unterscheiden über $k\epsilon\omega_0$.

c.3.i.a) Sei $k=0$, d.h. $S^k(\emptyset)$ <u>is</u> $\emptyset$.

Dann gilt für alle $p,q\epsilon T_{\Sigma\cup\Sigma 1,a}$

$EQS(ST(x_a^1,\emptyset),ST(t,\emptyset))[p/x_a^1] \equiv_{EI1,d} tt$ und

$EQS(ST(x_a^1,\emptyset),ST(t,\emptyset))[q/x_a^1] \equiv_{EI1,d} tt$

nach Gleichung E0D.1; *131 ist für diesen Unterfall gezeigt.

c.3.i.b) Sei $k>0$, d.h. $S^k(\emptyset)$ ~~is~~ $\emptyset$.

Dann gilt für alle $p,q\epsilon T_{\Sigma\cup\Sigma 1,a}$

$EQS(ST(x_a^1,\emptyset),ST(t,S^k(\emptyset)))[p/x_a^1] \equiv_{EI1,d} ff$ und

$EQS(ST(x_a^1,\emptyset),ST(t,S^k(\emptyset)))[q/x_a^1] \equiv_{EI1,d} ff$

nach Gleichung E0D.2.

c.3.ii) Induktionsannahme.

Die Behauptung gelte für alle $EQS(ST(x_a^1,S^j(\emptyset)),ST(t,S^k(\emptyset)))\epsilon CON$ mit beliebigem, aber festem $j\epsilon\omega_0$ und beliebigem $k\epsilon\omega_0$.

c.3.iii) Induktionsschluß.

Sei $EQS(ST(x_a^1,S^{j+1}(\emptyset)),ST(t,S^k(\emptyset)))\epsilon CON$. Es ist $S^{j+1}(\emptyset)$ ~~is~~ $\emptyset$.

Wir unterscheiden über $k \epsilon \omega_0$.

c.3.iii.a) Sei k=0, d.h. $S^k(\emptyset)$ is $\emptyset$.

Dann gilt für alle $p,q \epsilon T_{\Sigma \cup \Sigma 1,a}$

$$EQS(ST(x_a^1,S^{j+1}(\emptyset)),ST(t,\emptyset))[p/x_a^1] \equiv_{EI1,d} ff \quad \text{und}$$
$$EQS(ST(x_a^1,S^{j+1}(\emptyset)),ST(t,\emptyset))[q/x_a^1] \equiv_{EI1,d} ff$$

nach Gleichung E0D.3.

c.3.iii.b) Sei k>0, d.h. $S^k(\emptyset)$ ~~is~~ $\emptyset$.

Seien $p,q \epsilon T_{\Sigma \cup \Sigma 1,a}$ mit $p \sim_{E \cup E1,a} q$. Zunächst zeigen wir, daß die Bedingungen der Gleichungen E0D.4 bis E0D.6 auswertbar sind.

Wegen $T_{\Sigma I1,a} = T_{\Sigma \cup \Sigma 1,a}$ ist bereits $t \epsilon T_{\Sigma \cup \Sigma 1,a}$ und damit

$p_1 := INDEX(p,S^{j+1}(\emptyset))$ EQV $INDEX(t,S^k(\emptyset)) \epsilon T_{\Sigma \cup \Sigma 1,d}$,

$p_2 := INDEX(q,S^{j+1}(\emptyset))$ EQV $INDEX(t,S^k(\emptyset)) \epsilon T_{\Sigma \cup \Sigma 1,d}$,

$p_3 := EQN(AC(p,S^{j+1}(\emptyset)),AC(t,S^k(\emptyset))) \epsilon T_{\Sigma \cup \Sigma 1,d}$ und

$p_4 := EQN(AC(q,S^{j+1}(\emptyset)),AC(t,S^k(\emptyset))) \epsilon T_{\Sigma \cup \Sigma 1,d}$.

Wegen der SE von $\sim_{E \cup E1}$ und wegen $\equiv_{E \cup E1,d} = \sim_{E \cup E1,d}$ gilt

$$p_1 \equiv_{E \cup E1,d} p_2 \quad \text{und} \quad p_3 \equiv_{E \cup E1,d} p_4.$$

Nach Lemma 7 folgt daraus

*132 $\quad p_1 \equiv_{EI1,d} p_2 \quad$ und $\quad p_3 \equiv_{EI1,d} p_4$.

Dann gibt es aufgrund der Vollständigkeit und Konsistenz von I1 genau drei Unterfälle.

c.3.iii.b.1) Seien $p_1 \equiv_{EI1,d} tt$ und $p_3 \equiv_{EI1,d} tt$.

Damit gilt nach *132 auch $p_2 \equiv_{EI1,d} tt$ und $p_4 \equiv_{EI1,d} tt$.

Die Bedingungen der Gleichung E0D.4 sind erfüllt und wir erhalten

*133 $\quad EQS(ST(x_a^1,S^{j+1}(\emptyset)),ST(t,S^k(\emptyset)))[p/x_a^1] \equiv_{EI1,d}$
$\quad EQS(ST(x_a^1,S^j(\emptyset)),ST(t,S^{k-1}(\emptyset)))[p/x_a^1]$ und

*134 $\quad EQS(ST(x_a^1,S^{j+1}(\emptyset)),ST(t,S^k(\emptyset)))[q/x_a^1] \equiv_{EI1,d}$
$\quad EQS(ST(x_a^1,S^j(\emptyset)),ST(t,S^{k-1}(\emptyset)))[q/x_a^1]$.

Nach Induktionsannahme c.3.ii) gilt

*135 $EQS(ST(x_a^1,s^j(\emptyset)),ST(t,s^{k-1}(\emptyset)))[p/x_a^1] \equiv_{EI1,d}$

$EQS(ST(x_a^1,s^j(\emptyset)),ST(t,s^{k-1}(\emptyset)))[q/x_a^1]$.

Der Rest ist Transitivität von $\equiv_{EI1,d}$ mit *133 bis *135.

c.3.iii.b.2) Sei $p_1 \equiv_{EI1,d}$ tt und $p_3 \equiv_{EI1,d}$ ff.

Damit gilt nach *132 auch $p_2 \equiv_{EI1,d}$ tt und $p_4 \equiv_{EI1,d}$ ff.

Die Bedingungen von Gleichung EOD.5 und nur die Bedingungen von EOD.5 sind erfüllt und wir erhalten

$EQS(ST(x_a^1,s^{j+1}(\emptyset)),ST(t,s^k(\emptyset)))[p/x_a^1] \equiv_{EI1,d}$ ff und

$EQS(ST(x_a^1,s^{j+1}(\emptyset)),ST(t,s^k(\emptyset)))[q/x_a^1] \equiv_{EI1,d}$ ff.

Der Rest ist Symmetrie und Transitivität von $\equiv_{EI1,d}$.

c.3.iii.b.3) Sei $p_1 \equiv_{EI1,d}$ ff.

Dann gilt nach *132 auch $p_2 \equiv_{EI1,d}$ ff.

Damit sind die Bedingungen von Gleichung EOD.6 und nur die von EOD.6 erfüllt und wir erhalten

$EQS(ST(x_a^1,s^{j+1}(\emptyset)),ST(t,s^k(\emptyset)))[p/x_a^1] \equiv_{EI1,d}$ ff und

$EQS(ST(x_a^1,s^{j+1}(\emptyset)),ST(t,s^k(\emptyset)))[q/x_a^1] \equiv_{EI1,d}$ ff.

Damit ist die Induktion unter c.3) geschlossen und *131 gezeigt.

c.4) i=3.

Dieser Fall verläuft symmetrisch zu c.3).

Damit ist *129 und die zu *129 äquivalente Aussage

$(\forall p,q \in T_{\Sigma \cup \Sigma 1,a})(p \sim_{E \cup E1,a} q \Rightarrow$

$(\forall ct \in CA)(ct[p/x_a^1] \equiv_{EI1,d} ct[q/x_a^1]))$

gezeigt. Die Voraussetzung von Lemma 18 ist erfüllt und daher ist I1 auf SP1 t-konsistent bzgl. der Sorte <u>array</u> und mit a) bis c) gilt, daß I1 t-konsistent auf SP1 ist. □

Lemma EQ.STACK.I1

Beh.: $(\forall t_1,t_2,t_3 \in T_{\Sigma I1,s})$

$$\begin{aligned}(\; 1)\quad & EQS(t_1,t_1) \equiv_{EI1,d} tt\\ 2)\quad & EQS(t_1,t_2) \equiv_{EI1,d} tt \Rightarrow EQS(t_2,t_1) \equiv_{EI1,d} tt \wedge\\ 3)\quad & (EQS(t_1,t_2) \equiv_{EI1,d} tt \wedge EQS(t_2,t_3) \equiv_{EI1,d} tt) \Rightarrow\\ & EQS(t_1,t_3) \equiv_{EI1,d} tt \;)\end{aligned}$$

Bew.:

ad 1) Sei $ST(q,r) \in T_{\Sigma I1,s}$ mit $q \in T_{\Sigma I1,a}$ und $r \in T_{\Sigma I1,n}$.

Nach Lemma N.I1 gibt es ein $j \in \omega_0$, sodaß $r \equiv_{EI1,n} S^j(\emptyset)$ und nach SE von $\equiv_{EI1}$ auch

$EQS(ST(q,r),ST(q,r)) \equiv_{EI1,d} EQS(ST(q,S^j(\emptyset)),ST(q,S^j(\emptyset)))$ gilt.

Wir zeigen durch Induktion über $j \in \omega_0$

$$EQS(ST(q,S^j(\emptyset)),ST(q,S^j(\emptyset))) \equiv_{EI1,d} tt.$$

1.i) Induktionsanfang.

Sei $j=0$, d.h. $S^j(\emptyset)$ is $\emptyset$.

Dann folgt die Gültigkeit der Behauptung aus Gleichung EOD.1.

1.ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $j \in \omega_0$.

1.iii) Induktionsschluß.

Wir zeigen zunächst, daß die Bedingungen von Gleichung EOD.4 erfüllt sind.

Nach Lemma V.I1 gibt es ein $tv \in \{tt,ff\}$, sodaß

$$INDEX(q,S^{j+1}(\emptyset)) \equiv_{EI1,d} tv.$$

Aus Lemma EQV.I1 und der Reflexivität von $\equiv_{EI1,d}$ folgt, daß

$$tv \; EQV \; tv \equiv_{EI1,d} tt,$$

und zwar für beide möglichen Belegungen von tv.

Nach Korollar EQ.NAT.I1, Punkt 1), gilt

$$EQN(AC(q,S^{j+1}(\emptyset)),AC(q,S^{j+1}(\emptyset))) \equiv_{EI1,d} tt.$$

Wir können also Gleichung EOD.4 anwenden und erhalten

$$EQS(ST(q,S^{j+1}(\emptyset)),ST(q,S^{j+1}(\emptyset))) \equiv_{EI1,d} EQS(ST(q,S^{j}(\emptyset)),ST(q,S^{j}(\emptyset)))$$
$$\equiv_{EI1,d} tt,$$

wobei die letzte Kongruenz nach Induktionsannahme 1.ii) gilt.

Damit ist die Induktion unter ad 1) geschlossen.

ad 2) Seien $ST(q_1,r_1),ST(q_2,r_2)\in T_{\Sigma I1,s}$ mit $q_1,q_2\in T_{\Sigma I1,a}$ und $r_1,r_2\in T_{\Sigma I1,n}$.

Nach Lemma N.I1 gibt es $j,k\in\omega_0$, sodaß zusammen mit der SE von $\equiv_{EI1}$ gilt

$$EQS(ST(q_1,r_1),ST(q_2,r_2)) \equiv_{EI1,d} EQS(ST(q_1,S^{j}(\emptyset)),ST(q_2,S^{k}(\emptyset))).$$

Wir zeigen durch Induktion über $j\in\omega_0$ die zu 2) äquivalente Behauptung

$$(\forall j,k\in\omega_0)(\forall q_1,q_2\in T_{\Sigma I1,a})$$
$$(EQS(ST(q_1,S^{j}(\emptyset)),ST(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} tt \Rightarrow$$
$$EQS(ST(q_2,S^{k}(\emptyset)),ST(q_1,S^{j}(\emptyset))) \equiv_{EI1,d} tt).$$

2.i) Induktionsanfang.

Sei $j=0$, d.h. $S^{j}(\emptyset)$ <u>is</u> $\emptyset$.

Seien $q_1,q_2\in T_{\Sigma I1,a}$ und $k\in\omega_0$. Es gelte

$$EQS(ST(q_1,\emptyset),ST(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} tt.$$

Wäre $k>0$, d.h. $S^{k}(\emptyset)$ ~~is~~ $\emptyset$, dann gälte nach Gleichung E0D.2

$$EQS(ST(q_1,\emptyset),ST(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} ff,$$

was einen Widerspruch zur Konsistenz von I1 implizierte.

Also ist $k=0$ und nach Gleichung E0D.1 gilt

$$EQS(ST(q_2,S^{k}(\emptyset)),ST(q_1,\emptyset)) \equiv_{EI1,d} tt.$$

2.ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $j\in\omega_0$ und für alle $k\in\omega_0$.

2.iii) Induktionsschluß.

Seien $q_1,q_2\in T_{\Sigma I1,a}$ und $k\in\omega_0$. Es gelte

*136 $EQS(ST(q_1,S^{j+1}(\emptyset)),ST(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} tt.$

Wäre k=0, d.h. $S^{k}(\emptyset)\ \underline{\text{is}}\ \emptyset$, dann gälte wegen $S^{j+1}(\emptyset)\ \cancel{\underline{\text{is}}}\ \emptyset$

Nach Gleichung EOD.3

$$EQS(ST(q_1,S^{j+1}(\emptyset)),ST(q_2,\emptyset)) \equiv_{EI1,d} ff,$$

woraus sich ein Widerspruch zur Konsistenz von I1 ergäbe.

Also ist k>0, d.h. $S^{k}(\emptyset)\ \cancel{\underline{\text{is}}}\ \emptyset$.

Jetzt zeigen wir, daß die Bedingungen von Gleichung EOD.4 und nur diese erfüllt sind.

Gälte $INDEX(q_1,S^{j+1}(\emptyset))\ EQV\ INDEX(q_2,S^{k}(\emptyset)) \equiv_{EI1,d} ff$,

dann könnten wir Gleichung EOD.6 anwenden und erhielten mit

$$EQS(ST(q_1,S^{j+1}(\emptyset)),ST(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} ff$$

erneut einen Widerspruch zur Konsistenz von I1. Also gilt

*137 $INDEX(q_1,S^{j+1}(\emptyset))\ EQV\ INDEX(q_2,S^{k}(\emptyset)) \equiv_{EI1,d} tt.$

Gälte $EQN(AC(q_1,S^{j+1}(\emptyset)),AC(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} ff$,

dann könnten wir Gleichung EOD.5 anwenden und erhielten

$$EQS(ST(q_1,S^{j+1}(\emptyset)),ST(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} ff,$$

was, wie wir wissen, nicht sein kann. Es muß also gelten

*138 $EQN(AC(q_1,S^{j+1}(\emptyset)),AC(q_2,S^{k}(\emptyset))) \equiv_{EI1,d} tt.$

Damit sind die Bedingungen von Gleichung EOD.4 erfüllt und wir erhalten zusammen mit *136

$$EQS(ST(q_1,S^{j+1}(\emptyset)),ST(q_2,S^{k}(\emptyset))) \equiv_{EI1,d}$$

$$EQS(ST(q_1,S^{j}(\emptyset)),ST(q_2,S^{k-1}(\emptyset))) \equiv_{EI1,d} tt.$$

Nach Induktionsannahme 2.ii) gilt damit

*139 $EQS(ST(q_2,S^{k-1}(\emptyset)),ST(q_1,S^{j}(\emptyset))) \equiv_{EI1,d} tt.$

Nach Lemma EQV.I1 und Symmetrie von $\equiv_{EI1,d}$ gilt mit *137 auch

*140 $INDEX(q_2,S^{k}(\emptyset))\ EQV\ INDEX(q_1,S^{j+1}(\emptyset))) \equiv_{EI1,d} tt.$

Nach Punkt 2) von Korollar EQ.NAT.I1 und *138 gilt

*141 $EQN(AC(q_2,S^{k}(\emptyset)),AC(q_1,S^{j+1}(\emptyset))) \equiv_{EI1,d} tt.$

Durch *140 und *141 wird die Anwendung von Gleichung EOD.4 ermöglicht und wir erhalten

*142 $EQS(ST(q_2,S^{k}(\emptyset)),ST(q_1,S^{j+1}(\emptyset))) \equiv_{EI1,d}$

$$EQS(ST(q_2,S^{k-1}(\emptyset)),ST(q_1,S^{j}(\emptyset))) \quad .$$

Die Transitivität von $\equiv_{EI1,d}$ zusammen mit *142 und *139 sichern uns die Gültigkeit der Behauptung.

Damit ist die Induktion unter ad 2) geschlossen.

ad 3) Seien $ST(q_1,r_1), ST(q_2,r_2), ST(q_3,r_3) \in T_{\Sigma I1,s}$ mit $q_1,q_2,q_3 \in T_{\Sigma I1,a}$ und $r_1,r_2,r_3 \in T_{\Sigma I1,n}$.

Nach Lemma N.I1 gibt es $j,k,l \in \omega_0$, sodaß zusammen mit der SE von $\equiv_{EI1}$ gilt

$$EQS(ST(q_1,r_1),ST(q_2,r_2)) \equiv_{EI1,d} EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset))),$$
$$EQS(ST(q_2,r_2),ST(q_3,r_3)) \equiv_{EI1,d} EQS(ST(q_2,S^k(\emptyset)),ST(q_3,S^l(\emptyset))) \text{ und}$$
$$EQS(ST(q_1,r_1),ST(q_3,r_3)) \equiv_{EI1,d} EQS(ST(q_1,S^j(\emptyset)),ST(q_3,S^l(\emptyset))).$$

Wir zeigen durch Induktion über $k \in \omega_0$ die zu 3) äquivalente Aussage

$$(\forall j,k,l \in \omega_0)(\forall q_1,q_2,q_3 \in T_{\Sigma I1,a})$$
$$((EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d} tt \quad \wedge$$
$$EQS(ST(q_2,S^k(\emptyset)),ST(q_3,S^l(\emptyset))) \equiv_{EI1,d} tt) \quad \Rightarrow$$
$$EQS(ST(q_1,S^j(\emptyset)),ST(q_3,S^l(\emptyset))) \equiv_{EI1,d} tt).$$

3.i) Induktionsanfang.

Sei k=0, d.h. $S^k(\emptyset)$ <u>is</u> $\emptyset$.

Seien $j,l \in \omega_0$ und $q_1,q_2,q_3 \in T_{\Sigma I1,a}$. Es gelte

$$EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d} tt \quad \text{und}$$
$$EQS(ST(q_2,S^k(\emptyset)),ST(q_3,S^l(\emptyset))) \equiv_{EI1,d} tt.$$

Wie unter 2.i) zeigt man durch Widerspruch zur Konsistenz von I1, daß j=0 und l=0, d.h. $S^j(\emptyset)$ <u>is</u> $\emptyset$ und $S^l(\emptyset)$ <u>is</u> $\emptyset$ gelten muß. Dann gilt

$$EQS(ST(q_1,S^j(\emptyset)),ST(q_3,S^l(\emptyset))) \equiv_{EI1,d} tt$$

nach Gleichung E0D.1.

3.ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $k \in \omega_0$ und für alle $j,l \in \omega_0$.

3.iii) Induktionsschluß.

Seien $q_1, q_2, q_3 \in T_{\sum I1,a}$ und $j, l \in \omega_0$. Es gelte

$*143 \quad EQS(ST(q_1, S^j(\emptyset)), ST(q_2, S^{k+1}(\emptyset))) \equiv_{EI1,d} tt$ und

$*144 \quad EQS(ST(q_2, S^{k+1}(\emptyset)), ST(q_3, S^l(\emptyset))) \equiv_{EI1,d} tt$.

Wie unter 2.iii) zeigt man durch Widerspruch zur Konsistenz von I1, daß $j, l > 0$, d.h. $S^j(\emptyset), S^l(\emptyset)$ ~~is~~ $\emptyset$, und daß

$*145 \quad INDEX(q_1, S^j(\emptyset))\ EQV\ INDEX(q_2, S^{k+1}(\emptyset)) \equiv_{EI1,d} tt$,

$*146 \quad INDEX(q_2, S^{k+1}(\emptyset))\ EQV\ INDEX(q_3, S^l(\emptyset)) \equiv_{EI1,d} tt$,

$*147 \quad EQN(AC(q_1, S^j(\emptyset)), AC(q_2, S^{k+1}(\emptyset))) \equiv_{EI1,d} tt$ und daß

$*148 \quad EQN(AC(q_2, S^{k+1}(\emptyset)), AC(q_3, S^l(\emptyset))) \equiv_{EI1,d} tt$ gelten

muß. Damit sind die Bedingungen von Gleichung E0D.4 und nur die von E0D.4 erfüllt und wir erhalten daraus

$$*149 \begin{cases} EQS(ST(q_1, S^j(\emptyset)), ST(q_2, S^{k+1}(\emptyset))) \equiv_{EI1,d} \\ EQS(ST(q_1, S^{j-1}(\emptyset)), ST(q_2, S^k(\emptyset))) \equiv_{EI1,d} tt \end{cases}$$

und

$$*150 \begin{cases} EQS(ST(q_2, S^{k+1}(\emptyset)), ST(q_3, S^l(\emptyset))) \equiv_{EI1,d} \\ EQS(ST(q_2, S^k(\emptyset)), ST(q_3, S^{l-1}(\emptyset))) \equiv_{EI1,d} tt, \end{cases}$$

wobei die letzte Kongruenz von $*149$ und $*150$ jeweils aus der Transitivität von $\equiv_{EI1,d}$ zusammen mit $*143$ und $*144$ folgt. Auf $*149$ und $*150$ trifft die Induktionsannahme 3.ii) zu und daher gilt

$*151 \quad EQS(ST(q_1, S^{j-1}(\emptyset)), ST(q_3, S^{l-1}(\emptyset))) \equiv_{EI1,d} tt$.

Nach Lemma EQV.I1 und SE von $\equiv_{EI1}$ folgt aus $*145$ und $*146$

$*152 \quad INDEX(q_1, S^j(\emptyset))\ EQV\ INDEX(q_3, S^l(\emptyset)) \equiv_{EI1,d} tt$.

Aus $*147$, $*148$ und Punkt 3) von Korollar EQ.NAT.I1 folgt

$*153 \quad EQN(AC(q_1, S^j(\emptyset)), AC(q_3, S^l(\emptyset))) \equiv_{EI1,d} tt$.

Durch $*152$ und $*153$ wird die Anwendung von Gleichung E0D.4 und nur die von E0D.4 ermöglicht und wir erhalten

$$*154 \begin{cases} EQS(ST(q_1, S^j(\emptyset)), ST(q_3, S^l(\emptyset))) \equiv_{EI1,d} \\ EQS(ST(q_1, S^{j-1}(\emptyset)), ST(q_3, S^{l-1}(\emptyset))). \end{cases}$$

Aus $*154$ und $*151$ folgt aufgrund der Transitivität von $\equiv_{EI1,d}$

die Gültigkeit der Behauptung für den Induktionsschluß.

Damit ist die Induktion unter ad 3) geschlossen. $\square$

Als nächstes charakterisieren wir den stack-Anteil der t-Kongruenz $\sim_{EI1}$. Zwei pointer/array-Paare sind dann t-kongruent, wenn ihre pointer gleich (unter $\equiv_{EI1,n}$) sind und die array-Komponenten an allen Stellen $\emptyset \leq j \leq$pointer die gleichen Einträge haben bzw. an den gleichen Stellen undefinierte Einträge besitzen.

Lemma K.STACK.I1

Beh.: $(\forall q_1,q_2 \in T_{\Sigma I1,a})(\forall r_1,r_2 \in T_{\Sigma I1,n})$

$(ST(q_1,r_1) \sim_{EI1,s} ST(q_2,r_2) \quad <=>$

$($ 1) $r_1 \equiv_{EI1,n} r_2 \quad \wedge$

2) $((\exists j \in \omega_0)(r_1 \equiv_{EI1,n} S^j(\emptyset) \wedge j>0) => ((\forall i \in \{1,...,j\})$

$($ 2.a) $IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \quad \wedge$

2.b) $AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset)) \quad)))))$

Bew.:

Wir zeigen zunächst die beiden Hilfsaussagen

*155 $(\forall t_1,t_2 \in T_{\Sigma I1,s})(EQS(t_1,t_2) \equiv_{EI1,d} tt \quad => \quad t_1 \sim_{EI1,s} t_2)$

und

*156 $(\forall t_1,t_2 \in T_{\Sigma I1,s})(EQS(t_1,t_2) \equiv_{EI1,d} ff \quad => \quad t_1 \not\sim_{EI1,s} t_2)$.

ad *155)

Seien $t_1,t_2 \in T_{\Sigma I1,s}$ und es gelte

*157 $EQS(t_1,t_2) \equiv_{EI1,d} tt$. Wir zeigen

*158 $(\forall ct \in COI(\Sigma I1,s))(ct[t_1/x_s^1] \equiv_{EI1,d} ct[t_2/x_s^1])$.

Es ist $COI(\Sigma I1,s)=\{EQS(x_s^1,p) \mid p \in T_{\Sigma I1,s}\} \cup \{EQS(p,x_s^1) \mid p \in T_{\Sigma I1,s}\}$.

Sei $p \in T_{\Sigma I1,s}$. Dann gibt es nach Lemma V.I1 vier Fälle.

a) Es gelte $EQS(x_s^1,p)[t_1/x_s^1] \equiv_{EI1,d} ff$.

Dann muß auch $EQS(x_s^1,p)[t_2/x_s^1] \equiv_{EI1,d} ff$ gelten, denn aus

$EQS(x_s^1,p)[t_2/x_s^1] \equiv_{EI1,d} tt$ und *157 folgte nach Punkt 3) von Lemma EQ.STACK.I1

$EQS(t_1,p) \equiv_{EI1,d} tt.$

Wegen $EQS(t_1,p)$ <u>is</u> $EQS(x_s^1,p)[t_1/x_s^1]$ erhielten wir damit einen Widerspruch zur Konsistenz von I1.

b) Es gelte $EQS(x_s^1,p)[t_1/x_s^1] \equiv_{EI1,d} tt.$

Nach Punkt 2) von Lemma EQ.STACK.I1 gilt mit *157 auch

$EQS(t_2,t_1) \equiv_{EI1,d} tt.$ Dies, die Voraussetzung von b) und Punkt 3) von Lemma EQ.STACK.I1 ergeben

$EQS(x_s^1,p)[t_2/x_s^1]$ <u>is</u> $EQS(t_2,p) \equiv_{EI1,d} tt.$

Die Beweise von

$EQS(p,x_s^1)[t_1/x_s^1] \equiv_{EI1,d} ff \Rightarrow EQS(p,x_s^1)[t_2/x_s^1] \equiv_{EI1,d} ff$

und

$EQS(p,x_s^1)[t_1/x_s^1] \equiv_{EI1,d} tt \Rightarrow EQS(p,x_s^1)[t_2/x_s^1] \equiv_{EI1,d} tt$

verlaufen symmetrisch zu a) und b).

Damit ist die Gültigkeit von *158 gezeigt und nach Lemma 17 gilt $t_1 \sim_{EI1,s} t_2$; also ist auch *155 gezeigt.

ad *156)

Seien $t_1,t_2 \in T_{\Sigma I1,s}$ und es gelte

*159 $EQS(t_1,t_2) \equiv_{EI1,d} ff.$

Nach Punkt 1) von Lemma EQ.STACK.I1 gilt

*160 $EQS(t_1,t_1) \equiv_{EI1,d} tt.$

Da I1 konsistent ist, folgt aus *159 und *160:

Es gibt einen Kontext $EQS(t_1,x_s^1) \in COI(\Sigma I1,s) \subseteq C_{\Sigma I1}(s,dis)$, sodaß

$EQS(t_1,x_s^1)[t_1/x_s^1] \not\equiv_{EI1,d} EQS(t_1,x_s^1)[t_2/x_s^1]$,

d.h. es gilt $t_1 \not\sim_{EI1,s} t_2$.

Aufgrund der Vollständigkeit und Konsistenz von I1 sind zu *155 und *156 die Aussagen

*161 $(\forall t_1,t_2 \in T_{\Sigma I1,s})(t_1 \not\sim_{EI1,s} t_2 \Rightarrow EQS(t_1,t_2) \equiv_{EI1,s} ff)$

bzw.

$^{*}162 \quad (\forall t_1,t_2 \in T_{\Sigma I1,s})(t_1 \sim_{EI1,s} t_2 \;\Rightarrow\; EQS(t_1,t_2) \equiv_{EI1,d} tt)$

äquivalent.

Für den Rest des Beweises brauchen wir nach Lemma N.I1 und SE von $\equiv_{EI1}$ nur noch <u>stack</u>-Terme zu betrachten, deren pointer-Komponente die kanonische Form $S^j(\emptyset)$ mit $j \in \omega_0$ hat.

c) "=>"

Wir zeigen

$$
\begin{array}{l}
(\forall j,k \in \omega_0)(\forall q_1,q_2 \in T_{\Sigma I1,a}) \\
(ST(q_1,S^j(\emptyset)) \sim_{EI1,s} ST(q_2,S^k(\emptyset)) \quad \Rightarrow \\
(\;\;3)\quad j=k \qquad \wedge \\
\quad\;\; 4)\quad (j>0 \;\Rightarrow\; ((\forall i \in \{1,\dots,j\}) \\
\qquad\qquad (\quad IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \quad \wedge \\
\qquad\qquad\quad\;\; AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset)) \quad)))))
\end{array}
$$

durch Induktion über $j \in \omega_0$.

c.i) Induktionsanfang.

Seien $q_1,q_2 \in T_{\Sigma I1,a}$, $j,k \in \omega_0$ und es gelte

$^{*}163 \quad ST(q_1,S^j(\emptyset)) \sim_{EI1,s} ST(q_2,S^k(\emptyset))$.

c.i.a) Sei j=0, d.h. $S^j(\emptyset)$ <u>is</u> $\emptyset$.

Angenommen, es ist k>0, d.h. $S^k(\emptyset)$ ~~is~~ $\emptyset$. Dann gilt

$$EQS(ST(q_1,\emptyset),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d} ff$$

nach Gleichung E0D.2, woraus sich wegen $^{*}156$ ein Widerspruch zu $^{*}163$ ergibt. Also ist j=k=0, d.h. 3) gilt und 4) ist nicht anwendbar.

c.i.b) Sei j=1, d.h. $S^j(\emptyset)$ <u>is</u> $S(\emptyset)$ ~~is~~ $\emptyset$.

Angenommen, es ist k=0, d.h. $S^k(\emptyset)$ <u>is</u> $\emptyset$. Dann gilt

$$EQS(ST(q_1,S(\emptyset)),ST(q_2,\emptyset)) \equiv_{EI1,d} ff$$

nach Gleichung E0D.3, woraus sich wegen $^{*}156$ ein Widerspruch zu $^{*}163$ ergibt. Also ist k>0, d.h. $S^k(\emptyset)$ ~~is~~ $\emptyset$.

Angenommen, es gilt $\mathrm{IN}(q_1,S(\emptyset)) \not\equiv_{EI1,d} \mathrm{IN}(q_2,S^k(\emptyset))$. Daraus, aus Lemma EQV.I1 und der Vollständigkeit und Konsistenz von I1 folgte dann

$$\mathrm{IN}(q_1,S(\emptyset)) \;\mathrm{EQV}\; \mathrm{IN}(q_2,S^k(\emptyset)) \equiv_{EI1,d} ff.$$

Wir könnten Gleichung EOD.6 anwenden und erhielten

$$\mathrm{EQS}(\mathrm{ST}(q_1,S(\emptyset)),\mathrm{ST}(q_2,S^k(\emptyset))) \equiv_{EI1,d} ff,$$

was nach *156 einen Widerspruch zu *163 implizierte.

Also gilt

*164 $\mathrm{IN}(q_1,S(\emptyset)) \;\mathrm{EQV}\; \mathrm{IN}(q_2,S^k(\emptyset)) \equiv_{EI1,d} tt$ bzw.

*165 $\mathrm{IN}(q_1,S(\emptyset)) \equiv_{EI1,d} \mathrm{IN}(q_2,S^k(\emptyset))$.

Angenommen, es gilt $\mathrm{AC}(q_1,S(\emptyset)) \not\equiv_{EI1,n} \mathrm{AC}(q_2,S^k(\emptyset))$. Nach Konsistenz und Vollständigkeit von I1 und nach Korollar 12 folgte daraus

*166 $\mathrm{EQN}(\mathrm{AC}(q_1,S(\emptyset)),\mathrm{AC}(q_2,S^k(\emptyset))) \equiv_{EI1,d} ff.$

Wir könnten wegen *164 und *166 Gleichung EOD.5 anwenden und erhielten

$$\mathrm{EQS}(\mathrm{ST}(q_1,S(\emptyset)),\mathrm{ST}(q_2,S^k(\emptyset))) \equiv_{EI1,d} ff,$$

was nach *156 einen Widerspruch zu *163 implizierte.

Also gilt

*167 $\mathrm{EQN}(\mathrm{AC}(q_1,S(\emptyset)),\mathrm{AC}(q_2,S^k(\emptyset))) \equiv_{EI1,d} tt$ bzw.

*168 $\mathrm{AC}(q_1,S(\emptyset)) \equiv_{EI1,n} \mathrm{AC}(q_2,S^k(\emptyset))$.

Angenommen, es ist $k>1$. Dann wäre auch $S^{k-1}(\emptyset) \;\not\underline{\mathrm{is}}\; \emptyset$. Wegen *164 und *167 könnten wir Gleichung EOD.4 und dann EOD.2 anwenden und erhielten

$$\mathrm{EQS}(\mathrm{ST}(q_1,S(\emptyset)),\mathrm{ST}(q_2,S^k(\emptyset))) \equiv_{EI1,d}$$
$$\mathrm{EQS}(\mathrm{ST}(q_1,\emptyset),\mathrm{ST}(q_2,S^{k-1}(\emptyset))) \equiv_{EI1,d} ff,$$

was nach *156 erneut einen Widerspruch zu *163 hervorruft.

Also gilt $k=j=1$.

Damit und mit *165 und *168 sind die Aussagen 3) und 4) für diesen Unterfall des Induktionsanfanges gezeigt.

c.ii) Induktionsannahme.

Die Behauptung gilt für ein beliebiges, aber festes $j \in \omega_0$ und für alle $k \in \omega_0$.

c.iii) Induktionsschluß.

Seien $q_1, q_2 \in T_{\Sigma I1,a}$ und $k \in \omega_0$. Es gelte

*169 $\quad ST(q_1, S^{j+1}(\emptyset)) \sim_{EI1,s} ST(q_2, S^k(\emptyset))$.

Wie in c.i.b) zeigt man, daß k>0, d.h. $S^k(\emptyset)$ ~~is~~ $\emptyset$ ist, daß

*170 $\quad IN(q_1, S^{j+1}(\emptyset))\ EQV\ IN(q_2, S^k(\emptyset)) \equiv_{EI1,d} tt$ bzw.

*171 $\quad IN(q_1, S^{j+1}(\emptyset)) \equiv_{EI1,d} IN(q_2, S^k(\emptyset))$ und daß

*172 $\quad EQN(AC(q_1, S^{j+1}(\emptyset)), AC(q_2, S^k(\emptyset))) \equiv_{EI1,d} tt$ bzw.

*173 $\quad AC(q_1, S^{j+1}(\emptyset)) \equiv_{EI1,n} AC(q_2, S^k(\emptyset))$ gilt.

Wegen *170 und *174 ist Gleichung EOD.4 und nur diese anwendbar und wir erhalten

*174 $\quad EQS(ST(q_1, S^{j+1}(\emptyset)), ST(q_2, S^k(\emptyset))) \equiv_{EI1,d}$
$\quad EQS(ST(q_1, S^j(\emptyset)), ST(q_2, S^{k-1}(\emptyset)))$.

Wegen *169 und *162 gilt

$$EQS(ST(q_1, S^{j+1}(\emptyset)), ST(q_2, S^k(\emptyset))) \equiv_{EI1,d} tt$$

und dies, *174 und die Äquivalenzeigenschaften von $\equiv_{EI1,d}$ ergeben

$$EQS(ST(q_1, S^j(\emptyset)), ST(q_2, S^{k-1}(\emptyset))) \equiv_{EI1,d} tt.$$

Daraus folgt nach *155 $\quad ST(q_1, S^j(\emptyset)) \sim_{EI1,s} ST(q_2, S^{k-1}(\emptyset))$.

Nach Induktionsannahme c.ii) gilt

$j=k-1$ und

*175 $\quad (j>0 \Rightarrow ((\forall i \in \{1,...,j\})(IN(q_1, S^i(\emptyset)) \equiv_{EI1,d} IN(q_2, S^i(\emptyset)) \wedge$
$\quad AC(q_1, S^i(\emptyset)) \equiv_{EI1,n} AC(q_2, S^i(\emptyset))))).$

Also gilt auch $j+1=k$, d.h. 3) ist erfüllt, und mit *175, *171 und *173 gilt

*176 $\quad (j+1>0 \Rightarrow ((\forall i \in \{1,...,j+1\})(IN(q_1, S^i(\emptyset)) \equiv_{EI1,d} IN(q_2, S^i(\emptyset)) \wedge$
$\quad AC(q_1, S^i(\emptyset)) \equiv_{EI1,n} AC(q_2, S^i(\emptyset))))),$

d.h. 4) ist erfüllt.

Damit ist die Induktion unter c) geschlossen.

d) "<="

Wir zeigen durch Induktion über $j \in \omega_0$

$(\forall j,k \in \omega_0)\ (\forall q_1,q_2 \in T_{\Sigma I1,a})$

$(($ 5) $j=k \quad \wedge$

6) $(j>0 \quad \Rightarrow \quad ((\forall i \in \{1,\ldots,j\})(IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \quad \wedge$

$AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset)) \;))))$

$\Rightarrow \quad ST(q_1,S^j(\emptyset)) \sim_{EI1,s} ST(q_2,S^k(\emptyset)) \quad).$

d.i) Induktionsanfang.

Seien $j,k \in \omega_0$, $j=k$ und $q_1,q_2 \in T_{\Sigma I1,a}$.

d.i.a) Sei $j=k=0$, d.h. $S^j(\emptyset)$ is $S^k(\emptyset)$ is $\emptyset$.

Dann gilt nach Gleichung E0D.1

$$EQS(ST(q_1,\emptyset),ST(q_2,\emptyset)) \equiv_{EI1,d} tt$$

und nach *155 folgt daraus $ST(q_1,\emptyset) \sim_{EI1,s} ST(q_2,\emptyset)$.

d.i.b) Sei $j=k=1$, d.h. $S^j(\emptyset)$ is $S^k(\emptyset)$ is $S(\emptyset)$.

Es gelte $IN(q_1,S(\emptyset)) \equiv_{EI1,d} IN(q_2,S(\emptyset))$

und $AC(q_1,S(\emptyset)) \equiv_{EI1,n} AC(q_2,S(\emptyset))$.

Nach Lemma EQV.I1 und Korollar 12 gilt dann auch

$$IN(q_1,S(\emptyset)) \text{ EQV } IN(q_2,S(\emptyset)) \equiv_{EI1,d} tt \qquad \text{und}$$

$$EQN(AC(q_1,S(\emptyset)),AC(q_2,S(\emptyset))) \equiv_{EI1,d} tt.$$

Wir können daher Gleichung E0D.4 und E0D.1 anwenden und erhalten

$$EQS(ST(q_1,S(\emptyset)),ST(q_2,S(\emptyset))) \equiv_{EI1,d} EQS(ST(q_1,\emptyset),ST(q_2,\emptyset))$$
$$\equiv_{EI1,d} tt,$$

woraus nach *155 folgt: $ST(q_1,S(\emptyset)) \sim_{EI1,s} ST(q_2,S(\emptyset))$.

d.ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $j \in \omega_0$ und für alle $k \in \omega_0$.

d.iii) Induktionsschluß.

Seien $q_1,q_2 \in T_{\Sigma I1,a}$, $k \in \omega_0$ und es gelte $j+1=k$ und

*177 $(\forall i\in\{1,\dots,j+1\})(IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \quad \wedge$

$AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset)) \;).$

Dann gilt natürlich auch $j=k-1$ und

$(j>0 \;\; \Rightarrow \;\; ((\forall i\in\{1,\dots,j\})(IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \quad \wedge$

$AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset)) \;))),$

woraus nach Induktionsannahme d.ii) folgt:

$$ST(q_1,S^j(\emptyset)) \sim_{EI1,s} ST(q_2,S^{k-1}(\emptyset)).$$

Wegen *162 gilt damit auch

*178 $EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^{k-1}(\emptyset))) \equiv_{EI1,d} tt.$

Aus *177 folgt mit Lemma EQV.I1 und Korollar 12 speziell

$IN(q_1,S^{j+1}(\emptyset)) \; EQV \; IN(q_2,S^k(\emptyset)) \equiv_{EI1,d} tt$ und

$EQN(AC(q_1,S^{j+1}(\emptyset)),AC(q_2,S^k(\emptyset))) \equiv_{EI1,d} tt.$

Wir können Gleichung EOD.4 anwenden und erhalten

*179 $EQS(ST(q_1,S^{j+1}(\emptyset)),ST(q_2,S^k(\emptyset))) \equiv_{EI1,d}$

$EQS(ST(q_1,S^j(\emptyset)),ST(q_2,S^{k-1}(\emptyset))).$

Aus *178, *179, Transitivität von $\equiv_{EI1,d}$ und aus *155 folgt

$$ST(q_1,S^{j+1}(\emptyset)) \sim_{EI1,s} ST(q_2,S^k(\emptyset)).$$

Damit ist die Induktion unter d) geschlossen. $\square$

Lemma TK.I2

Beh.: $(\forall s\in\{\underline{dis},\underline{nat},\underline{array},\underline{stack}\})(\forall t_1,t_2\in T_{\Sigma I1,s})$

$(t_1 \sim_{EI1,s} t_2 \;\; \Rightarrow \;\; t_1 \sim_{EI2,s} t_2)$

Bew.:

Durch Fallunterscheidung über die Sorten $s\in S\cup S0\cup S1$.

a) $s=\underline{dis}$.

I2 ist t-konsistent auf I1 bzgl. $\underline{dis}$ nach Korollar 5.

b) $s=\underline{nat}$.

Dieser Fall verläuft völlig analog zu b) aus dem Beweis von Lemma TK.I1.

c) $s \in \{\underline{\text{array}}, \underline{\text{stack}}\}$.

Wir wollen zeigen, daß die Voraussetzungen von Lemma 18 erfüllt sind. Dabei gehen wir von folgenden Überlegungen aus: Durch die Hinzunahme der restlichen $\Sigma 0$-Operationen NEWST, PUSH, POP und TOP und deren definierenden Gleichungen zu I1 entstehen in I2 neue "contexts of interest". Bei der Bestimmung des array- und stack-Anteils der t-Kongruenz $\sim_{EI2}$ kann man, da I2 i- bzw. t-vollständig auf I1 ist, alle σ-wurzelnden Subterme, $\sigma \in \{NEWST, PUSH, POP, TOP\}$, eines Kontextes ct bis auf einen, der die Variable enthält, durch äquivalente Terme aus I1 ersetzen. Für diesen letzten Subterm t_0 gibt es dann eine Gleichung $(L,R) \in E\Sigma 0$ und Belegungen b,b' Bel$(T_{\Sigma I2}, L, R)$, sodaß für alle $t, t' \in T_{\Sigma I1, s'}$ mit $s' \in \{\underline{\text{array}}, \underline{\text{stack}}\}$ gilt:

$$t_0[t/x^1_{s'}] \ \underline{\text{is}}\ int_b(L) \qquad \text{und} \qquad t_0[t'/x^1_{s'}] \ \underline{\text{is}}\ int_{b'}(L).$$

Jetzt kann man $int_b(L)$ bzw. $int_{b'}(L)$ durch $int_b(R)$ bzw. $int_{b'}(R)$ ersetzen und es gilt

$$ct[t/x^1_{s'}] \equiv_{EI2,d} ct[int_b(R)/int_b(L)] \qquad \text{bzw.}$$
$$ct[t'/x^1_{s'}] \equiv_{EI2,d} ct[int_{b'}(R)/int_{b'}(L)].$$

Die rechts stehenden Terme sind, da ja durch die Ersetzung auch der letzte σ-wurzelnde Term in ct durch einen ΣI1-Term substituiert wurde, dis-Terme aus $T_{\Sigma I1}$. Wenn man zeigen kann, daß $int_b(R) \sim_{EI1,s"} int_{b'}(R)$ gilt, dann weiß man auch, daß

$$ct[int_b(R)/int_b(L)] \equiv_{EI2,d} ct[int_{b'}(R)/int_{b'}(L)]$$

gilt und damit

$$ct[t/x^1_{s'}] \equiv_{EI2,d} ct[t'/x^1_{s'}].$$

Wir wollen die oben angestellten Überlegungen formalisieren, indem wir die Hilfsbehauptung *180 zeigen. Dabei steht *A für: t ist Subterm von $int_b(L)$ und $int_{b'}(L)$ entsteht aus $int_b(L)$ durch einmaliges ersetzen von t durch t'.

$^*180 \quad (\forall s' \in \{\underline{array},\underline{stack}\})(\forall t,t' \in T_{\Sigma I1,s'})$

$$(^*181 \begin{cases} (t \sim_{EI1,s'} t' \;\wedge \\ (\forall (L,R) \in \{E\Sigma 0.8,\ldots,E\Sigma 0.12\})(\forall b,b' \in Bel(T_{\Sigma I2},L,R)) \\ ((^*A \wedge int_b(R),int_{b'}(R) \in T_{\Sigma I1}) \\ \Rightarrow \; int_b(R) \sim_{EI1} int_{b'}(R))) \end{cases}$$

$$\Rightarrow \quad (\forall ct \in COI(\Sigma I2,s')-C_{\Sigma I1}(s',\underline{dis}))(ct[t/x^1_{s'}] \equiv_{EI2,d} ct[t'/x^1_{s'}])).$$

c.1) Beweis von ^{*180}.

Wir geben zunächst ein Verfahren zur Ersetzung der meisten σ-wurzelnden Subterme in den dis-Kontexten für s'-Terme an und untersuchen dann die übrigbleibenden dis-Kontexte.

c.1.1) Sei $s' \in \{\underline{array},\underline{stack}\}$, sei $ct \in COI(\Sigma I2,s')-C_{\Sigma I1}(s',dis)$ mit ct is $\sigma(t_1,t_2)$, $\sigma \in \{EQN,EQA,INDEX,EQS\}$ und t_1,t_2 seien zur Stelligkeit von σ passende Terme aus $T_{\Sigma I2(X)}$. Sei $x^1_{s'}$ Subterm von t_k für ein festes $k \in \{1,2\}$. Dann enthält t_{3-k} keine Variable und wir wenden die Punkte c.1.1.1) bis c.1.1.3) auf $\sigma(t_1,t_2)$ an.

c.1.1.1) Falls t_{3-k} keine Operationssymbole aus {NEWST,PUSH, POP,TOP} enthält, dann benenne t_{3-k} in t'_{3-k} um und fahre fort mit c.1.1.3), andernfalls fahre fort mit c.1.1.2).

c.1.1.2) t_{3-k} enthält Operationssymbole aus {NEWST,PUSH,POP, TOP}. Dann gibt es nach den Lemmata N.I2, A.I2 oder S.I2 einen Term $t'_{3-k} \in T_{\Sigma I2}$, sodaß $t_{3-k} \equiv_{EI2} t'_{3-k}$ und t'_{3-k} enthält keine Operationssymbole aus {NEWST,PUSH,POP,TOP} mehr. Ersetze t_{3-k} in $\sigma(t_1,t_2)$ durch t'_{3-k} und fahre fort mit c.1.1.3).

c.1.1.3) Ersetze in t_k alle Subterme τ von t_k, die $x^1_{s'}$ nicht als Subterm haben, in denen jedoch Operationssymbole aus {NEWST,PUSH,POP,TOP} vorkommen, durch die nach den Lemmata N.I2, A.I2 und S.I2 existierenden, zu τ $\equiv_{EI2}$-äquivalenten Terme τ', die nur noch Operationssymbole aus $\Sigma I1$ enthalten.

Der so entstandene Term heiße t_k'.

Das Verfahren endet hier und wir haben einen Kontext $\sigma(t_1',t_2')\in COI(\Sigma I2,s')$ erhalten, sodaß gilt:

$(\forall t\in T_{\Sigma I1,s'})(\sigma(t_1,t_2)[t/x_{s'}^1] \equiv_{EI2,d} \sigma(t_1',t_2')[t/x_{s'}^1])$.

c.1.2) Für $s'\in\{\underline{array},\underline{stack}\}$ kann man die Menge $COI(\Sigma I2,s')-C_{\Sigma I1}(s',dis)$ in zwei disjunkte Teilmengen zerlegen. Sei $n(\sigma(t_1',t_2'))$ die Anzahl der in $\sigma(t_1',t_2')\in COI(\Sigma I2,s')$ vorkommenden Operationssymbole aus {NEWST,PUSH,POP,TOP}.
Wir definieren

$C1(s'):=\{\sigma(t_1,t_2)\in COI(\Sigma I2,s')-C_{\Sigma I1}(s',dis)\,|\,n(\sigma(t_1',t_2'))=0\}$ und

$C2(s'):=\{\sigma(t_1,t_2)\in COI(\Sigma I2,s')-C_{\Sigma I1}(s',dis)\,|\,n(\sigma(t_1',t_2'))>0\}$,

wobei $\sigma(t_1',t_2')$ das Resultat der Anwendung des Verfahrens c.1.1) auf $\sigma(t_1,t_2)$ bezeichnet. Es ist

$COI(\Sigma I2,s')-C_{\Sigma I1}(s',dis)=C1(s')\cup C2(s')$.

Wir machen eine Fallunterscheidung über $Ci(s')$, $i=1,2$.

c.1.2.1) Es gilt für alle $s'\in\{\underline{array},\underline{stack}\}$ und für alle $t,t'\in T_{\Sigma I1,s'}$ mit $t \sim_{EI1,s'} t'$

$$\begin{aligned}(\forall\sigma(t_1,t_2)\in C1(s'))(\sigma(t_1,t_2)[t/x_{s'}^1] &\equiv_{EI2,d} \sigma(t_1',t_2')[t/x_{s'}^1]\\ &\equiv_{EI2,d} \sigma(t_1',t_2')[t'/x_{s'}^1]\\ &\equiv_{EI2,d} \sigma(t_1,t_2)[t'/x_{s'}^1]).\end{aligned}$$

Dabei gilt die mittlere i-Kongruenz aus folgenden Gründen:
Wegen $n(\sigma(t_1',t_2'))=0$ ist $\sigma(t_1',t_2')\in C_{\Sigma I1}(s',dis)$. Mit $t \sim_{EI1,s'} t'$ gilt $\sigma(t_1',t_2')[t/x_{s'}^1] \equiv_{EI1,d} \sigma(t_1',t_2')[t'/x_{s'}^1]$ und der Rest ist Lemma 7.

Für den Teil C1(s') der Kontexte gilt *180 also unabhängig von den Aussagen über die Gleichungen $E\Sigma 0.8,\ldots,E\Sigma 0.12$.

c.1.1.2) Für den Teil C2(s') zeigen wir jetzt

$^*182 \quad (\forall s' \in \{\underline{array}, \underline{stack}\})(\forall t,t' \in T_{\Sigma I1,s'})$

$$(^*183 \begin{cases} (t \sim_{EI1,s'} t' \;\wedge \\ (\forall (L,R) \in \{E\Sigma 0.8,\ldots,E\Sigma 0.12\})(\forall b,b' \in Bel(T_{\Sigma I2},L,R)) \\ ((^*A \wedge int_b(R), int_{b'}(R) \in T_{\Sigma I1}) \\ \Rightarrow \; int_b(R) \sim_{EI1} int_{b'}(R))) \end{cases}$$

$$\Rightarrow \quad (\forall \sigma(t_1,t_2) \in C2(s'))(\sigma(t_1,t_2)[t/x^1_{s'}] \equiv_{EI2,d} \sigma(t_1,t_2)[t'/x^1_{s'}]))$$

durch Induktion über die Anzahl $n(t'_k)$ der im Subterm t'_k von $\sigma(t'_1,t'_2)$ vorkommenden Operationssymbole aus {PUSH,POP, TOP} für ein beliebiges, aber festes $k \in \{1,2\}$; dabei haben wir ausgenutzt, daß es genau ein $k \in \{1,2\}$ gibt, sodaß $n(\sigma(t'_1,t'_2))=n(t'_k)$ und daß NEWST nach Anwendung von c.1.1) in $\sigma(t'_1,t'_2)$ nicht mehr vorkommt.

c.1.2.2.i) Induktionsanfang.

Sei $s' \in \{\underline{array}, \underline{stack}\}$, $t,t' \in T_{\Sigma I1,s'}$ und sei $\sigma(t'_1,t'_2) \in C2(s')$ mit $n(\sigma(t'_1,t'_2))=n(t'_k)=1$. Es gelte ^{*183}.

Sei t_0 der σ'-wurzelnde Subterm von t'_k, $\sigma' \in$ {PUSH,POP,TOP}, der $x^1_{s'}$ als Subterm enthält. Dann kommen in allen anderen echten Subtermen von $t_0[t/x^1_{s'}]$ bzw. $t_0[t'/x^1_{s'}]$ nur noch Operationssymbole aus $\Sigma I1$ vor. Jetzt gibt es eine Gleichung $(L,R) \in \{E\Sigma 0.8,\ldots,E\Sigma 0.12\}$ und Belegungen $b,b' \in Bel(T_{\Sigma I2},L,R)$, sodaß $int_b(L)$ $\underline{is}$ $t_0[t/x^1_{s'}]$ und $int_{b'}(L)$ $\underline{is}$ $t_0[t'/x^1_{s'}]$, d.h. (L,R) ist anwendbar und wir erhalten

$$t_0[t/x^1_{s'}] \equiv_{EI2,s''} t_{01} \quad \text{bzw.} \quad t_0[t'/x^1_{s'}] \equiv_{EI2,s''} t_{02},$$

wobei $s'' \in \{\underline{nat}, \underline{stack}\}$, t_{01} $\underline{is}$ $int_b(R)$, t_{02} $\underline{is}$ $int_{b'}(R)$ und $t_{01},t_{02} \in T_{\Sigma I1,s''}$. Nach ^{*183} ist $t_{01} \sim_{EI1,s''} t_{02}$.

Jetzt gilt nach SE von $\equiv_{EI2}$

$$\begin{aligned} \sigma(t_1,t_2)[t/x^1_{s'}] &\equiv_{EI2,d} \sigma(t'_1,t'_2)[t/x^1_{s'}] \\ &\equiv_{EI2,d} \sigma(t'_1,t'_2)[t_{01}/t_0] \\ &\equiv_{EI2,d} \sigma(t'_1,t'_2)[t_{02}/t_0] \\ &\equiv_{EI2,d} \sigma(t'_1,t'_2)[t'/x^1_{s'}] \\ &\equiv_{EI2,d} \sigma(t_1,t_2)[t'/x^1_{s'}]. \end{aligned}$$

Dabei gilt die mittlere i-Kongruenz, weil mit $t_{o1} \sim_{EI1,s''} t_{o2}$ und $\sigma(t_1',t_2')[t_{o1}/t_o], \sigma(t_1',t_2')[t_{o2}/t_o] \in T_{\Sigma I1,d}$

auch $\sigma(t_1',t_2')[t_{o1}/t_o] \equiv_{EI1,d} \sigma(t_1',t_2')[t_{o2}/t_o]$

und nach Lemma 7 $\sigma(t_1',t_2')[t_{o1}/t_o] \equiv_{EI2,d} \sigma(t_1',t_2')[t_{o2}/t_o]$.

c.1.2.2.ii) Induktionsannahme.

Die Behauptung gelte für alle $s' \in \{\underline{array},\underline{stack}\}$ und für alle $\sigma(t_1,t_2) \in C2(s')$ mit $n(\sigma(t_1',t_2'))=n(t_k')=m$, wobei $m \in \omega$ beliebig, aber fest ist.

c.1.2.2.iii) Induktionsschluß.

Sei $s' \in \{\underline{array},\underline{stack}\}$, seien $t,t' \in T_{\Sigma I1,s'}$, sei $\sigma(t_1',t_2') \in C2(s')$ mit $n(\sigma(t_1',t_2'))=n(t_k')=m+1$ und es gelte *183.
Die m+1 σ'-wurzelnden Subterme, $\sigma' \in \{PUSH,POP,TOP\}$, von t_k' haben alle $x^1_{s'}$ als Subterm, d.h. sie bilden eine Kette.
Sei t_o der <u>kleinste</u> σ'-wurzelnde Subterm von t_k', d.h. für alle σ''-wurzelnden Subterme t_o' von t_k' mit $\sigma'' \in \{PUSH,POP,TOP\}$ gilt: t_o ist Subterm von t_o'.
In allen echten Subtermen von $t_o[t/x^1_{s'}]$ bzw. $t_o[t'/x^1_{s'}]$ kommen dann nur noch Operationssymbole aus $\Sigma I1$ vor. Jetzt gibt es eine Gleichung $(L,R) \in \{E\Sigma0.8,\ldots,E\Sigma0.12\}$ und Belegungen $b,b' \in Bel(T_{\Sigma I2},L,R)$, sodaß $int_b(L)$ <u>is</u> $t_o[t/x^1_{s'}]$ und $int_{b'}(L)$ <u>is</u> $t_o[t'/x^1_{s'}]$, d.h. (L,R) ist anwendbar und wir erhalten

$$t_o[t/x^1_{s'}] \equiv_{EI2,s''} t_{o1} \quad \text{bzw.} \quad t_o[t'/x^1_{s'}] \equiv_{EI2,s''} t_{o2},$$

wobei $s'' \in \{\underline{nat},\underline{stack}\}$, t_{o1} <u>is</u> $int_b(R)$, t_{o2} <u>is</u> $int_{b'}(R)$ und $t_{o1},t_{o2} \in T_{\Sigma I1,s''}$. Nach *183 ist $t_{o1} \sim_{EI1,s''} t_{o2}$.
Jetzt gilt nach SE von $\equiv_{EI2}$

*184 $\sigma(t_1,t_2)[t/x^1_{s'}] \equiv_{EI2,d} \sigma(t_1',t_2')[t/x^1_{s'}]$
$\equiv_{EI2,d} \sigma(t_1',t_2')[t_{o1}/t_o]$ und

*185 $\sigma(t_1,t_2)[t'/x^1_{s'}] \equiv_{EI2,d} \sigma(t_1',t_2')[t'/x^1_{s'}]$
$\equiv_{EI2,d} \sigma(t_1',t_2')[t_{o2}/t_o]$.

Wir machen eine Fallunterscheidung über die Sorte $s'' \in \{\underline{nat}, \underline{stack}\}$.

c.1.2.2.iii.a) Für $s''=\underline{nat}$ folgt wegen $\equiv_{EI1,n} = \sim_{EI1,n}$ und Lemma 7 aus $t_{01} \sim_{EI1,n} t_{02}$ auch $t_{01} \equiv_{EI2,n} t_{02}$ und die SE von $\equiv_{EI2}$ liefert

*186 $\sigma(t_1', t_2')[t_{01}/t_0] \equiv_{EI2,d} \sigma(t_1', t_2')[t_{02}/t_0]$.

Der Rest ist Symmetrie und Transitivität von $\equiv_{EI2,d}$ zusammen mit *184, *186 und *185.

c.1.2.2.iii.b) Für $s''=\underline{stack}$ definieren wir

$\sigma(t_1'', t_2'') := \sigma(t_1', t_2')[x^2_{s''}/t_0]$. Es ist $\sigma(t_1'', t_2'') \in C2(s'')$ und $n(t_k'')=m$. Damit gilt

$$\begin{aligned}
\sigma(t_1, t_2)[t/x^1_{s'}] &\equiv_{EI2,d} \sigma(t_1', t_2')[t/x^1_{s'}] \\
&\equiv_{EI2,d} \sigma(t_1', t_2')[t_{01}/t_0] \\
&\ \underline{is}\ \ \sigma(t_1'', t_2'')[t_{01}/x^2_{s''}] \\
&\equiv_{EI2,d} \sigma(t_1'', t_2'')[t_{02}/x^2_{s''}] \\
&\ \underline{is}\ \ \sigma(t_1', t_2')[t_{02}/t_0] \\
&\equiv_{EI2,d} \sigma(t_1', t_2')[t'/x^1_{s'}] \\
&\equiv_{EI2,d} \sigma(t_1, t_2)[t'/x^1_{s'}].
\end{aligned}$$

Die mittlere i-Kongruenz gilt wegen $n(t_k'')=m$ nach Induktionsannahme c.1.2.2.ii).

Damit ist die Induktion unter c.1.2.2) geschlossen und mit c.1.2.1) und c.1.2.2) wurde gezeigt, daß *180 gilt.

Wenn wir also zeigen können, daß die Gleichungen E∑0.8 bis E∑0.12 die Bedingung *181 erfüllen, dann wissen wir nach *180, daß die Voraussetzungen von Lemma 18 erfüllt sind. Daraus können wir dann auf die t-Konsistenz von I2 auf I1 bzgl. der Sorten <u>array</u> und <u>stack</u> schließen.

c.2) s=<u>array</u>.

Seien $t,t' \in T_{\Sigma I1,a}$ mit $t \sim_{EI1,a} t'$.

Seien $(L,R) \in \{E\Sigma 0.8,\ldots,E\Sigma 0.12\}$ und $b,b' \in Bel(T_{\Sigma I2},L,R)$ mit t ist Subterm von $int_b(L)$ und $int_{b'}(L)$ entsteht aus $int_b(L)$ durch einmaliges ersetzen von t durch t'. Seien $int_b(R), int_{b'}(R) \in T_{\Sigma I1,s'}$ mit $s' \in \{\underline{nat},\underline{stack}\}$.

Dann ist je nach Gleichung t bzw. t' entweder Subterm von $b(x_a^1)$ (bzw. $b'(x_a^1)$) oder Subterm von $b(x_n^i)$ (bzw. $b'(x_n^i)$) für i=1,2. Wir berücksichtigen $E\Sigma 0.11$ zunächst nicht.

Da für alle $(L,R) \in \{E\Sigma 0.8,\ldots,E\Sigma 0.10,E\Sigma 0.12\}$ $int_b(R)$ und $int_{b'}(R)$ $\Sigma I1$-Terme sind, die sich höchstens darin unterscheiden, daß in $int_{b'}(R)$ t' an Stelle von t in $int_b(R)$ vorkommt, gilt nach SE von $\sim_{EI1}$ mit passendem $s' \in \{\underline{nat},\underline{stack}\}$

$$int_b(R) \sim_{EI1,s'} int_{b'}(R).$$

Für Gleichung $E\Sigma 0.11$ gilt speziell R <u>is</u> ∅ und damit

$$\emptyset \ \underline{is}\ int_b(R) \sim_{EI1,n} int_{b'}(R)\ \underline{is}\ \emptyset$$

unabhängig von den Belegungen b und b'.

Damit ist *181 für die Sorte <u>array</u> gezeigt.

c.3) s=<u>stack</u>.

Seien $q_1,q_2 \in T_{\Sigma I1,a}$, $r_1,r_2 \in T_{\Sigma I1,n}$ und es gelte

$$ST(q_1,r_1) \sim_{EI1,s} ST(q_2,r_2),$$

woraus nach Lemma K.STACK.I1 sofort $r_1 \equiv_{EI1,n} r_2$ folgt.

Nach Lemma N.I1, Lemma 9 und SE von $\sim_{EI1}$ gilt für alle $q \in T_{\Sigma I1,a}$ und für alle $r \in T_{\Sigma I1,n}$

$(\exists j \in \omega_0)(r \equiv_{EI1,n} S^j(\emptyset) \ \wedge\ ST(q,r) \sim_{EI1,s} ST(q,S^j(\emptyset)))$.

Daher ist es zulässig, nur noch Terme $ST(q_i,r_i)$ mit $r_i \in \{S^j(\emptyset) \mid j \in \omega_0\}$ für i=1,2 zu betrachten.

Wir zeigen *181 durch Fallunterscheidung über die Gleichungen $(L,R)\epsilon\{E\Sigma 0.8,\ldots,E\Sigma 0.12\}$.

c.3.1) (L,R) <u>is</u> $PU(ST(x_a^1,x_n^1),x_n^2) = ST(AS(x_a^1,S(x_n^1),x_n^2),S(x_n^1))$.

Seien $b,b'\epsilon Bel(T_{\Sigma I2},L,R)$ mit $b(x_a^1)$ <u>is</u> q_1, $b'(x_a^1)$ <u>is</u> q_2, $b(x_n^1)$ <u>is</u> r_1, $b'(x_n^1)$ <u>is</u> r_2 und $b(x_n^2)$ <u>is</u> $b'(x_n^2)$, $b(x_n^2)\epsilon T_{\Sigma I1,n}$. Sei r_1 <u>is</u> $S^j(\emptyset)$ mit $j\epsilon\omega_0$. Dann gilt nach Lemma K.STACK.I1

r_2 <u>is</u> $S^j(\emptyset)$ und

*187 $$j>0 \Rightarrow ((\forall i\epsilon\{1,\ldots,j\})(IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \wedge AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset)))).$$

Wir müssen zeigen, daß

$$ST(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset)) \sim_{EI1,s} ST(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset))$$

gilt; dazu unterscheiden wir über j.

c.3.1.1) $j=0$.

Nach Punkt 1) von Korollar EQ.NAT.I1 gilt $EQN(S(\emptyset),S(\emptyset)) \equiv_{EI1,d} tt$. Wir können Gleichung E1.8 und E1.11 anwenden und erhalten

$$IN(AS(q_1,S(\emptyset),b(x_n^2)),S(\emptyset)) \equiv_{EI1,d} tt \equiv_{EI1,d} IN(AS(q_2,S(\emptyset),b(x_n^2)),S(\emptyset))$$ und

$$AC(AS(q_1,S(\emptyset),b(x_n^2)),S(\emptyset)) \equiv_{EI1,n} b(x_n^2) \equiv_{EI1,n} AC(AS(q_2,S(\emptyset),b(x_n^2)),S(\emptyset)).$$

Mit $j'=1$ folgt daraus nach Lemma K.STACK.I1

$$ST(AS(q_1,S(\emptyset),b(x_n^2)),S(\emptyset)) \sim_{EI1,s} ST(AS(q_2,S(\emptyset),b(x_n^2)),S(\emptyset)).$$

c.3.1.2) $j>0$.

Nach Punkt 1) von Korollar EQ.NAT.I1 gilt $EQN(S^{j+1}(\emptyset),S^{j+1}(\emptyset)) \equiv_{EI1,d} tt$. Wir können die Gleichungen E1.8 und E1.11 anwenden und erhalten

*188 $$IN(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset)) \equiv_{EI1,d} tt \equiv_{EI1,d} IN(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset))$$ und

*189 $$AC(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset)) \equiv_{EI1,n} b(x_n^2) \equiv_{EI1,n} AC(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset)).$$

Nach Lemma K.NAT.I1 gilt

$(\forall i \in \{1,\ldots,j\})(EQN(S^{j+1}(\emptyset),S^{i}(\emptyset)) \equiv_{EI1,d} ff)$,

d.h. wir können Gleichung E1.9 j-mal anwenden und daher gilt

*190 $(\forall i \in \{1,\ldots,j\})$

$$
\begin{aligned}
(IN(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)) &\equiv_{EI1,d} IN(q_1,S^{i}(\emptyset)) \\
&\equiv_{EI1,d} IN(q_2,S^{i}(\emptyset)) \\
&\equiv_{EI1,d} \\
IN(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)) &\ ),
\end{aligned}
$$

wobei die erste und letzte i-Kongruenz jeweils nach Gleichung E1.9 und die mittlere i-Kongruenz nach *187 gilt.

Wir können auch Gleichung E1.12 j-mal anwenden und erhalten

*191 $(\forall i \in \{1,\ldots,j\})$

$$
\begin{aligned}
(AC(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)) &\equiv_{EI1,n} AC(q_1,S^{i}(\emptyset)) \\
&\equiv_{EI1,n} AC(q_2,S^{i}(\emptyset)) \\
&\equiv_{EI1,n} \\
AC(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)) &\ ),
\end{aligned}
$$

wobei die erste und letzte i-Kongruenz jeweils nach Gleichung E1.12 und die mittlere i-Kongruenz nach *187 gilt.

Mit *188 bis *191 haben wir gezeigt

$$
\begin{aligned}
&(j+1>0 \;\Rightarrow\; (\forall i \in \{1,\ldots,j+1\}) \\
&(IN(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)) \equiv_{EI1,d} IN(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)) \;\wedge \\
&\ AC(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)) \equiv_{EI1,n} AC(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{i}(\emptyset)))),
\end{aligned}
$$

woraus nach Lemma K.STACK.I1 folgt

$$ST(AS(q_1,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset)) \sim_{EI1,s} ST(AS(q_2,S^{j+1}(\emptyset),b(x_n^2)),S^{j+1}(\emptyset)).$$

Damit gilt unter den angegebenen Voraussetzungen

$$int_b(R) \sim_{EI1,s} int_{b'}(R) \qquad \text{für} \quad (L,R) \ \underline{\text{is}}\ E\Sigma 0.8.$$

c.3.2) (L,R) is $POP(ST(x_a^1,\emptyset)) = ST(x_a^1,\emptyset)$.

Seien $b,b' \in Bel(T_{\Sigma I2},L,R)$ mit $b(x_a^1)$ is q_1 und $b'(x_a^1)$ is q_2 und r_1 is $\emptyset$, r_2 is $\emptyset$. Für die Subterme $ST(q_1,\emptyset)$ von $int_b(L)$ und $ST(q_2,\emptyset)$ von $int_{b'}(L)$ gilt nach Voraussetzung von c.3)

$int_b(R)$ $\underline{is}$ $ST(q_1,\emptyset) \sim_{EI1,s} ST(q_2,\emptyset)$ $\underline{is}$ $int_{b'}(R)$.

c.3.3) (L,R) $\underline{is}$ $POP(ST(x_a^1,S(x_n^1))) = ST(x_a^1,x_n^1)$.

Seien $b,b' \in Bel(T_{\sum I2},L,R)$ mit $b(x_a^1)$ $\underline{is}$ q_1, $b'(x_a^1)$ $\underline{is}$ q_2 und r_1,r_2 $\underline{is}$ $S^{j+1}(\emptyset)$ mit $j \in \omega_0$, d.h. $b(x_n^1),b'(x_n^1)$ $\underline{is}$ $S^j(\emptyset)$. Dann gilt für die Subterme $ST(q_1,S^{j+1}(\emptyset))$ von $int_b(L)$ und $ST(q_2,S^{j+1}(\emptyset))$ von $int_{b'}(L)$

$$ST(q_1,S^{j+1}(\emptyset)) \sim_{EI1,s} ST(q_2,S^{j+1}(\emptyset))$$

nach Voraussetzung von c.3). Nach Lemma K.STACK.I1 folgt

$$(\forall i \in \{1,\dots,j+1\})(IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \wedge AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset)))$$

und natürlich erst recht

$$(j>0 \Rightarrow ((\forall i \in \{1,\dots,j\})(IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \wedge AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset))))).$$

Daraus folgt nach Lemma K.STACK.I1 wiederum

$int_b(R)$ $\underline{is}$ $ST(q_1,S^j(\emptyset)) \sim_{EI1,s} ST(q_2,S^j(\emptyset))$ $\underline{is}$ $int_{b'}(R)$.

c.3.4) (L,R) $\underline{is}$ $TOP(ST(x_a^1,\emptyset)) = \emptyset$.

Seien $b,b' \in Bel(T_{\sum I2},L,R)$ mit $b(x_a^1)$ $\underline{is}$ q_1, $b'(x_a^1)$ $\underline{is}$ q_2, r_1 $\underline{is}$ $\emptyset$ und r_2 $\underline{is}$ $\emptyset$. Dann gilt

$int_b(R)$ $\underline{is}$ $\emptyset \sim_{EI1,n} \emptyset$ $\underline{is}$ $int_{b'}(R)$.

c.3.5) (L,R) $\underline{is}$ $TOP(ST(x_a^1,S(x_n^1))) = AC(x_a^1,S(x_n^1))$.

Seien $b,b' \in Bel(T_{\sum I2},L,R)$ mit $b(x_a^1)$ $\underline{is}$ q_1, $b'(x_a^1)$ $\underline{is}$ q_2 und seien r_1,r_2 $\underline{is}$ $S^{j+1}(\emptyset)$ mit $j \in \omega_0$, d.h. $b(x_n^1),b'(x_n^1)$ $\underline{is}$ $S^j(\emptyset)$. Nach Lemma K.STACK.I1 gilt wegen $j+1>0$

$$(\forall i \in \{1,\dots,j+1\})(IN(q_1,S^i(\emptyset)) \equiv_{EI1,d} IN(q_2,S^i(\emptyset)) \wedge AC(q_1,S^i(\emptyset)) \equiv_{EI1,n} AC(q_2,S^i(\emptyset))).$$

Also gilt speziell

$$AC(q_1,S^{j+1}(\emptyset)) \equiv_{EI1,n} AC(q_2,S^{j+1}(\emptyset)).$$

Mit Lemma 9 folgt daraus

$int_b(R)$ $\underline{is}$ $AC(q_1,S^{j+1}(\emptyset)) \sim_{EI1,n} AC(q_2,S^{j+1}(\emptyset))$ $\underline{is}$ $int_{b'}(R)$.

Damit ist die Fallunterscheidung unter c.3) beendet. Es gilt *181 für die Sorte stack, d.h. daß nach *180 die Bedingungen von Lemma 18 erfüllt sind. Somit ist I2 t-konsistent auf I1 bzgl. stack. Aus a) bis c) folgt, daß I2 t-konsistent auf I1 ist. $\square$

Jetzt wissen wir, daß IMPL die beiden Korrektheitskriterien ii) und iii) der Definition 44 erfüllt.

Korollar T.ERW.I1

Beh.: I1 ist t-Erweiterung von SP1.

Bew.:

Der Beweis verläuft völlig analog zum Beweis von Korollar T.ERW.SP0. $\square$

Korollar T.ANR.I2

Beh.: I2 ist t-Anreicherung von I1.

Bew.:

Analog zum Beweis von Korollar T.ERW.SP0 zeigt man, daß I2 t-Erweiterung von I1 ist. Nach Definition gilt

$$SI2 = S \cup S0 \cup S1 = \{\underline{dis}, \underline{nat}, \underline{array}, \underline{stack}\} = SI1,$$

also ist I2 speziell t-Anreicherung von I1. $\square$

Es bleibt zu zeigen, daß IMPL R-korrekt ist. Die Hauptarbeit wird dabei darin bestehen, die Relationen $\sim_{E \cup E0,s}$ und $\sim_{EI2,s}$ zu charakterisieren. Zunächst machen wir jedoch die Beobachtung, daß die i- und t-Kongruenzrelation von SP0 übereinstimmen.

Lemma KAT.SP0

Beh.: SP0 ist kategorisch, d.h. $(\forall s \in \{\underline{dis}, \underline{nat}, \underline{stack}\})$

$$(\forall t_1, t_2 \in T_{\Sigma \cup \Sigma 0, s})(t_1 \equiv_{E \cup E0,s} t_2 \iff t_1 \sim_{E \cup E0,s} t_2).$$

Bew.:

"=>" Gilt nach Lemma 9.

"<=" Durch Fallunterscheidung über die Sorten $s \in S \cup S0$.

a) $s=$dis.

Für dis gilt die Behauptung nach Definition von $\sim_{E \cup E0}$. Siehe auch die note nach Lemma 2.2.4. in [HR 79].

b) $s=$nat.

Der Beweis zu diesem Unterfall verläuft völlig analog zu Punkt b.2) im Beweis von Lemma TK.I1.

c) $s=$stack.

Seien $t_1, t_2 \in T_{\Sigma \cup \Sigma 0, s}$ mit $t_1 \sim_{E \cup E0, s} t_2$.

Seien $n, m, j_1, \ldots, j_n, k_1, \ldots, k_m \in \omega_0$ die nach Lemma S.SP0 und N.SP0 existierenden Zahlen, für die gilt

$$t_1 \equiv_{E \cup E0, s} PU(\ldots PU(NS, S^{j_1}(\emptyset)), \ldots, S^{j_n}(\emptyset)) =: t_1'$$

$$t_2 \equiv_{E \cup E0, s} PU(\ldots PU(NS, S^{k_1}(\emptyset)), \ldots, S^{k_m}(\emptyset)) =: t_2'.$$

Wir zeigen indirekt, daß $m=n$ und $(\forall i \in \{1, \ldots, n\})(j_i = k_i)$ gilt. Dazu benötigen wir die Hilfsbehauptung

$$^*192 \quad (\forall n \in \omega_0)(\forall r_1, \ldots, r_n \in T_{\Sigma \cup \Sigma 0, n})(\forall m \in \omega_0)(m \leq n \Rightarrow$$
$$\underbrace{POP(..(POP}_{m\text{-mal}}(PU(..PU(NS, r_1), .., r_n))..) \equiv_{E \cup E0, s} PU(..PU(NS, r_1), .., r_{n-m})),$$

die man leicht durch Induktion über n und m zeigen kann, was wir aber hier aus Platzgründen unterlassen.

Angenommen, es ist $n \neq m$ bzw. o.B.d.A. $m<n$. Dann gilt

$$EQS(\underbrace{POP(\ldots POP}_{m\text{-mal}}(t_1)\ldots), \underbrace{POP(\ldots POP}_{m\text{-mal}}(t_2)\ldots)) \equiv_{E \cup E0, d}$$

$$EQS(\underbrace{POP(\ldots POP}_{m\text{-mal}}(t_1')\ldots), \underbrace{POP(\ldots POP}_{m\text{-mal}}(t_2')\ldots)) \equiv_{E \cup E0, d}$$

$$EQS(PU(\ldots PU(NS, S^{j_1}(\emptyset)), \ldots, S^{j_{n-m}}(\emptyset)), NS) \equiv_{E \cup E0, d} ff$$

nach SE von $\equiv_{E \cup E0}$, ^{*192} und nach Gleichung E0.3, die wir anwenden können, da $n-m>0$ und somit

$$PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_{n-m}}(\emptyset)) \;\cancel{is}\; NS$$

gilt. Andererseits gilt

$$EQS(\underbrace{POP(\ldots POP}_{m\text{-mal}}(t_2)\ldots),\underbrace{POP(\ldots POP}_{m\text{-mal}}(t_2)\ldots)) \equiv_{E\cup E0,d}$$

$$EQS(\underbrace{POP(\ldots POP}_{m\text{-mal}}(t_2')\ldots),\underbrace{POP(\ldots POP}_{m\text{-mal}}(t_2')\ldots)) \equiv_{E\cup E0,d}$$

$$EQS(NS,NS) \equiv_{E\cup E0,d} tt$$

nach SE von $\equiv_{E\cup E0}$, *192 und Gleichung E0.1.

Wir haben also einen Kontext

$$ct:=EQS(\underbrace{POP(\ldots POP}_{m\text{-mal}}(x_s^1)\ldots),\underbrace{POP(\ldots POP}_{m\text{-mal}}(t_2)\ldots))\in C_{\Sigma\cup\Sigma 0}(s,dis)$$

gefunden, sodaß wegen der Konsistenz von SP0 gilt

$$ct[t_1/x_s^1] \not\equiv_{E\cup E0,d} ct[t_2/x_s^1].$$

Dies ist ein Widerspruch zur Voraussetzung $t_1 \sim_{E\cup E0,s} t_2$; also muß $n=m$ sein.

Angenommen, es gibt ein $i\in\{1,\ldots,n\}$, sodaß $k_i \neq j_i$. Dann gilt

$$EQN(TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(t_1)\ldots)),TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(t_2)\ldots))) \equiv_{E\cup E0,d}$$

$$EQN(TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(t_1')\ldots)),TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(t_2')\ldots))) \equiv_{E\cup E0,d}$$

$$EQN(TOP(PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_i}(\emptyset))),$$

$$TOP(PU(\ldots PU(NS,S^{k_1}(\emptyset)),\ldots,S^{k_i}(\emptyset)))) \equiv_{E\cup E0,d}$$

$$EQN(S^{j_i}(\emptyset),S^{k_i}(\emptyset)) \equiv_{E\cup E0,d} ff$$

nach SE von $\equiv_{E\cup E0}$, *192, Gleichung E0.7 und Punkt 2) von Korollar K.NAT.SP0. Außerdem gilt

$$EQN(TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(t_1)\ldots)),TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(t_1)\ldots))) \equiv_{E\cup E0,d} tt$$

nach Reflexivität von $\equiv_{E\cup E0,n}$ und nach Korollar 10.

Damit haben wir einen Kontext

$$ct:=EQN(TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(t_1)\ldots)),TOP(\underbrace{POP(\ldots POP}_{(n-i)\text{-mal}}(x_s^1)\ldots)))\in C_{\Sigma\cup\Sigma 0}(s,dis)$$

gefunden, sodaß wegen der Konsistenz von SP0 gilt

$$ct[t_1/x_s^1] \not\equiv_{E\cup E0,d} ct[t_2/x_s^1].$$

Dies ist ein Widerspruch zur Voraussetzung $t_1 \sim_{E\cup E0,s} t_2$.

Also muß gelten $(\forall i\in\{1,\ldots,n\})(k_i=j_i)$.

Damit gilt dann

$$t_1 \equiv_{E \cup E0,s} PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_n}(\emptyset))$$
$$\underline{is} \quad PU(\ldots PU(NS,S^{k_1}(\emptyset)),\ldots,S^{k_m}(\emptyset))$$
$$\equiv_{E \cup E0,s} t_2, \text{ q.e.d..}$$

Aus a), b) und c) folgt, daß SP0 kategorisch ist. □

Als nächstes zeigen wir, daß jeder kanonische stack-Term in I2 i-kongruent zu einem pointer/array-Paar aus kanonischen array- und nat-Termen ist.

Lemma NF.I2

Beh.: $(\forall n,j_1,\ldots,j_n \in \omega_0)$

$$(PUSH(\ldots PUSH(NEWST,S^{j_1}(\emptyset)),\ldots,S^{j_n}(\emptyset)) \equiv_{EI2,s}$$
$$ST(AS(\ldots AS(NA,S(\emptyset),S^{j_1}(\emptyset)),\ldots,S^n(\emptyset),S^{j_n}(\emptyset)),S^n(\emptyset)))$$

Bew.:

Der Beweis erfolgt durch Induktion über $n \in \omega_0$.

i) Induktionsanfang.

Sei $n \in \omega_0$, n=0. Dann gilt

$PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_n}(\emptyset))$ $\underline{is}$ NS und

$ST(AS(..AS(NA,S(\emptyset),S^{j_1}(\emptyset)),..,S^n(\emptyset),S^{j_n}(\emptyset)),S^n(\emptyset))$ $\underline{is}$ $ST(NA,\emptyset)$.

Nach Gleichung $E\Sigma0.7$ gilt $\quad NS \equiv_{EI2,s} ST(NA,\emptyset)$.

ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $n \in \omega_0$.

iii) Induktionsschluß.

Seien $j_1,\ldots,j_n,j_{n+1} \in \omega_0$. Dann gilt

$$PU(PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_n}(\emptyset)),S^{j_{n+1}}(\emptyset)) \equiv_{EI2,s}$$
$$PU(ST(AS(\ldots AS(NA,S(\emptyset),S^{j_1}(\emptyset)),\ldots,S^n(\emptyset),S^{j_n}(\emptyset)),S^n(\emptyset)),S^{j_{n+1}}(\emptyset)) \equiv_{EI2,s}$$
$$ST(AS(AS(\ldots AS(NA,S(\emptyset),S^{j_1}(\emptyset)),..,S^n(\emptyset),S^{j_n}(\emptyset)),S^{n+1}(\emptyset),S^{j_{n+1}}(\emptyset)),S^{n+1}(\emptyset))$$

nach Induktionsannahme ii), SE von $\equiv_{EI2}$ und Gleichung $E\Sigma0.8$.

Damit ist die Induktion geschlossen. □

Die Charakterisierung des stack-Anteils der t-Kongruenz von I1 in Lemma K.STACK.I1 gilt entsprechend für I2.

Lemma K.STACK.I2

Beh.: $(\forall q_1,q_2 \in T_{\Sigma I2,a})(\forall r_1,r_2 \in T_{\Sigma I2,n})$

$(ST(q_1,r_1) \sim_{EI2,s} ST(q_2,r_2) \quad <=>$

$($ 1) $r_1 \equiv_{EI2,n} r_2 \quad \wedge$

2) $((\exists j \in \omega_0)(r_1 \equiv_{EI2,n} S^j(\emptyset) \wedge j>0) => ((\forall i \in \{1,...,j\})$

$($ 2.a) $IN(q_1,S^i(\emptyset)) \equiv_{EI2,d} IN(q_2,S^i(\emptyset)) \quad \wedge$

2.b) $AC(q_1,S^i(\emptyset)) \equiv_{EI2,n} AC(q_2,S^i(\emptyset)) \quad)))))$

Bew.:

Seien $q_1,q_2 \in T_{\Sigma I2,a}$ und $r_1,r_2 \in T_{\Sigma I2,n}$. Nach Lemma A.I2 und N.I2 gibt es $q_1',q_2' \in T_{\Sigma I2,a}$ und $j_1,j_2 \in \omega_0$, sodaß für i=1,2 gilt

$q_i \equiv_{EI2,a} q_i' \quad$ und $\quad r_i \equiv_{EI2,n} S^{j_i}(\emptyset)$.

Daraus folgt nach SE von $\equiv_{EI2}$ für i=1,2

*193 $ST(q_i,r_i) \equiv_{EI2,s} ST(q_i',S^{j_i}(\emptyset))$,

wobei $ST(q_i',S^{j_i}(\emptyset)) \in T_{\Sigma I1,s}$.

Nach Lemma 9 folgt aus *193 für i=1,2

*194 $ST(q_i,r_i) \sim_{EI2,s} ST(q_i',S^{j_i}(\emptyset))$.

Wegen der Transitivität von $\sim_{EI2}$ gilt mit *194 auch

*195 $ST(q_1,r_1) \sim_{EI2,s} ST(q_2,r_2) \quad <=>$

$ST(q_1',S^{j_1}(\emptyset)) \sim_{EI2,s} ST(q_2',S^{j_2}(\emptyset))$.

Aufgrund der t-Konsistenz von I2 auf I1 und nach Lemma 10 gilt

*196 $ST(q_1',S^{j_1}(\emptyset)) \sim_{EI2,s} ST(q_2',S^{j_2}(\emptyset)) \quad <=>$

$ST(q_1',S^{j_1}(\emptyset)) \sim_{EI1,s} ST(q_2',S^{j_2}(\emptyset))$.

Lemma K.STACK.I1 liefert

*197 $ST(q_1',S^{j_1}(\emptyset)) \sim_{EI1,s} ST(q_2',S^{j_2}(\emptyset)) \quad <=>$

*198 $\{($ 3) $S^{j_1}(\emptyset) \equiv_{EI1,n} S^{j_2}(\emptyset) \quad \wedge$

4) $((\exists j \in \omega_0)(S^{j_1}(\emptyset) \equiv_{EI1,n} S^j(\emptyset) \wedge j>0) \quad =>$

$(\forall i \in \{1,...,j\})(IN(q_1',S^i(\emptyset)) \equiv_{EI1,d} IN(q_2',S^i(\emptyset))$

$AC(q_1',S^i(\emptyset)) \equiv_{EI1,n} AC(q_2',S^i(\emptyset)) \quad)))$.

An dieser Stelle benötigen wir die

Hilfsbehauptung: I2 ist kategorisch bzgl. nat, d.h. es gilt

$$(\forall t_1,t_2 \in T_{\Sigma I2,n})(t_1 \equiv_{EI2,n} t_2 \Leftrightarrow t_1 \sim_{EI2,n} t_2).$$

Der Beweis verläuft analog zum Beweis der Kategorizität von SP1 bzgl. nat unter b.1) und b.2) des Beweises von Lemma TK.I1.

Damit haben wir für $s\in\{\underline{dis},\underline{nat}\}$ und $t_1,t_2 \in T_{\Sigma I2,s}$ folgende Situation

$$\begin{array}{ccc} t_1 \equiv_{EI1,s} t_2 & \overset{1}{\Longleftrightarrow} & t_1 \sim_{EI1,s} t_2 \\ {}^{2}\Downarrow & & \Uparrow_{3} \\ t_1 \equiv_{EI2,s} t_2 & \overset{4}{\Longleftrightarrow} & t_1 \sim_{EI2,s} t_2 \end{array} .$$

Es gilt

1 weil I1 kategorisch bzgl. dis und nat ist, wobei letzteres unter b) des Beweises von Lemma TK.I2 gezeigt wurde,

2 nach Lemma 7,

3 nach Lemma 10 und

4 nach unserer Hilfsbehauptung und weil I2 per definitionem kategorisch bzgl. dis ist.

Also gilt

$$(\forall s\in\{\underline{dis},\underline{nat}\})(\forall t_1,t_2 \in T_{\Sigma I2,s})(t_1 \equiv_{EI1,s} t_2 \Leftrightarrow t_1 \equiv_{EI2,s} t_2)$$

und damit auch

$$\begin{array}{l} *199 \quad (*198 \Leftrightarrow \\ \qquad (\ 3')\quad S^{j_1}(\emptyset) \equiv_{EI2,n} S^{j_2}(\emptyset) \quad \wedge \\ \qquad\quad 4')\quad ((\exists j\in\omega_0)(S^{j_1}(\emptyset) \equiv_{EI2,n} S^{j}(\emptyset) \wedge j>0) \quad \Rightarrow \\ \qquad\qquad (\forall i\in\{1,\dots,j\})(IN(q_1',S^i(\emptyset)) \equiv_{EI2,d} IN(q_2',S^i(\emptyset)) \wedge \\ \qquad\qquad\qquad AC(q_1',S^i(\emptyset)) \equiv_{EI2,n} AC(q_2',S^i(\emptyset)))))). \end{array}$$

In 3') und 4') können wir für i=1,2 q_i' und $S^{j_i}(\emptyset)$ durch die $\equiv_{EI2}$-äquivalenten q_i und r_i ersetzen. Die Gültigkeit der

Behauptung folgt dann aus *195, *196, *197 und *199 und dem gerade Gesagten. $\square$

Das nächste Lemma besagt, daß in I2 zwei stacks, deren pointer an die gleiche Stelle zeigen und deren Inhalt "unter" dem pointer gleich ist, als verhaltensgleich angesehen werden, unabhängig von ihrem Inhalt "über" dem pointer.

<u>Lemma ST.I2</u>

<u>Beh.:</u> $(\forall k,n,j_1,\dots,j_n \in \omega_0)(k \le n \Rightarrow$

$$ST(AS(\dots AS(NA,S(\emptyset),S^{j_1}(\emptyset)),\dots,S^n(\emptyset),S^{j_n}(\emptyset)),S^k(\emptyset)) \sim_{EI2,s}$$

$$ST(AS(\dots AS(NA,S(\emptyset),S^{j_1}(\emptyset)),\dots,S^k(\emptyset),S^{j_k}(\emptyset)),S^k(\emptyset))\quad)$$

<u>Bew.:</u>

Durch Fallunterscheidung über $k \in \omega_0$.

Seien $k,n,j_1,\dots,j_n \in \omega_0$. Wir kürzen ab:

$a_n := AS(\dots AS(NA,S(\emptyset),S^{j_1}(\emptyset)),\dots,S^n(\emptyset),S^{j_n}(\emptyset))$ und

$a_k := AS(\dots AS(NA,S(\emptyset),S^{j_1}(\emptyset)),\dots,S^k(\emptyset),S^{j_k}(\emptyset))$.

a) $k=n$.

Es ist $ST(a_n,S^k(\emptyset))$ <u>is</u> $ST(a_k,S^k(\emptyset))$ und die Behauptung gilt hier nach Reflexivität von $\sim_{EI2,s}$.

b) $k=0$, $k<n$.

Es gilt $S^k(\emptyset)$ <u>is</u> $\emptyset$ und nach Lemma K.NAT.I2 und Korollar 13 gilt $(\nexists j \in \omega_0)(S^k(\emptyset) \equiv_{EI2,n} S^j(\emptyset) \wedge j>0)$.

Daher ist Punkt 2) von Lemma K.STACK.I2 nicht anwendbar aber nach Punkt 1) desselben gilt $ST(a_n,\emptyset) \sim_{EI2,s} ST(a_k,\emptyset)$.

c) $0<k<n$.

Man kann leicht durch Induktion zeigen, daß

$$(\forall i \in \{1,\dots,k\})(IN(a_n,S^i(\emptyset)) \equiv_{EI2,d} tt \equiv_{EI2,d} IN(a_k,S^i(\emptyset))$$
$$AC(a_n,S^i(\emptyset)) \equiv_{EI2,n} S^{j_i}(\emptyset) \equiv_{EI2,n} AC(a_k,S^i(\emptyset))\ )$$

gilt, jedoch unterlassen wir den Beweis aus Platzgründen.

Damit sind die Bedingungen von Lemma K.STACK.I2 erfüllt und es gilt $ST(an, S^k(\emptyset)) \sim_{EI2,s} ST(a_k, S^k(\emptyset))$. $\square$

Wir zeigen jetzt, daß zwei kanonische $\Sigma \cup \Sigma 0$-stack-Terme genau dann unter EQS gleich sind, wenn sie identisch sind, und zwar sowohl in SP0 als auch in I2.

Lemma K.S.SP0

Beh.: $(\forall m_1, m_2, j_1^1, \ldots, j_{m_1}^1, j_1^2, \ldots, j_{m_2}^2 \in \omega_0)$

$$\begin{aligned}
(\ 1)\quad & ((m_1=m_2 \wedge (\forall i \in \{1,\ldots,m_1\})(j_i^1=j_i^2)) \Rightarrow \\
& \quad EQS(PU(\ldots PU(NS, S^{j_1^1}(\emptyset)), \ldots, S^{j_{m_1}^1}(\emptyset)), \\
& \quad\quad PU(\ldots PU(NS, S^{j_1^2}(\emptyset)), \ldots, S^{j_{m_2}^2}(\emptyset))) \equiv_{E \cup E0,d} tt) \quad \wedge \\
2)\quad & (((m_1=m_2 \wedge (\exists i \in \{1,\ldots,m_1\})(j_i^1 \neq j_i^2)) \oplus m_1 \neq m_2) \Rightarrow \\
& \quad EQS(PU(\ldots PU(NS, S^{j_1^1}(\emptyset)), \ldots, S^{j_{m_1}^1}(\emptyset)), \\
& \quad\quad PU(\ldots PU(NS, S^{j_1^2}(\emptyset)), \ldots, S^{j_{m_2}^2}(\emptyset))) \equiv_{E \cup E0,d} ff) \quad)
\end{aligned}$$

Bew.:

ad 1) Durch Induktion über $m_1 \in \omega_0$.

1.i) Induktionsanfang.

Sei $m_1=0$ und $m_2 \in \omega_0$ mit $m_1=m_2$.

Dann ist $\{1,\ldots,m_1\}=\emptyset$ und für $i=1,2$

$PU(\ldots PU(NS, S^{j_1^i}(\emptyset)), \ldots, S^{j_{m_i}^i}(\emptyset))$ is NS.

Nach Gleichung E0.1 gilt $EQS(NS,NS) \equiv_{E \cup E0,d} tt$.

1.ii) Induktionsannahme.

Die Behauptung gelte für ein beliebiges, aber festes $m_1 \in \omega_0$ und für alle $m_2 \in \omega_0$.

1.iii) Induktionsschluß.

Seien $m_2, j_1^1, \ldots, j_{m_1}^1, j_{m_1+1}^1, j_1^2, \ldots, j_{m_2}^2 \in \omega_0$ und es gelte $m_1+1=m_2$ und $(\forall i \in \{1,\ldots,m_1+1\})(j_i^1=j_i^2)$.

Für $i=1,2$ sei $sti := PU(\ldots PU(NS, S^{j_1^i}(\emptyset)), \ldots, S^{j_{m_i}^i}(\emptyset))$.

Es gilt st2 $\cancel{\underline{\text{is}}}$ NS und $PU(st1,S^{j^1_{m_1+1}}(\emptyset))$ $\cancel{\underline{\text{is}}}$ NS.

Nach Lemma K.NAT.SP0 gilt $EQN(S^{j^1_{m_1+1}}(\emptyset),S^{j^2_{m_2}}(\emptyset)) \equiv_{E\cup E0,d} tt$,

d.h. wir können Gleichung E0.4 anwenden und erhalten

$$EQS(PU(st1,S^{j^1_{m_1+1}}(\emptyset)),st2) \equiv_{E\cup E0,d}$$

$$EQS(st1,PU(\ldots PU(NS,S^{j^2_1}(\emptyset)),\ldots,S^{j^2_{m_2-1}}(\emptyset))) \equiv_{E\cup E0,d} tt,$$

wobei die letzte i-Kongruenz wegen $m_1=m_2-1$ und

$(\forall i\in\{1,\ldots,m_1\})(j^1_i=j^2_i)$ nach Induktionsannahme 1.ii) gilt.

Damit ist die Induktion unter ad 1) geschlossen.

ad 2) Durch Induktion über $m_1\in\omega_0$.

2.i) Induktionsanfang.

Sei $m_1=0$ und $m_2,j^2_1,\ldots,j^2_{m_2}\in\omega_0$ mit $m_2>0$.

Es sei $sti:=PU(\ldots PU(NS,S^{j^i_1}(\emptyset)),\ldots,S^{j^i_{m_i}}(\emptyset))$ für $i=1,2$.

Dann ist st1 $\underline{\text{is}}$ NS und st2 $\cancel{\underline{\text{is}}}$ NS. Nach Gleichung E0.2

gilt daher $EQS(st1,st2) \equiv_{E\cup E0,d} ff$.

Für $m_1=m_2=0$ ist die Prämisse von 2) falsch, denn es gilt

$(\nexists i\in\{1,\ldots,m_1\})(j^1_i\neq j^2_i)$, und die Implikation ist daher

richtig.

2.ii) Induktionsannahme.

Die Behauptung gilt für ein beliebiges, aber festes $m_1\in\omega_0$

und für alle $m_2\in\omega_0$.

2.iii) Induktionsschluß.

Seien $m_2,j^1_1,\ldots,j^1_{m_1},j^1_{m_1+1},j^2_1,\ldots,j^2_{m_2}\in\omega_0$. Es gilt

$PU(st1,S^{j^1_{m}+1}(\emptyset))$ $\cancel{\underline{\text{is}}}$ NS. Wir unterscheiden zwei Fälle.

2.iii.a) $m_1+1=m_2 \wedge (\exists i\in\{1,\ldots,m_2\})(j^1_i\neq j^2_i)$.

Dann ist auch st2 $\cancel{\underline{\text{is}}}$ NS.

Sei $V:=\{i\in\omega_0 \mid i\in\{1,\ldots,m_1+1\} \wedge j^1_i\neq j^2_i\}$. Es ist $V\neq\emptyset$.

Sei $k\in V$ mit $(\forall l\in V)(l\leq k)$, d.h. k ist größtes Element in V.

Dann gilt $(\forall l\in\{k+1,\ldots,m_1+1\})(j^1_l=j^2_l)$ und nach Lemma

K.NAT.SP0 gilt

$(\forall l\in\{k+1,\ldots,m_1+1\})(\mathrm{EQN}(S^{j^1_l}(\emptyset),S^{j^2_l}(\emptyset)) \equiv_{E\cup E0,d} tt)$.

Wir können daher Gleichung E0.4 (m_1-k+1)-mal anwenden und erhalten

$$\begin{aligned}&\mathrm{EQS}(\mathrm{PU}(st1,S^{j^1_{m_1+1}}(\emptyset)),st2) &&\equiv_{E\cup E0,d}\\ &\mathrm{EQS}(\mathrm{PU}(\ldots\mathrm{PU}(\mathrm{NS},S^{j^1_1}(\emptyset)),\ldots,S^{j^1_k}(\emptyset)),\\ &\quad\mathrm{PU}(\ldots\mathrm{PU}(\mathrm{NS},S^{j^2_1}(\emptyset)),\ldots,S^{j^2_k}(\emptyset))) &&\equiv_{E\cup E0,d} ff,\end{aligned}$$

wobei die letzte i-Kongruenz nach Gleichung E0.5 gilt, die wir anwenden können, weil mit $j^1_k \neq j^2_k$ nach Lemma K.NAT.SP0 auch $\mathrm{EQN}(S^{j^1_k}(\emptyset),S^{j^2_k}(\emptyset)) \equiv_{E\cup E0,d} ff$ gilt.

2.iii.b) $m_1+1\neq m_2$.

2.iii.b.1) $m_2=0$, d.h. st2 <u>is</u> NS.

Dann gilt nach Gleichung E0.3

$$\mathrm{EQS}(\mathrm{PU}(st1,S^{j^1_{m_1+1}}(\emptyset)),\mathrm{NS}) \equiv_{E\cup E0,d} ff.$$

2.iii.b.2.1) $m_2>0$ und $j^1_{m_1+1}=j^2_{m_2}$.

Es ist st2 ~~is~~ NS. Nach Lemma K.NAT.SP0 gilt

$\mathrm{EQN}(S^{j^1_{m_1+1}}(\emptyset),S^{j^2_{m_2}}(\emptyset)) \equiv_{E\cup E0,d} tt$, d.h. wir können Gleichung E0.4 anwenden und erhalten

$$\begin{aligned}&\mathrm{EQS}(\mathrm{PU}(st1,S^{j^1_{m_1+1}}(\emptyset)),st2) &&\equiv_{E\cup E0,d}\\ &\mathrm{EQS}(st1,\mathrm{PU}(..\mathrm{PU}(\mathrm{NS},S^{j^2_1}(\emptyset)),..,S^{j^2_{m_2-1}}(\emptyset))) &&\equiv_{E\cup E0,d} ff,\end{aligned}$$

wobei die letzte i-Kongruenz wegen $m_1\neq m_2-1$ nach Induktionsannahme 2.ii) gilt.

2.iii.b.2.2) $m_2>0$ und $j^1_{m_1+1}\neq j^2_{m_2}$.

Es ist st2 ~~is~~ NS. Nach Lemma K.NAT.SP0 gilt

$$\mathrm{EQN}(S^{j^1_{m_1+1}}(\emptyset),S^{j^2_{m_2}}(\emptyset)) \equiv_{E\cup E0,d} ff,$$

d.h. wir können Gleichung E0.5 anwenden und erhalten

$$\mathrm{EQN}(\mathrm{PU}(st1,S^{j^1_{m_1+1}}(\emptyset)),st2) \equiv_{E\cup E0,d} ff.$$

Damit ist die Induktion unter ad 2) geschlossen. □

Lemma K.S.I2

Beh.: $(\forall m_1,m_2,j_1^1,\ldots,j_{m_1}^1,j_1^2,\ldots,j_{m_2}^2\in\omega_0)$

$$(\;1)\quad ((m_1=m_2 \wedge (\forall i\in\{1,\ldots,m_1\})(j_i^1=j_i^2)) \Rightarrow$$
$$EQS(PU(\ldots PU(NS,S^{j_1^1}(\emptyset)),\ldots,S^{j_{m_1}^1}(\emptyset)),$$
$$PU(\ldots PU(NS,S^{j_1^2}(\emptyset)),\ldots,S^{j_{m_2}^2}(\emptyset))) \equiv_{EI2,d} tt) \quad \wedge$$
$$2)\quad (((m_1=m_2 \wedge (\exists i\in\{1,\ldots,m_1\})(j_i^1\neq j_i^2)) \oplus m_1\neq m_2) \Rightarrow$$
$$EQS(PU(\ldots PU(NS,S^{j_1^1}(\emptyset)),\ldots,S^{j_{m_1}^1}(\emptyset)),$$
$$PU(\ldots PU(NS,S^{j_1^2}(\emptyset)),\ldots,S^{j_{m_2}^2}(\emptyset))) \equiv_{EI2,d} ff) \quad)$$

Bew.:

ad 1)

Seien $m_1,m_2,j_1^1,\ldots,j_{m_1}^1,j_1^2,\ldots,j_{m_2}^2\in\omega_0$. Wir kürzen ab:

$sti:=PU(\ldots PU(NS,S^{j_1^i}(\emptyset)),\ldots,S^{j_{mi}^i}(\emptyset))$ und

$ai:=AS(\ldots AS(NA,S(\emptyset),S^{j_1^i}(\emptyset)),\ldots,S^{m_i}(\emptyset),S^{j_{mi}^i}(\emptyset))$ für $i=1,2$.

Es gelte $m_1=m_2$ und $(\forall i\in\{1,\ldots,m_1\})(j_i^1=j_i^2)$.

Also gilt st1 is st2, a1 is a2 und $S^{m_1}(\emptyset)$ is $S^{m_2}(\emptyset)$.

Nach Lemma NF.I2 und SE von $\equiv_{EI2}$ gilt

$$EQS(st1,st2) \equiv_{EI2,d} EQS(ST(a1,S^{m_1}(\emptyset)),ST(a2,S^{m_2}(\emptyset)))$$
$$\text{is}\quad EQS(ST(a1,S^{m_1}(\emptyset)),ST(a1,S^{m_1}(\emptyset))).$$

Der letztgenannte Term ist bereits aus $T_{\Sigma I1,d}$ und nach Punkt 1) von Lemma EQ.STACK.I1 und nach Lemma 7 gilt

$$EQS(ST(a1,S^{m_1}(\emptyset)),ST(a1,S^{m_1}(\emptyset))) \equiv_{EI2,d} tt.$$

Aus der Transitivität von $\equiv_{EI2,d}$ folgt dann

$$EQS(st1,st2) \equiv_{EI2,d} tt,\quad \text{q.e.d.}.$$

Für den zweiten Teil des Beweises verwenden wir außer den obigen Abkürzungen auch noch

$a1':=AS(a1,S^{m_1+1}(\emptyset),S^{j_{m_1+1}^1}(\emptyset))$ mit $j_{m_1+1}^1\in\omega_0$ und

$a2':=AS(\ldots AS(NA,S(\emptyset),S^{j_1^2}(\emptyset)),\ldots,S^{m_2-1}(\emptyset),S^{j_{m_2-1}^2}(\emptyset))$.

ad 2) Durch Induktion über $m_1\in\omega_0$.

2.i) Induktionsanfang.

Sei $m_1=0$ und seien $m_2,j_1^2,\ldots,j_{m_2}^2\in\omega_0$.

Für $m_1=m_2=0$ ist die Prämisse von 2) falsch, denn es gilt $(\nexists i\in\{1,\dots,m_1\})(j_i^1\neq j_i^2)$; damit ist die ganze Implikation richtig.

Sei $m_2>0$, d.h. es ist $S^{m_1}(\emptyset)$ $\underline{is}$ $\emptyset$ und $S^{m_2}(\emptyset)$ $\cancel{\underline{is}}$ $\emptyset$.
Mit den Abkürzungen von ad 1) ist

$$st1\ \underline{is}\ NS \quad \text{und} \quad st2\ \cancel{\underline{is}}\ NS.$$

Nach Lemma NF.I2 gilt

$$NS \equiv_{EI2,s} ST(NA,\emptyset) \quad \text{und} \quad st2 \equiv_{EI2,s} ST(a2,S^{m_2}(\emptyset)),$$

woraus nach SE von $\equiv_{EI2}$ folgt

$$EQS(NS,st2) \equiv_{EI2,d} EQS(ST(NA,\emptyset),ST(a2,S^{m_2}(\emptyset))) \equiv_{EI2,d} ff;$$

dabei gilt die letzte i-Kongruenz nach Gleichung EOD.2.

2.ii) Induktionsannahme.
Die Behauptung gelte für ein beliebiges, aber festes $m_1\in\omega_0$ und für alle $m_2\in\omega_0$.

2.iii) Induktionsschluß.
Seien $m_2,j_1^1,\dots,j_{m_1}^1,j_{m_1+1}^1,j_1^2,\dots,j_{m_2}^2\in\omega_0$. Es gilt

$$PU(st1,S^{j_{m_1+1}^1}(\emptyset))\ \cancel{\underline{is}}\ NS.$$

Nach Lemma NF.I2 gilt

$$PU(st1,S^{j_{m_1+1}^1}(\emptyset)) \equiv_{EI2,s} ST(a1',S^{m_1+1}(\emptyset)) \quad \text{und}$$

$$st2 \equiv_{EI2,s} ST(a2,S^{m_2}(\emptyset)),$$

woraus nach SE von $\equiv_{EI2}$ folgt

*204 $EQS(PU(st1,S^{j_{m_1+1}^1}(\emptyset)),st2) \equiv_{EI2,d}$
$EQS(ST(a1',S^{m_1+1}(\emptyset)),ST(a2,S^{m_2}(\emptyset)))$.

Wir unterscheiden zwei Fälle.

2.iii.a) $m_1+1=m_2 \wedge (\exists i\in\{1,\dots,m_1+1\})(j_i^1\neq j_i^2)$.
Dann gilt $S^{m_1+1}(\emptyset)$ $\cancel{\underline{is}}$ $\emptyset$ und $S^{m_2}(\emptyset)$ $\cancel{\underline{is}}$ $\emptyset$.
Sei $V:=\{i\in\omega_0 \mid i\in\{1,\dots,m_1+1\} \wedge j_i^1\neq j_i^2\}$. V ist nicht leer.
Sei $k\in V$ mit $(\forall l\in V)(l\leq k)$, d.h. k ist größtes Element in V.
Dann gilt $(\forall l\in\{k+1,\dots,m_1+1\})(j_l^1=j_l^2)$. Man kann leicht durch

Induktion zeigen, daß gilt

$(\forall l \in \{k+1,\dots,m_1+1\})$

$(IN(a1',S^l(\emptyset))\ EQV\ IN(a2,S^l(\emptyset)) \equiv_{EI2,d} tt \quad \wedge$

$EQN(AC(a1',S^l(\emptyset)),AC(a2,S^l(\emptyset))) \equiv_{EI2,d} tt)$.

Der Übersichtlichkeit halber unterlassen wir den Beweis; es ist jedoch klar, daß die arrays a1' und a2 auf den letzten m_1-k+1 Stellen dieselben Einträge haben.

Durch (m_1-k+1)-malige Anwendung von Gleichung EOD.4 erhalten wir

*205 $EQS(ST(a1',S^{m_1+1}(\emptyset)),ST(a2,S^{m_2}(\emptyset))) \equiv_{EI2,d}$
$EQS(ST(a1',S^k(\emptyset)),ST(a2,S^k(\emptyset)))$.

Ebenfalls leicht einzusehen ist, daß

$IN(a1',S^k(\emptyset)) \equiv_{EI2,d} tt \equiv_{EI2,d} IN(a2,S^k(\emptyset))$,

$AC(a1',S^k(\emptyset)) \equiv_{EI2,n} S^{j_k^1}(\emptyset)$ und

$AC(a2,S^k(\emptyset)) \equiv_{EI2,n} S^{j_k^2}(\emptyset)$ gilt.

Nach Lemma EQV.I1 und Lemma 7 gilt

$IN(a1',S^k(\emptyset))\ EQV\ IN(a2,S^k(\emptyset)) \equiv_{EI2,d} tt$ und

nach SE von $\equiv_{EI2}$ und Lemma K.NAT.I2 gilt

$EQN(AC(a1',S^k(\emptyset)),AC(a2,S^k(\emptyset))) \equiv_{EI2,d} ff$.

Damit sind die Bedingungen von Gleichung EOD.5 erfüllt und es gilt

*206 $EQS(ST(a1',S^k(\emptyset)),ST(a2,S^k(\emptyset))) \equiv_{EI2,d} ff$.

Der Rest ist Transitivität von $\equiv_{EI2,d}$ mit *204 bis *206.

2.iii.b) $m_1+1 \neq m_2$.

2.iii.b.1) $m_2=0$, d.h. $ST(a2,S^{m_2}(\emptyset))$ is $ST(NA,\emptyset)$.

Dann gilt nach Gleichung EOD.3

*207 $EQS(ST(a1',S^{m_1+1}(\emptyset)),ST(NA,\emptyset)) \equiv_{EI2,d} ff$.

2.iii.b.2.1) $m_2>0$ und $j^1_{m_1+1}=j^2_{m_2}$.

Es ist $ST(a2,S^{m_2}(\emptyset))$ ~~is~~ $ST(NA,\emptyset)$. Es gilt

$IN(a1',S^{m_1+1}(\emptyset)) \equiv_{EI2,d} tt$ und $IN(a2,S^{m_2}(\emptyset)) \equiv_{EI2,d} tt$

und damit auch nach SE von $\equiv_{EI2}$ und Gleichung E.5

$$IN(a1',S^{m_1+1}(\emptyset))\ EQV\ IN(a2,S^{m_2}(\emptyset)) \equiv_{EI2,d} tt.$$

Ebenso gilt

$$AC(a1',S^{m_1+1}(\emptyset)) \equiv_{EI2,n} S^{j^1_{m_1+1}}(\emptyset),\ AC(a2,S^{m_2}(\emptyset)) \equiv_{EI2,n} S^{j^2_{m_2}}(\emptyset)$$

und nach SE von $\equiv_{EI2}$

$$\begin{aligned} &EQN(AC(a1',S^{m_1+1}(\emptyset)),AC(a2,S^{m_2}(\emptyset))) &&\equiv_{EI2,d}\\ &EQN(S^{j^1_{m_1+1}}(\emptyset),S^{j^2_{m_2}}(\emptyset)) &&\equiv_{EI2,d} tt, \end{aligned}$$

wobei die letzte i-Kongruenz nach Lemma K.NAT.I2 gilt.

Die Bedingungen von Gleichung EOD.4 sind erfüllt. Es gilt

$$\begin{aligned} *208\quad &EQS(ST(a1',S^{m_1+1}(\emptyset)),ST(a2,S^{m_2}(\emptyset))) &&\overset{1}{\equiv}_{EI2,d}\\ &EQS(ST(a1',S^{m_1}(\emptyset)),ST(a2,S^{m_2-1}(\emptyset))) &&\overset{2}{\equiv}_{EI2,d}\\ &EQS(ST(a1,S^{m_1}(\emptyset)),ST(a2',S^{m_2-1}(\emptyset))) &&\overset{3}{\equiv}_{EI2,d}\\ &EQS(st1,PU(\ldots PU(NS,S^{j^2_1}(\emptyset)),\ldots,S^{j^2_{m_2-1}}(\emptyset)))) &&\overset{4}{\equiv}_{EI2,d} ff, \end{aligned}$$

wobei benutzt wurde

1 Gleichung EOD.4,

2 Lemma ST.I2 liefert $ST(a1',S^{m_1}(\emptyset)) \sim_{EI2,s} ST(a1,S^{m_1}(\emptyset))$ und $ST(a2,S^{m_2-1}(\emptyset)) \sim_{EI2,s} ST(a2',S^{m_2-1}(\emptyset))$ und dann kann man die SE von $\sim_{EI2}$ und die Tatsache, daß $\equiv_{EI2,d}=\sim_{EI2,d}$ ist, verwenden.

3 Lemma NF.I2 und SE von $\equiv_{EI2}$

4 Induktionsannahme 2.ii).

2.iii.b.2.2) $m_2>0$ und $j^1_{m_1+1} \neq j^2_{m_2}$.

Es ist $ST(a2,S^{m_2}(\emptyset))$ ~~is~~ $ST(NA,\emptyset)$. wie in 2.iii.b.2.1) gilt

$$IN(a1',S^{m_1+1}(\emptyset))\ EQV\ IN(a2,S^{m_2}(\emptyset)) \equiv_{EI2,d} tt \quad \text{und}$$

$$AC(a1',S^{m_1+1}(\emptyset)) \equiv_{EI2,d} S^{j^1_{m_1+1}}(\emptyset),\ AC(a2,S^{m_2}(\emptyset)) \equiv_{EI2,d} S^{j^2_{m_2}}(\emptyset).$$

Nach SE von $\equiv_{EI2}$ und Lemma K.NAT.I2 gilt

$$\begin{aligned} &EQN(AC(a1',S^{m_1+1}(\emptyset)),AC(a2,S^{m_2}(\emptyset))) &&\equiv_{EI2,d}\\ &EQN(S^{j^1_{m_1+1}}(\emptyset),S^{j^2_{m_2}}(\emptyset)) &&\equiv_{EI2,d} ff. \end{aligned}$$

Die Bedingungen von Gleichung EOD.5 sind erfüllt. Es gilt

$$*209\quad EQS(ST(a1',S^{m_1+1}(\emptyset)),ST(a2,S^{m_2}(\emptyset))) \equiv_{EI2,d} ff.$$

Die Gültigkeit der Behauptung folgt für 2.iii.b.1), 2.iii.b.2.1) und 2.iii.b.2.2) aus der Transitivität von $\equiv_{EI2,d}$ zusammen mit *204 und jeweils *207, *208 bzw. *209.

Damit ist die Induktion unter ad 2) geschlossen. $\square$

Jetzt haben wir alle Hilfsmittel zusammen, um den Nachweis der R-Korrektheit von IMPL zu führen. Wir zeigen die zur R-Korrektheit äquivalente Aussage (iv) des Theorems 1. Vorbereitend dazu wird in den nächsten drei Lemmata für die Sorten nat, stack und dis gezeigt: wenn ein Term t in SP0 zu einem kanonischen Term i- bzw. t-kongruent ist, dann ist t in I2 zu demselben kanonischen Term i- bzw. t-kongruent.

Lemma RK.NAT

Beh.: $(\forall t \in T_{\Sigma\cup\Sigma 0,n})(\forall j \in \omega_0)(t \equiv_{E\cup E0,n} S^j(\emptyset) \Rightarrow t \equiv_{EI2,n} S^j(\emptyset))$

Bew.:

Durch Induktion über das Gewicht w(t) der Terme $t \in T_{\Sigma\cup\Sigma 0,n}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma\cup\Sigma 0,n}$ mit $w(t)=1$, d.h. t is $\emptyset$.

Nach Lemma K.NAT.SP0, Korollar 10 und der Konsistenz von SP0 wissen wir, daß es kein $j \in \omega_0$, $j>0$, gibt, sodaß

$\emptyset \equiv_{E\cup E0,n} S^j(\emptyset)$ gilt. Die Gültigkeit der Behauptung folgt für j=0 aus der Reflexivität von $\equiv_{E\cup E0,n}$ und $\equiv_{EI2,n}$.

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.

Die Behauptung gilt für alle $t \in T_{\Sigma\cup\Sigma 0,n}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma\cup\Sigma 0,n}$ mit $w(t)=n+1$, d.h. t ~~is~~ $\emptyset$.

Wir unterscheiden nach dem Aufbau von t zwei Fälle.

iii.a) t is S(t') mit $t' \in T_{\Sigma\cup\Sigma 0,n}$.

Es ist $w(t') \leq n$ und die Gültigkeit der Behauptung folgt für diesen Unterfall aus der Induktionsannahme ii) und der SE von $\sim_{E \cup E0}$ und $\sim_{EI2}$.

iii.b) t <u>is</u> TOP(p) mit $p \in T_{\Sigma \cup \Sigma 0, s}$.

Wir zeigen durch Induktion über das Gewicht w(p) der Terme $p \in T_{\Sigma \cup \Sigma 0, s}$ mit $w(p) \leq n$ zunächst die Hilfsbehauptung

*210 $(\forall m, j_1, \ldots, j_m \in \omega_0)$

$$(p \sim_{E \cup E0, s} PU(\ldots PU(NS, S^{j_1}(\emptyset)), \ldots, S^{j_m}(\emptyset)) \Rightarrow$$
$$p \sim_{EI2, s} PU(\ldots PU(NS, S^{j_1}(\emptyset)), \ldots, S^{j_m}(\emptyset)))$$

iii.b.i) Induktionsanfang.

Sei $p \in T_{\Sigma \cup \Sigma 0, s}$ mit $w(p)=1$, d.h. p <u>is</u> NS.

Aus dem Punkt c) des Beweises von Lemma KAT.SP0 wissen wir, daß es kein $m \in \omega_0$, $m>0$, und keine $j_1, \ldots, j_m \in \omega_0$ gibt, sodaß

$$NS \sim_{E \cup E0, s} PU(\ldots PU(NS, S^{j_1}(\emptyset)), \ldots, S^{j_m}(\emptyset)).$$

Die Gültigkeit der Behauptung folgt dann mit m=0 aus der Reflexivität von $\sim_{E \cup E0, s}$ und $\sim_{EI2, s}$.

iii.b.ii) Induktionsannahme.

Sei $k \in \omega$ fest mit $k<n$.

Die Behauptung gelte für alle $p \in T_{\Sigma \cup \Sigma 0, s}$ mit $w(p) \leq k$.

iii.b.iii) Induktionsschluß.

Sei $p \in T_{\Sigma \cup \Sigma 0, s}$ mit $w(p)=k+1$, d.h. p <u>~~is~~</u> NS.

Wir unterscheiden nach dem Aufbau von p zwei Fälle.

iii.b.iii.a) p <u>is</u> PU(p',r') mit $p' \in T_{\Sigma \cup \Sigma 0, s}$ und $r' \in T_{\Sigma \cup \Sigma 0, n}$.

Nach Lemma S.SP0, N.SP0 und Lemma 9 gibt es $m, j_1, \ldots, j_m \in \omega_0$, sodaß mit der Abkürzung

$p_m := PU(\ldots PU(NS, S^{j_1}(\emptyset)), \ldots, S^{j_m}(\emptyset))$ gilt

$p' \sim_{E \cup E0, s} p_m$ und $w(p_m) \leq w(p')$.

Es ist $w(p') \leq k$ und nach Induktionsannahme iii.b.ii) gilt

$$p' \sim_{E\cup E0,s} p_m \quad => \quad p' \sim_{EI2,s} p_m,$$

woraus nach SE von $\sim_{E\cup E0}$ und $\sim_{EI2}$ folgt

*211 $PU(p',r') \sim_{E\cup E0,s} PU(p_m,r') \;=>\; PU(p',r') \sim_{EI2,s} PU(p_m,r')$.

Nach Lemma N.SP0 gibt es ein $j_{m+1}\epsilon\omega_0$, sodaß

$$r' \equiv_{E\cup E0,n} S^{j_{m+1}}(\emptyset) \quad \text{bzw.} \quad r' \sim_{E\cup E0,n} S^{j_{m+1}}(\emptyset).$$

Es ist $w(r')\leq n$ und nach Induktionsannahme ii) gilt auch

*212 $r' \equiv_{EI2,n} S^{j_{m+1}}(\emptyset)$ bzw. $r' \sim_{EI2,n} S^{j_{m+1}}(\emptyset)$.

Mit *212 und SE von $\sim_{E\cup E0}$ und $\sim_{EI2}$ folgt aus *211

$$PU(p',r') \sim_{E\cup E0,s} PU(p_m,S^{j_{m+1}}(\emptyset)) \quad =>$$
$$PU(p',r') \sim_{EI2,s} PU(p_m,S^{j_{m+1}}(\emptyset)).$$

iii.b.iii.b) p <u>is</u> POP(p') mit $p'\epsilon T_{\Sigma\cup\Sigma0,s}$.

Nach Lemma S.SP0, N.SP0 und Lemma 9 gibt es $m,j_1,\ldots,j_m\epsilon\omega_0$, sodaß mit der Abkürzung p_m wie in iii.b.iii.a) gilt

$$p' \sim_{E\cup E0,s} p_m \quad \text{und} \quad w(p_m)\leq w(p').$$

Es ist $w(p')\leq k$ und nach Induktionsannahme iii.b.ii) gilt

$$p' \sim_{E\cup E0,s} p_m \quad => \quad p' \sim_{EI2,s} p_m,$$

woraus nach SE von $\sim_{E\cup E0}$ und $\sim_{EI2}$ folgt

*213 $POP(p') \sim_{E\cup E0,s} POP(p_m) \;=>\; POP(p') \sim_{EI2,s} POP(p_m)$.

Wir unterscheiden nach $m\epsilon\omega_0$.

iii.b.iii.b.1) m=0, d.h. p_m <u>is</u> NS.

Jetzt gilt nach Gleichung E0.8 und Lemma 9

*214 $POP(NS) \sim_{E\cup E0,s} NS$

und nach Gleichung EΣ0.7, EΣ0.11 und nach Lemma 9

*215 $POP(NS) \sim_{EI2,s} POP(ST(NA,\emptyset))$

$$\sim_{EI2,s} ST(NA,\emptyset) \quad \sim_{EI2,s} NS.$$

iii.b.iii.b.2) m>0, d.h. p_m ~~is~~ NS.

Nach Gleichung E0.9 und Lemma 9 gilt

*216 $POP(p_m) \sim_{E\cup E0,s} PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_{m-1}}(\emptyset))$.

Es gilt auch

$$*217\quad POP(p_m) \overset{1}{\sim}_{EI2,s} POP(ST(AS(..AS(NA,S(\emptyset),S^{j_1}(\emptyset)),..,S^m(\emptyset),S^{j_m}(\emptyset)),S^m(\emptyset))$$
$$\overset{2}{\sim}_{EI2,s} ST(AS(..AS(NA,S(\emptyset),S^{j_1}(\emptyset)),..,S^m(\emptyset),S^{j_m}(\emptyset)),S^{m-1}(\emptyset))$$
$$\overset{3}{\sim}_{EI2,s} ST(AS(..AS(NA,S(\emptyset),S^{j_1}(\emptyset)),..,S^{m-1}(\emptyset),S^{j_{m-1}}(\emptyset)),S^{m-1}(\emptyset))$$
$$\overset{4}{\sim}_{EI2,s} PU(...PU(NS,S^{j_1}(\emptyset)),...,S^{j_{m-1}}(\emptyset)),$$

wobei benutzt wurde

1 Lemma NF.I2, Lemma 9 und SE von $\sim_{EI2}$,

2 Gleichung EΣ0.10 und Lemma 9,

3 Lemma ST.I2 und

4 Lemma NF.I2 und Lemma 9.

Für iii.b.iii.b.2.1) und iii.b.iii.b.2.2) folgt die Gültigkeit der Behauptung aus der Transitivität von $\sim_{E\cup E0,s}$ und $\sim_{EI2,s}$ zusammen mit *213 und *214 und *215 bzw. *216 und *217. Damit ist die Induktion unter iii.b) geschlossen.

Nach Lemma S.SP0, N.SP0 und Lemma 9 gibt es $m,j_1,...,j_m \in \omega_0$, sodaß mit der Abkürzung p_m wie in iii.b.iii.a) gilt

$$p \sim_{E\cup E0,s} p_m \quad \text{und} \quad w(p_m) \leq w(p).$$

Nach *210 gilt dann auch

$$p \sim_{EI2,s} p_m$$

und die SE von $\sim_{E\cup E0}$ und $\sim_{EI2}$ liefert

$$TOP(p) \sim_{E\cup E0,n} TOP(p_m) \quad => \quad TOP(p) \sim_{EI2,n} TOP(p_m).$$

Da SP0 und I2 kategorisch bzgl. <u>nat</u> sind, gilt auch

$$*218\quad TOP(p) \equiv_{E\cup E0,n} TOP(p_m) \quad => \quad TOP(p) \equiv_{EI2,n} TOP(p_m).$$

Wir unterscheiden nach $m \in \omega_0$.

iii.b.1) m=0, d.h. p_m <u>is</u> NS.

Dann gilt nach Gleichung E0.6

$$*219\quad TOP(NS) \equiv_{E\cup E0,n} \emptyset$$

und nach Gleichung EΣ0.7 und EΣ0.11

$$*220\quad TOP(NS) \equiv_{EI2,n} TOP(ST(NA,\emptyset)) \equiv_{EI2,n} \emptyset.$$

iii.b.2) m>0, d.h. p_m ~~is~~ NS.

Dann gilt nach Gleichung E0.7

$*221 \quad TOP(p_m) \equiv_{E\cup E0,n} S^{j_m}(\emptyset)$ und es gilt

$$*222 \quad TOP(p_m) \overset{1}{\equiv}_{EI2,n} TOP(ST(AS(..AS(NA,S(\emptyset),S^{j_1}(\emptyset)),..,S^m(\emptyset),S^{j_m}(\emptyset)),S^m(\emptyset))$$
$$\overset{2}{\equiv}_{EI2,n} AC(AS(..AS(NA,S(\emptyset),S^{j_1}(\emptyset)),..,S^m(\emptyset),S^{j_m}(\emptyset)),S^m(\emptyset))$$
$$\overset{3}{\equiv}_{EI2,n} S^{j_m}(\emptyset),$$

wobei benutzt wurde

1 Lemma NF.I2 und SE von $\equiv_{EI2}$,

2 Gleichung $E\Sigma0.12$ und

3 Gleichung E1.11, die anwendbar war, da nach Punkt 1) von Korollar EQ.NAT.I2 $EQN(S^m(\emptyset),S^m(\emptyset)) \equiv_{EI2,n} tt$ galt.

Für iii.b.1) und iii.b.2) folgt die Gültigkeit der Behauptung aus der Transitivität von $\equiv_{E\cup E0,n}$ und $\equiv_{EI2,n}$ zusammen mit *218 und *219 und *220 bzw. *221 und *222.

Damit ist die Induktion geschlossen. □

Lemma RK.STACK

Beh.: $(\forall t \in T_{\Sigma\cup\Sigma0,s})(\forall m,j_1,\dots,j_m \in \omega_0)$
$(t \sim_{E\cup E0,s} PU(\dots PU(NS,S^{j_1}(\emptyset)),\dots,S^{j_m}(\emptyset)) \Rightarrow$
$t \sim_{EI2,s} PU(\dots PU(NS,S^{j_1}(\emptyset)),\dots,S^{j_m}(\emptyset)))$

Bew.:

Der Beweis verläuft analog zum Beweis von *210, wobei jedoch die Beschränkung des Gewichts der zu betrachtenden Terme entfällt. Daher wird auch, um die *212 entsprechende Aussage zu zeigen, nicht wie im Beweis von *210 die Induktionsannahme ii) verwendet, sondern Lemma RK.NAT. □

Lemma RK.DIS

Beh.: $(\forall t \in T_{\Sigma\cup\Sigma0,d})(\forall tv \in \{tt,ff\})(t \equiv_{E\cup E0,d} tv \Rightarrow t \equiv_{EI2,d} tv)$

Bew.:

Durch Induktion über das Gewicht w(t) der Terme $t \in T_{\Sigma\cup\Sigma0,d}$.

i) Induktionsanfang.

Sei $t \in T_{\Sigma \cup \Sigma 0,d}$ mit $w(t)=1$, d.h. t is tt oder t is ff.
In beiden Fällen folgt die Gültigkeit der Behauptung aus der Reflexivität von $\equiv_{E \cup E0,d}$ und $\equiv_{EI2,d}$ und der Konsistenz von SP0.

ii) Induktionsannahme.

Sei $n \in \omega$ beliebig, aber fest.
Die Behauptung gelte für alle $t \in T_{\Sigma \cup \Sigma 0,d}$ mit $w(t) \leq n$.

iii) Induktionsschluß.

Sei $t \in T_{\Sigma \cup \Sigma 0,d}$ mit $w(t)=n+1$, d.h. t ~~is~~ tt und t ~~is~~ ff.
Wir unterscheiden nach dem Aufbau von t. Die Fälle

iii.a) t is $\neg t'$ mit $t' \in T_{\Sigma \cup \Sigma 0,d}$,
iii.b) t is t_1 AND t_2 mit $t_1, t_2 \in T_{\Sigma \cup \Sigma 0,d}$ und
iii.c) t is t_1 EQV t_2 mit $t_1, t_2 \in T_{\Sigma \cup \Sigma 0,d}$

sind klar nach Vollständigkeit von SP0, Induktionsannahme ii) und den Gleichungen E.1 bis E.6. Es bleiben zwei Fälle.

iii.d) t is $EQN(t_1,t_2)$ mit $t_1, t_2 \in T_{\Sigma \cup \Sigma 0,n}$.

Nach Lemma N.SP0 gibt es $j_1, j_2 \in \omega_0$, sodaß für $i=1,2$

$$t_i \equiv_{E \cup E0,n} S^{j_i}(\emptyset)$$

gilt, woraus nach Lemma RK.NAT folgt

$$t_i \equiv_{EI2,n} S^{j_i}(\emptyset).$$

Wegen der SE von $\equiv_{E \cup E0}$ und $\equiv_{EI2}$ gilt damit auch

*223 $EQN(t_1,t_2) \equiv_{E \cup E0,d} EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset)) \Rightarrow$
$EQN(t_1,t_2) \equiv_{EI2,d} EQN(S^{j_1}(\emptyset),S^{j_2}(\emptyset))$.

Die Gültigkeit der Behauptung für diesen Unterfall folgt aus der Transitivität von $\equiv_{E \cup E0,d}$ und $\equiv_{EI2,d}$ zusammen mit *223 und Punkt 1) von Lemma K.NAT.SP0 und K.NAT.I2 für $j_1=j_2$ bzw. Punkt 2) derselben Lemmata für $j_1 \neq j_2$.

iii.e) t is $EQS(t_1,t_2)$ mit $t_1, t_2 \in T_{\Sigma \cup \Sigma 0,s}$.

Nach Lemma S.SP0, N.SP0 und Lemma 9 gibt es

$m_1, m_2, j_1^1, \ldots, j_{m_1}^1, j_1^2, \ldots, j_{m_2}^2 \in \omega_0$, sodaß mit der Abkürzung

$sti := PU(\ldots PU(NS, S^{j_1^i}(\emptyset)), \ldots, S^{j_{m_i}^i}(\emptyset))$ für $i=1,2$ gilt

$$t_i \sim_{E \cup E0,s} sti.$$

Nach Lemma RK.STACK gilt daher auch für $i=1,2$

$$t_i \sim_{EI2,s} sti.$$

Mit der SE von $\sim_{E \cup E0}$ und $\sim_{EI2}$ folgt daraus

$$EQS(t_1,t_2) \sim_{E \cup E0,d} EQS(st1,st2) \quad \Rightarrow$$

$$EQS(t_1,t_2) \sim_{EI2,d} EQS(st1,st2)$$

bzw. wegen $\equiv_{E \cup E0,d} = \sim_{E \cup E0,d}$ und $\equiv_{EI2,d} = \sim_{EI2,d}$

*224 $EQS(t_1,t_2) \equiv_{E \cup E0,d} EQS(st1,st2) \Rightarrow EQS(t_1,t_2) \equiv_{EI2,d} EQS(st1,st2)$.

Jetzt gibt es zwei Möglichkeiten.

iii.e.1) Es gilt $st1 \sim_{E \cup E0,s} st2$.

Dann gilt nach Lemma RK.STACK auch $st1 \sim_{EI2,s} st2$ und

*225 $EQS(st1,st2) \overset{1}{\equiv}_{E \cup E0,d} EQS(st1,st1) \overset{2}{\equiv}_{E \cup E0,d} tt$ und

*226 $EQS(st1,st2) \overset{3}{\equiv}_{EI2,d} EQS(st1,st1) \overset{4}{\equiv}_{EI2,d} tt$,

wobei benutzt wurde

1 SE von $\sim_{E \cup E0}$ und $\equiv_{E \cup E0,d} = \sim_{E \cup E0,d}$,

2 Punkt 1) von Lemma K.S.SP0,

3 SE von $\sim_{EI2}$ und $\equiv_{EI2,d} = \sim_{EI2,d}$ und

4 Punkt 1) von Lemma K.S.I2.

iii.e.2) Es gilt $st1 \not\sim_{E \cup E0,s} st2$.

Dann gilt nach Punkt c) des Beweises von Lemma KAT.SP0 und wegen der Vollständigkeit und Konsistenz von SP0 entweder $m_1 \neq m_2$ oder $(m_1 = m_2 \wedge (\exists i \in \{1, \ldots, m_1\})(j_i^1 \neq j_i^2))$.

Nach Punkt 2) von Lemma K.S.SP0 gilt

*227 $EQS(st1,st2) \equiv_{E \cup E0,d} ff$

und nach Punkt 2) von Lemma K.S.I2 gilt

*228 $EQS(st1,st2) \equiv_{EI2,d} ff$.

Die Gültigkeit der Behauptung folgt für iii.e.1) und iii.e.2) aus der Transitivität von $\equiv_{E\cup E0,d}$ und $\equiv_{EI2,d}$ zusammen mit *224 und *225 und *226 bzw. *227 und *228.

Damit ist die Induktion geschlossen. □

Lemma RK

Beh.: $(\forall s\in\{\underline{dis},\underline{nat},\underline{stack}\})(\forall t_1,t_2\in T_{\Sigma\cup\Sigma 0,s})$

$(t_1 \sim_{E\cup E0,s} t_2 \quad\Rightarrow\quad t_1 \sim_{EI2,s} t_2)$

Bew.:

Durch Fallunterscheidung über die Sorten $s\in S\cup S0$.

a) $s=\underline{dis}$.

Seien $t_1,t_2\in T_{\Sigma\cup\Sigma 0,d}$ und $t_1 \sim_{E\cup E0,d} t_2$ bzw. $t_1 \equiv_{E\cup E0,d} t_2$.

Da SP0 konsistent und vollständig ist, gilt

$(\overset{1}{\exists} tv\in\{tt,ff\})(\forall i\in\{1,2\})(t_i \equiv_{E\cup E0,d} tv)$.

Für dieses tv gilt nach Lemma RK.DIS und wegen der Vollständigkeit und Konsistenz von I2

$$(\forall i\in\{1,2\})(t_i \equiv_{EI2,d} tv),$$

und damit auch $t_1 \equiv_{EI2,d} t_2$ bzw. $t_1 \sim_{EI2,d} t_2$.

b) $s=\underline{nat}$.

Seien $t_1,t_2\in T_{\Sigma\cup\Sigma 0,n}$ mit $t_1 \sim_{E\cup E0,n} t_2$.

Da SP0 kategorisch ist, gilt auch

$$t_1 \equiv_{E\cup E0,n} t_2.$$

Nach Lemma N.SP0 gibt es ein $j\in\omega_0$, das nach Korollar 10 und Lemma K.NAT.SP0 eindeutig ist, sodaß

$$t_1 \equiv_{E\cup E0,n} s^j(\emptyset).$$

Dann gilt auch

$$t_2 \equiv_{E\cup E0,n} s^j(\emptyset)$$

und nach Lemma RK.NAT folgt daraus, daß für i=1,2

$$t_i \equiv_{EI2,n} s^j(\emptyset).$$

Wegen der Transitivität von $\equiv_{EI2,n}$ gilt $t_1 \equiv_{EI2,n} t_2$,

woraus nach Lemma 9 folgt $t_1 \sim_{EI2,n} t_2$.

c) s=stack.

Seien $t_1, t_2 \in T_{\Sigma\cup\Sigma 0,s}$ und es gelte $t_1 \sim_{E\cup E0,s} t_2$.
Nach Lemma S.SP0, N.SP0 und Lemma 9 gibt es $m, j_1, \ldots, j_m \in \omega_0$, die nach Punkt c) aus dem Beweis von Lemma KAT.SP0 eindeutig sind, sodaß für i=1,2 gilt

$$t_i \sim_{E\cup E0,s} PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_m}(\emptyset)).$$

Nach Lemma RK.STACK gilt damit für i=1,2

$$t_i \sim_{EI2,s} PU(\ldots PU(NS,S^{j_1}(\emptyset)),\ldots,S^{j_m}(\emptyset))$$

bzw. nach den Äquivalenzeigenschaften von $\sim_{EI2,s}$

$$t_1 \sim_{EI2,s} t_2. \square$$

Korollar RK.IMPL

Beh.: IMPL ist R-korrekt.

Bew.:

Die Gültigkeit der Behauptung folgt aus Lemma RK und aus Theorem 1. $\square$

Korollar T.IMPL

Beh.: IMPL ist eine T-Implementierung von SP0 mit Hilfe von SP1.

Bew.:

Die Gültigkeit der Behauptung folgt aus den Korollaren T.ERW.I1, T.ANR.I2 und RK.IMPL. $\square$

4. Analyse

4.1. Vergleich der Implementierungskonzepte

Die Vergleichsmöglichkeiten des Konzeptes der T-Implementierungen mit dem EKP-Konzept der I-Implementierungen sind begrenzt, da EKP nur Spezifikationen ohne bedingte Axiome betrachten. Ein Vorteil des EKP-Konzeptes ist die klare Trennung von Syntax und Semantik. Demgegenüber ist die Bestimmung der "zusätzlichen Operationen", also der Menge $\Sigma 0D$ und der dazugehörigen Axiomenmenge E0D bei den T-Implementierungen nicht mehr rein syntaktisch möglich. Jedoch deutete sich im Beispiel 3.2.4. bereits eine Lösungsmöglichkeit an:

Wenn wir annehmen, daß es für praktische Zwecke ausreicht, als "zusätzliche Operation" ein Gleichheitsprädikat EQ.s für jede Sorte s zu spezifizieren, dann können wir folgendes festlegen:

- Wir betrachten nur noch t-Spezifikationen, die für jede Sorte s ein Operationssymbol EQ.s:s s $\rightarrow$ <u>dis</u> mit den folgenden Eigenschaften besitzen:

  $(\forall t,t' \in T_{\Sigma,s})(t \equiv_{E,s} t' \iff EQ.s(t,t') \equiv_{E,dis} tt)$.

- Auf der syntaktischen Ebene fordern wir für die (schwache) T-Implementierung:

  $\Sigma 0D := \{EQ.s_0 \mid s_0 \in S0\}$ und

  $E0D := \{(L,R) \mid (L,R) \in E\Sigma 0 \wedge L/2 \text{ ist } EQ.s_0\text{-wurzelnd} \wedge EQ.s_0 \in \Sigma 0D\}$.

Damit sind $\Sigma 0D$ und E0D syntaktisch klar festgelegt.

Die Einschränkung, nur solche t-Spezifikationen zu betrachten, die für jede Sorte ein durch Axiome spezifiziertes Gleichheitsprädikat besitzen, scheint uns eine natürliche zu sein. Der Nachweis dieser Eigenschaften wurde für EQV im Lemma EQV.I1 und für EQN in den Korollaren 9 bis 13 geführt, war jedoch nicht für alle Sorten in allen betrachteten Spezifikationen notwendig. Der Beweisaufwand würde also durch eine solche Einschränkung vergrößert.

Durch die Hinzunahme der zusätzlichen dis-Operationen zur ersten Implementierungsebene können wir auch nicht mehr allein durch eine syntaktische Bedingung wie in [EKP 79a] (sh. das Lemma 5.1. unter 3.1.4.) garantieren, daß die erste Implementierungsebene eine (t-)Erweiterung der implementierenden Spezifikation ist. Das entsprechende Ergebnis, das in Lemma 16 formuliert ist, ist schwächer und betrifft nur die t-Vollständigkeit der ersten Implementierungsebene.

Es sollte an dieser Stelle nochmals darauf hingewiesen werden, daß diese Einschränkungen in der Verwendung der terminalen Algebrasemantik wurzeln, nicht aber in der Wahl des T-Implementierungskonzeptes.

Im Anhang von [EKP 79a] findet sich ebenfalls ein Implementierungsbeispiel, daß die Implementierung von stacks durch pointer/array-Paare behandelt, wobei jedoch ein error handling stattfindet, auf welches in 3.2.4. aus Gründen der Übersichtlichkeit der zu führenden Beweise verzichtet wurde. EKP gehen beim Korrektheitsnachweis ihres Beispiels, der nicht sehr detailliert geführt ist, einen anderen Weg; sie benutzen ihre Kenntnis mathematischer Modelle der beteiligten Datentypen und zeigen die RI-Korrektheit und die Erweiterungseigenschaft von IMPL2 auf IMPL1 durch Angabe von Σ-Homomorphismen zwischen den Modellen und den jeweiligen i-Semantiken und durch Angabe der "Abstraktionsfunktion" $\alpha: A' \to B$; dabei ist $A' = h(T_{\Sigma SPEC0})$ das Bild des $\Sigma \cup \Sigma 0$-Homomorphismus $h: T_{\Sigma SPEC0} \to A$ in dem mathematischen Modell A von IMPL2 und B ist das mathematische Modell von SPEC0. α entspricht dem $\Sigma \cup \Sigma 0$-Homomorphismus REP aus 3.1.4..

In [P 79] wurde ein anderer Weg des Korrektheitsnachweises für die I-Implementierungen von EKP gewählt, und zwar der Korrekt-

heitsnachweis durch ausschließlichen Gebrauch kanonischer Termalgebren (CTA's). Dabei geht Padawitz von der Überlegung aus, daß der Korrektheitsnachweis über mathematische Modelle davon abhängt, daß der Designer des Datentyps diese Modelle kennt und somit schwerlich automatisierbar ist. Andererseits ist die (i-)Semantik einer Spezifikation immer durch eine kanonische Termalgebra darstellbar. Padawitz weist darauf hin, daß [P 79] einen experimentellen Charakter hat und keine endgültigen Resultate beinhaltet. Er verifiziert seine Überlegungen exemplarisch am Beispiel der bereits aus [EKP 79a] bekannten Implementierung von stacks durch pointer/array-Paare.
Das EKP-Konzept kann mächtiger werden, wenn es auf initial zu interpretierende Spezifikationen im Sinne von Definition 21, die bedingte Axiome zuläßt, erweitert wird, was meines Erachtens ohne Schwierigkeiten möglich ist.
Die T-Implementierungen sind als Konzept einer Beschreibungssprache für abstrakte Datentypen ebenso geeignet wie die I-Implementierungen von EKP.

4.2. Die terminale Algebrasemantik in der Anwendung

Der Nachweis der T-Implementierungseigenschaften in 3.2.4.2. war zugleich ein erstes größeres Anwendungsbeispiel für den Ansatz der terminalen Algebrasemantik. Es hat sich dabei gezeigt, daß die Bestimmung der t-Kongruenzrelation meist schwieriger ist als die Bestimmung der i-Kongruenz. Man muß sich damit behelfen, daß man die t-Kongruenz mit Hilfe der i-Kongruenz charakterisiert, wie etwa in Lemma K.STACK.I1 geschehen. Diese Charakterisierung entspricht der "Interpretation von Gleichheitsoperatoren" in [GHM 76a], Abschnitt 4.4..

Die Haupthilfsmittel in den Beweisen unter 3.2.4.2. waren:

1) Verwendung von algebraisch spezifizierten Gleichheitsprädikaten,
2) Kenntnis kanonischer Terme und
3) Vollständigkeits- und Konsistenzeigenschaften der beteiligten t-Spezifikationen.

Das Vorhandensein der Sorte dis und die Notwendigkeit, Vollständigkeitsnachweise zu führen, brachte zwar zusätzliche Beweislast, hatte aber auch erhebliche Vorteile, denn die algebraisch spezifizierten Gleichheitsprädikate und die Vollständigkeits- und Konsistenzeigenschaften ermöglichten präzise Korrektheitsaussagen; so wurde zum Beispiel im Beweis von Lemma KAT.SP0 gezeigt, daß zwei stack-Terme t_1 und t_2 genau dann i- bzw. t-kongruent sind, wenn sie zu demselben kanonischen stack-Term i-kongruent sind. Verschiedene kanonische stack-Terme sind also Repräsentanten verschiedener $\equiv_{E \cup E0,\text{stack}}$-Kongruenzklassen, und damit ist der Bezug zu den kanonischen Termalgebren hergestellt.

4.3. Offene Probleme

Wie bereits in [EKP 79a] gesagt, ist die detaillierte Ausführung des Nachweises der Korrektheit der (I-)Implementierungen für Spezifikationen realistischer Größe kaum von Hand zu bewältigen. Dies gilt mit Sicherheit auch für die T-Implementierungen, denn selbst unser einfaches Beispiel erforderte in dieser Hinsicht bereits einen enormen Aufwand. Damit taucht schon die Frage auf, ob die zu führenden Beweise maschinell unterstützt werden können.

Die Fragen, ob eine t-Spezifikation konsistent, vollständig,

t-Erweiterung oder t-Anreicherung einer anderen t-Spezifikation ist, sind jedoch im allgemeinen nicht entscheidbar. Es müssen also Kriterien gefunden werden, die diese Fragen für viele praktische Fälle beantwortbar machen und die maschinell nachzuweisen sind. Ein Beispiel dafür ist der Knuth-Bendix-Algorithmus [KB 70], mit dem in vielen Fällen die Eigenschaft der eindeutigen Terminierung für z.B. die aus der Axiomenmenge einer Spezifikation gebildete Rewrite-Rule-Menge nachgewiesen werden kann.
Für den Korrektheitsnachweis im terminalen Fall haben sich grundsätzlich nur vier verschiedene Arten von Beweisen ergeben:
- Konsistenzbeweise,
- t-Konsistenzbeweise,
- Nachweis der R-Korrektheit und
- t-Vollständigkeitsbeweise

(wobei die ersten drei noch zu einer Gruppe von konsistenzähnlichen Beweisen zusammengefaßt werden können).
Erstere können durch eine Erweiterung des KBA unterstützt werden, während für die nächsten beiden noch jegliche Kriterien fehlen. Die t-Vollständigkeit läßt sich fast immer auf die i-Vollständigkeit zurückführen.
In [EKP 79b] werden syntaktische Kriterien angegeben, die für Spezifikationen SP und SP' ohne bedingte Axiome mit SSP=SSP', $\Sigma SP \subseteq \Sigma SP'$ und $ESP \subseteq ESP'$ in vielen Fällen entscheiden können, ob SP' i-vollständig auf SP ist. Solche Kriterien werden auch für Spezifikationen mit bedingten Axiomen gesucht, wobei $SSP \subseteq SSP'$ zugelassen werden sollte.
Von Interesse sind auch weitergehende Untersuchungen über Komposition von T-Implementierungen, analog zu den Untersuchungen in [EKMP 80], und die Erweiterung des T-Implementierungskonzeptes auf parametrisierte Spezifikationen.

5. Index der Definitionen und Sätze

Dieser Index soll dem Leser das Nachvollziehen der Beweise erleichtern. Definitionen (D) und Theoreme (T) sind nach Nummern und Korollare (K), Lemmata (L) und Sätze (S) nach Nummern bzw. lexikographisch nach ihren mnemonischen Namen geordnet.

6. Literaturverzeichnis

[ADJ 77] Goguen, J.A. – Thatcher, J.W. – Wagner, E.G.
"An Initial Algebra Approach to the Specification, Correctness and Implementation of Abstract Data Types"
IBM Research Report RC 6487, New York, May 1977

[ADJ 78] Goguen, J.A. – Thatcher, J.W. – Wagner, E.G. – Wright, J.B.
"Data Type Specifications: Parametrization and the Power of Specification Techniques"
Proc. SIGACT 10th Ann. Symp. Thy. of Comp., 1978

[Ehr 78] Ehrich, H.D.
"On the Theory of Specification, Implementation and Parametrization of Abstract Data Types"
Universität Dortmund, Research Report, 1978

[EKMP 80] Ehrig, H. – Kreowski, H.J. – Mahr, B. – Padawitz, P.
"Compound Algebraic Implementations: An Approach to Stepwise Refinement of Software Systems"
TU Berlin, FB Informatik, Jan 1980

[EKP 78] Ehrig, H. – Kreowski, H.J. – Padawitz, P.
"Stepwise Specification and Implementation of Abstract Data Types"
TU Berlin, March 78

[EKP 79a] ——— " ———
"Algebraische Implementierungen Abstrakter Datentypen"
TU Berlin, März 1979, Bericht Nr. 79-3

[EKP 79b] ——— " ———
"Completeness in Algebraic Specifications"

TU Berlin, December 1979

[EKW 79] Ehrig,H. – Kreowski,H.J. – Weber,H.
"New Aspects of Algebraic Specification Schemes for Data Base Systems"
Proc. Fachtagung "Formale Modelle für Informationssysteme", Tutzing am See, Mai 1979

[GHM 76a] Guttag,J.V. – Horowitz,E. – Musser,D.R.
"Abstract Data Types and Software Validation"
USC, Marina del Rey, 90291 California, August 1976

[GN 78] Goguen,J.A. – Nourani,F.
"Some Algebraic Techniques for Proving Correctness of Data Type Implementation"
Extended Abstract, Comp. Sci. Dept., UCLA, Los Angeles, 1978

[Gut 75] Guttag,J.V.
"Specification and Application to Programming of Abstract Data Types"
University of Toronto, Computer Systems Research Group, Technical Report CSRG-59, 1975

[H 79] Hornung,G.
"Einige Probleme der Algebrasemantik Abstrakter Datentypen"
Universität Bonn, MEMO-SEKI-BN-79-07, Oktober 1979

[HO 80] Huet,G. – Oppen,D.C.
"Equations and Rewrite Rules: a Survey"
Technical Report CSL-111, SRI International, Menlo Park, January 1980

[HR 79] Hornung,G. – Raulefs,P.
"Terminal Algebra Semantics and Retractions for Abstract Data Types"
Universität Bonn, MEMO-SEKI-BN-79-6, 1979

[KB 70] Knuth,D.E. – Bendix,P.B.
"Simple Word Problems in Universal Algebras"
in: Computational Problems in Abstract Algebra, J. Leech ed., p. 263-297, Oxford 1970

[Kre 78] Kreowski,H.J.
"Algebra für Informatiker"
Skriptum zur Vorlesung im WS 78/79 an der TU Berlin

[LS 77] Lehmann,D.J. – Smyth,M.B.
"Data Types"
Proc. 18th IEEE Symp. on Found. of Computing, Providence, R.I., November 1977, pp. 7-12

[LZ 74] Liskov,B. – Zilles,S.
"Programming with Abstract Data Types"
Proc. of ACM Symp. on Very High Level Languages, SIGPLAN Notices 9 (1974), pp. 50-59

[M 78] Musser,D.R.
"A Data Type Verification System Based on Rewrite Rules"
USC, June 1978

[R 79] Raulefs,P.
"Einführung in die Theorie der Datenstrukturen"
Vorlesungsnotizen zur Vorlesung, Uni Bonn, 1979

[Wan 78] Wand,M.
"Final Algebra Semantics and Data Type Extensions"
Tech. Rep. No. 65, Indiana University,
Comp. Sci. Dept., Bloomington 1978

[Zil 74] Zilles,S.
"Algebraic Specification of Data Types"
Proj. MAC Report 11, MIT, Cambridge, Mass., (1974)
pp. 25-28