Logical Semantics of Modularisation *

Gerard R. Renardel de Lavalette

University of Groningen, Department of Computing Science,
P.O.Box 800, 9700 AV Groningen, the Netherlands

Abstract. An algebra of theories, signatures, renamings and the operations import and export is investigated. A normal form theorem for terms of this algebra is proved. Another algebraic approach and the relation with a fragment of second order logic are also considered.

1 Introduction

Modularisation is (together with parametrisation) a key feature in order to describe and design complex objects in a manageable and comprehensible way. In this paper we study the logical aspects of modularisation in formal (programming or specification) languages. This is done by investigating some natural and useful operations on *theories*, the objects that express the logical semantics of such languages. The usual names for these operations in the jargon of computer science are import (in logical terms: combination of theories), export (restricting the signature of a theory) and renaming. The results are presented in an algebraic fashion.

1.1 Relation with Other Work

Operators on modules and their semantics have been studied in e.g. [1] (in the context of CLEAR), [5] (in the context of PLUSS), [3] and [4] (using category theory), [16] (using model class semantics). Our main source of inspiration has been [2], where the approach is similar to Wirsing's in [16], extended to theory semantics and countable model semantics. The role of the interpolation theorem for the theory semantics of import and export has been pointed out in [7]. Besides giving a survey of logical aspects of modularisation, this paper contains, to the best of our knowledge, the following new points:

- investigation of the behaviour of import and export in combination with non-bijective renamings on theories;
- a normal form theorem for theory terms constructed with these operators;
- a (trivial) counterexample for interpolation in conditional equational logic;
- definition of import and export on theories using two orthogonal closure properties;
- relation with the $[\&, \exists]$-fragment of second order logic.

* appeared in: Computer Science Logic (E. Börger, G. Jäger, H. Kleine Büning, M.M. Richter, eds.), Springer Lecture Notes in Computer Science 626 (1992) 306 – 315

1.2 Survey of the Rest of the Paper

In Sect. 2 we introduce signatures, theories, renamings and operations defined on them. Axioms for these operations are given in Sect. 3, where also the relation of one of these properties with the interpolation theorem is considered, as well as some results on interpolation in (conditional) equational logic. Section 4 is about normal forms of so-called theory terms. In the last two sections we sketch some related ideas: reducing the theory operations of Sect. 2 to two orthogonal closure operators, and a theory semantics for the $[\&, \exists]$-fragment of second order logic.

1.3 Acknowledgements

The research reported here has been performed while the author was at the Department of Philosophy of the University of Utrecht. Partial support has been received from the European Communities under ESPRIT projects METEOR and ATMOSPHERE through Philips Research Laboratories Eindhoven and SERC (Software Engineering Research Centre, Utrecht), respectively. Part of the work reported here is also in [11]. Discussion and collaboration with Jan Bergstra, Loe Feijs, Hans Jonkers, Karst Koymans and Piet Rodenburg were influential and are gratefully acknowledged. Finally I thank Maureen Reyers and Peter de Bruin for converting an earlier version of this paper to LaTeX.

2 Signatures, Renamings and Theories

We assume some logical language L with a derivability relation $\vdash$. L contains signature elements, e.g. sorts, functions, predicates. Signature elements can have a type: an arity (required number of arguments) or a sort type (a list of input and output sorts). We assume that, for every type, there are infinitely many signature elements having that type: this will allow us to apply the *fresh signature element principle* (see the end of this section). A signature is a finite set of signature elements; it is called *closed* if it contains all sorts occurring in the types of its elements (observe that closedness is preserved under union and intersection). The closure $c(\Sigma)$ of a signature is the least closed signature containing Σ. If X is a (collection of) expression(s) in the language of L then $S(X)$ is the closure of the collection of all signature elements occurring in (elements of) X.

From now on, we adopt the default convention that Σ and Π range over closed signatures.

Let Γ be a collection of sentences of L, Σ a signature, then the *closure of Γ in Σ* is defined by

$$Cl(\Sigma, \Gamma) =_{\text{def}} \{A \mid \Gamma \vdash A \text{ and } S(A) \subseteq \Sigma\} .$$

These closures are called *theories*. The *union* of two theories is the smallest theory containing them, defined by

$$T + U =_{\text{def}} Cl(S(T \cup U), T \cup U) .$$

It is obvious that $+$ is commutative, associative and idempotent. The *restriction* of a theory to a signature is defined as

$$\Sigma \,\square\, T =_{\text{def}} Cl(\Sigma \cap S(T), T) \; (= \{A \mid A \in T \text{ and } S(A) \subseteq \Sigma\}) .$$

We also define the trivial theory over a signature:

$$T(\Sigma) =_{\text{def}} Cl(\Sigma, \emptyset) \ (= \{A \mid \vdash A \text{ and } S(A) \subseteq \Sigma\}) \ .$$

Renamings are finitely generated mappings defined on expressions of L, changing only signature elements and commuting with taking types (i.e. the type of a renamed signature element is the renaming of the type of that signature element). Observe that we are liberal in the definition of renamings in the sense that they need not be bijective, so e.g. $[P := R, Q := R]$ is a correct renaming (if P, Q and R have the same type); this is in contrast to [2], where renamings are bijective and even involutive (i.e. if $\rho(P) = Q$ then $\rho(Q) = P$). We define domain and range of a renaming by

$$\begin{aligned} dom(\rho) &=_{\text{def}} \{I \mid \rho(I) \neq I\} \\ rg(\rho) &=_{\text{def}} \{\rho(I) \mid \rho(I) \neq I\} \ (= \rho(dom(\rho))) \end{aligned}$$

All renamings are finitely generated, so domain and range of a renaming are finite and hence signatures. We shall use some injectivity properties of renamings, defined by (here Σ and Π are arbitrary, i.e. not necessarily closed)

$$\begin{aligned} inj(\rho, \Sigma, \Pi) &=_{\text{def}} \text{for all } I \in \Sigma, J \in \Pi, \text{ if } \rho(I) = \rho(J) \text{ then } I = J \\ inj(\rho, \Sigma) &=_{\text{def}} inj(\rho, \Sigma, \Sigma) \\ inj(\rho) &=_{\text{def}} inj(\rho, dom(\rho)) \end{aligned}$$

We also put, for arbitrary Σ, Π:

ρ *renames* Σ *outside* Π iff $dom(\rho) = \Sigma \cap \Pi$, $rg(\rho) \cap (\Sigma \cup \Pi) = \emptyset$ and $inj(\rho, \Sigma)$.

We shall use the *good renaming property*:

for arbitrary Σ, Π there is a renaming $\gamma(\Sigma, \Pi)$ renaming Σ outside Π,

which follows directly from the finiteness of signatures and the fresh signature element principle. The application of renamings on theories is defined by

$$\rho(T) =_{\text{def}} Cl(\rho(S(T)), \{\rho(A) \mid A \in T\}) \ .$$

Observe that $\rho(T) = \{\rho(A) \mid A \in T\}$ only if $inj(\rho, S(T))$, e.g. if $\rho = [P := Q]$ and $T = Cl(\{P, Q\}, \{P, \neg Q\})$ then $\perp \in \rho(T)$, $\perp \notin \{\rho(A) \mid A \in T\}$.

3 Properties

We now list some properties of the operations defined above. Completeness is not our aim and we restrict ourselves to the properties used in the proof of the normal form theorem in the next section; most of them deal with permutation of two operators. We let T, U, V range over theories, Σ and Π over (closed) signatures, ρ and σ over renamings.

$$
\begin{array}{ll}
(Sr) & S(\rho(T)) = \rho(S(T)) \\
(S+) & S(T+U) = S(T) \cup S(U) \\
(S\Box) & S(\Sigma \,\Box\, T) = \Sigma \cap S(T) \\
(rr) & \rho(\sigma(T)) = (\rho \cdot \sigma)(T) \\
(r+) & \rho(T+U) = \rho(T) + \rho(U) \\
(r\Box) & inj(\rho, S(T) - \Sigma, S(T) \cup \Sigma) \Rightarrow \rho(\Sigma \,\Box\, T) = \rho(\Sigma) \,\Box\, \rho(T) \\
(\Box r) & dom(\rho) \cap \Sigma \cap S(T) = \emptyset \;\&\; \rho(S(T) - \Sigma) \cap \Sigma = \emptyset \;\&\; inj(\rho, S(T)) \\
 & \Rightarrow \Sigma \,\Box\, T = \Sigma \,\Box\, \rho(T) \\
(\Box S) & T = S(T) \,\Box\, T \\
(\Box\Box) & \Sigma \,\Box\, (\Pi \,\Box\, T) = (\Sigma \cap \Pi) \,\Box\, T \\
(\Box +) & S(T) \cap S(U) \subseteq \Sigma \Rightarrow \Sigma \,\Box\, (T+U) = \Sigma \,\Box\, T + \Sigma \,\Box\, U \\
(\gamma) & dom(\gamma(\Sigma, \Pi)) = \Sigma \cap \Pi \;\&\; rg(\gamma(\Sigma, \Pi)) \cap (\Sigma \cup \Pi) = \emptyset \;\&\; inj(\gamma(\Sigma, \Pi), \Sigma)
\end{array}
$$

The axioms (Sr), $(S+)$, $(r+)$, $(\Box S)$, $(\Box\Box)$ and $(\Box+)$ are also found in [2]. The condition in $(r\Box)$ is required to prevent new identifications of names in $\rho(T)$ and between names in $\rho(\Sigma)$ and $\rho(T)$ not present in $\rho(\Sigma \,\Box\, T)$; the condition in $(\Box r)$ guarantees that ρ does not affect names in $\Sigma \,\Box\, T$, neither introduces new identifications in T. Most of the properties listed have a straightforward proof; only the proof of $(\Box+)$ is more involved and follows now.

Lemma 1. *Property $(\Box+)$ holds iff L satisfies the following version of the Interpolation Theorem:*

(*) if $\Gamma \cup \Delta \vdash A$ then there is an interpolant I with
$\Gamma \vdash I$, $\Delta \cup \{I\} \vdash A$ and $S(I) \subseteq S(\Gamma) \cap (S(\Delta) \cup S(A))$.

Proof sketch of the 'if' part (the other part is trivial; see [2] for more details). Assume $S(T) \cap S(U) \subseteq \Sigma$ and let $A \in \Sigma \,\Box\, (T+U)$, i.e. $T \cup U \vdash A$ and $S(A) \subseteq \Sigma \cap (S(T) \cup S(U))$. Now use the interpolant I with $T \vdash I, U \cup \{I\} \vdash A$ to see (after some rewriting of parameter conditions) that $Cl(\Sigma, T) \vdash I$ and $Cl(\Sigma, U) \vdash I \rightarrow A$, hence $Cl(\Sigma, T) \cup Cl(\Sigma, U) \vdash A$. Conclusion: $A \in \Sigma \,\Box\, T + \Sigma \,\Box\, U$. This proves the $\subseteq$-part of the equality, the other inclusion is trivial. $\Box$

Remark 1. There are several variants of the axiom $(\Box+)$:

$$
\begin{array}{ll}
(\Box+1) & \Sigma \,\Box\, (T \cup U) = \Sigma \,\Box\, ((\Sigma \cup S(U)) \,\Box\, T + (\Sigma \cup S(T)) \,\Box\, U) \\
(\Box+2) & (S(T) \cap S(U)) \,\Box\, (T+U) = S(U) \,\Box\, T + S(T) \,\Box\, U \\
(\Box+3) & S(T) \,\Box\, (T+U) = T + S(T) \,\Box\, U \\
(\Box+4) & \Sigma \,\Box\, (T(\Sigma) + T) = T(\Sigma) + \Sigma \,\Box\, T \\
(\Box+5) & \Sigma \,\Box\, (T(\Pi) + T) = T(\Sigma \cap \Pi) + \Sigma \,\Box\, T
\end{array}
$$

These equations have in common that they all require some version of the interpolation theorem for the underlying logic (see below). We have the following relations:

$$(\Box+) \Leftrightarrow (\Box+1) \Leftrightarrow (\Box+2) \Leftrightarrow (\Box+3) \Rightarrow (\Box+4) \Leftrightarrow (\Box+5)\ .$$

For proofs, we refer to [2] for $(\Box+2) \Rightarrow (\Box+)$, [12] for $(\Box+3) \Rightarrow (\Box+2)$ and [15] and [14] for $(\Box+4) \Rightarrow (\Box+5)$; the other implications are fairly easy (for $(\Box+) \Rightarrow (\Box+1)$, use that $\Sigma \,\Box\, (T \cup U) = \Sigma \,\Box\, (\Sigma \cup (S(T) \cap S(U)) \,\Box\, (T+U))$.

Remark 2. Interpolation as formulated in (*) holds for predicate logic, but not for equational logic (EL) neither for conditional equational logic (CEL). CEL has the easiest counterexample:

$$\Gamma = \{fc = c\},\ \Delta = \{fx = x \to a = b\},\ A = (a = b)$$

If I were an interpolant then $S(I) \subseteq \{f\}$, $fc = c \vdash I$ and we claim $I \vdash \exists x(fx = x)$; this would lead to definability of existential quantification in CEL, *quod non*. So it remains to prove the claim: observe $I \vdash (\forall x(fx = x) \to a = b) \to a = b$, hence $I \vdash \exists x(fx = x) \vee a = b$; if M is a model for the signature $\{f\}$ and $M \models I$, then either M has only one element and $M \models \exists x(fx = x)$, or M has at least two elements and we can expand M to a model M' for the signature $\{f, a, b\}$ with $M' \models a \neq b$, so $M' \models \exists x(fx = x)$ and hence $M \models \exists x(fx = x)$. This proves the claim. This counterexample to (*) in CEL can be transferred to EL using the embedding of CEL into EL given in [2, 5.2.1]. The essence of this embedding is given by

$s = s' \to t = t'$ becomes $\{g(x, x, y) = y, g(s, s', t) = g(s, s', t')\}$
(g a fresh function);

it leads to the counterexample $\Gamma = \{fc = c\}$, $\Delta = \{g(x, x, y) = y, g(x, fx, a) = g(x, fx, b)\}$, $A = (a = b)$ that is given in [2] and [10]. Another counterexample is given in [7]: $\Gamma = \{f(gx) = x\}$, $\Delta = \{f(fx) = fx\}$, $A = (fx = x)$: none of the possible interpolants (e.g. $\forall x \exists y(fy = x)$ or $f(fx) = fx \to fx = x$) is in EL.

Remark 3. Logics EL and CEL do satisfy a weaker interpolation theorem, viz. (*) with $\Delta = \emptyset$ and I a finite conjunction of formulas. This has been proved for EL by Rodenburg and Van Glabbeek in [15] (see also [8] and [13]) and for CEL by Rodenburg in [14]. It turns out that this weaker version of interpolation corresponds exactly with the axiom ($\Box$+4) (or, equivalently, ($\Box$+5)), as is proved in [15] and [14].

Remark 4. If one replaces the theory semantics by the model class semantics, the axiom ($\Box$+) follows directly and the interpolation theorem is not required. See [2].

4 Normal Forms

Now we consider normal forms of expressions involving theories, $\Box$, $+$ and renamings. Define the language TL of *theory terms* by

$$T ::= Th \mid \rho(T) \mid \Sigma \,\Box\, T \mid T + T\ ,$$

where Th denotes constants $Th_1, Th_2, \ldots$ referring to specific theories. We shall show the following Normal Form Theorem: any expression of TL is equivalent to a normal form expression

$$\Sigma \,\Box\, (\rho_1(Th_1) + \cdots + \rho_n(Th_n))\ .$$

To prove this, it suffices to show that the collection of normal forms contains the constants Th_i and is closed under renamings, $\Box$ and $+$. Now $Th_i = S(Th_i) \,\Box\, \rho_i(Th_i)$ which is in normal form, and closure under $\Box$ follows from ($\Box\Box$); for closure under renamings and $+$ we use the next lemma.

Lemma 2. *i) For any Σ, T, ρ there is a renaming σ with*

$$\rho(\Sigma \,\Box\, T) = \rho(\Sigma) \,\Box\, \sigma(T) \ .$$

ii) For any Σ, Π, T, U there are renamings ρ, σ with

$$\Sigma \,\Box\, T + \Pi \,\Box\, U = (\Sigma \cup \Pi) \,\Box\, (\rho(T) + \sigma(U)) \ .$$

Proof. i) Property $(r\Box)$ cannot be applied directly, therefore we put $\tau = \gamma(S(T) - \Sigma, S(T) \cup \Sigma \cup dom(\rho) \cup rg(\rho))$, so

$$\begin{aligned}
&dom(\tau) = S(T) - \Sigma,\\
&rg(\tau) \cap (S(T) \cup \Sigma \cup dom(\rho) \cup rg(\rho)) = \emptyset,\\
&inj(\tau, S(T) - \Sigma)
\end{aligned}$$

and one straightforwardly verifies the conditions of $(\Box r)$ (reading τ for ρ), so we have $\Sigma \,\Box\, T = \Sigma \,\Box\, \tau(T)$, hence $\rho(\Sigma \,\Box\, T) = \rho(\Sigma \,\Box\, \tau(T))$. Now the result follows if we can apply $(r\Box)$ to this last term, so we have to check $inj(\rho, S(\tau(T)) - \Sigma, S(\tau(T)) \cup \Sigma)$, i.e. (using the fact that $rg(\tau) = S(\tau(T)) - \Sigma$) $inj(\rho, rg(\tau), rg(\tau) \cup \Sigma)$. To see this, assume $I \in rg(\tau)$, $J \in rg(\tau) \cup \Sigma$, $\rho(I) = \rho(J)$; since $rg(\tau) \cap (dom(\rho) \cup rg(\rho)) = \emptyset$, we have $\rho(I) = I \notin rg(\rho)$ and ($\rho(J) = J$ or $J \in dom(\rho)$); the second alternative yields contradiction with $\rho(I) = \rho(J)$ and $\rho(I) \notin rg(\rho)$, so we conclude $I = J$ and the injectivity condition is proved. So we have $\rho(\Sigma \,\Box\, T) = \rho(\Sigma) \,\Box\, \sigma(T)$ with $\sigma = \rho \cdot \tau$.

ii) Let $\rho = \gamma(S(T) \cap \Pi - \Sigma, S(T) \cup \Sigma \cup \Pi)$, then $dom(\rho) = S(T) \cap \Pi - \Sigma$, $rg(\rho) \cap (S(T) \cup \Sigma \cup \Pi) = \emptyset$ and $inj(\rho, S(T) \cap \Pi - \Sigma)$, so

$$\begin{aligned}
\Sigma \,\Box\, T &= \Sigma \,\Box\, \rho(T) && (\text{ by } (\Box r))\\
&= \Sigma \cap S(\rho(T)) \,\Box\, \rho(T) && (\text{ by } (\Box S) \text{ and } (\Box\Box))\\
&= (\Sigma \cup \Pi) \cup S(\rho(T)) \,\Box\, \rho(T) && (\Pi \cap S(\rho(T)) \subseteq \Sigma)\\
&= (\Sigma \cup \Pi) \,\Box\, \rho(T) && (\text{ by } (\Box S) \text{ and } (\Box\Box))
\end{aligned}$$

Analogously: if $\rho' = \gamma(S(U) \cap \Sigma - \Pi, S(U) \cup \Sigma \cup \Pi)$, then $\Pi \Box U = (\Sigma \cup \Pi) \Box \rho'(U)$. Now put $\tau = \gamma(S(\rho(T)) \cap S(\rho'(U)) - (\Sigma \cup \Pi), S(\rho(T)) \cup S(\rho'(U)) \cup \Sigma \cup \Pi)$, then

$$\begin{aligned}
&dom(\tau) = S(\rho(T)) \cap S(\rho'(U)) - (\Sigma \cup \Pi),\\
&rg(\tau) \cap (S(\rho(T)) \cup S(\rho'(U)) \cup \Sigma \cup \Pi) = \emptyset,\\
&inj(\tau, S(\rho(T)) \cap S(\rho'(U)) - (\Sigma \cup \Pi)).
\end{aligned}$$

After some calculation we see $dom(\tau) \cap (\Sigma \cup \Pi) \cap S(\rho'(U)) = \emptyset, \tau(S(\rho'(U) - (\Sigma \cup \Pi)) \cap (\Sigma \cup \Pi) = \emptyset$ and $inj(\tau, S(\rho'(U)))$. Now

$$\begin{aligned}
\Sigma \,\Box\, T + \Pi \,\Box\, U &= (\Sigma \cup \Pi) \,\Box\, \rho(T) + (\Sigma \cup \Pi) \,\Box\, \rho'(U)\\
&= (\Sigma \cup \Pi) \,\Box\, \rho(T) + (\Sigma \cup \Pi) \,\Box\, \tau \cdot \rho'(U) && (\text{ by } (\Box r) \text{ and } (rr))\\
&= (\Sigma \cup \Pi) \,\Box\, (\rho(T) + \tau \cdot \rho'(U)) && (\text{ by } (\Box +))\\
&= (\Sigma \cup \Pi) \,\Box\, (\rho(T) + \sigma(U))
\end{aligned}$$

where $\sigma = \tau \cdot \rho'$. $\Box$

Now we can finish the proof of the Normal Form Theorem. Closure of the collection of normal forms under renamings follows from Lemma 2(i), $(r+)$ and (rr); closure under $+$ follows from Lemma 2(ii), $(r+)$, (rr) and the properties of $+$.

5 Two Orthogonal Closures

It is possible to define the operations on theories presented above (except renaming) in terms of two orthogonal closure operators: signature closure s and theory closure t. They are defined on the collection of *all* sets of sentences (i.e. not only sets closed w.r.t. derivability, as in Sect. 2). The definitions read (X, Y, Z range over sets of sentences, i.e. subsets of L):

$$sX =_{\text{def}} \{A \mid S(A) \subseteq S(X)\}$$
$$tX =_{\text{def}} \{A \mid X \vdash A\}$$

Lemma 3. *The following hold:*

$(s1)$ $X \subseteq sX$
$(s2)$ $ssX = sX$
$(s3)$ $X \subseteq Y \Rightarrow sX \subseteq sY$
$(s4)$ $s(sX \cap sY) = sX \cap sY$
$(s5)$ $s(X \cup sY) = s(X \cup Y)$

$(t1)$ $X \subseteq tX$
$(t2)$ $ttX = tX$
$(t3)$ $X \subseteq Y \Rightarrow tX \subseteq tY$
$(t4)$ $t(tX \cap tY) = tX \cap tY$
$(t5)$ $t(X \cup tY) = t(X \cup Y)$

$(st1)$ $stX = tsX = L$
$(st2)$ $s(sX \cap tY) = sX$
$(st3)$ $t(sX \cap tX) = tX$
$(st4)$ $sX \cap t(sX \cap tY) = sX \cap tY$
$(st5)$ $sX \cap (tY \cup tZ) = sX \cap t((tY \cap sY \cap s(X \cup Z)) \cup (tZ \cap sZ \cap s(X \cup Y)))$

Proof (sketch). Verifying $(s1-3)$ is easy; moreover, they imply $(s4-5)$. Similarly for $(t1-5)$.
$(st1)$: use $\{\varphi \to \varphi \mid \varphi \in L\} \subseteq t\emptyset$ and $\bot \in s\emptyset$.
$(st2)$: use $\{\varphi \to \varphi \mid \varphi \in sX\} \subseteq t\emptyset$.
$(st3)$: by $(s1), (t1-3)$ we have $tX \subseteq t(sX \cap tX) \subseteq ttX = tX$.
$(st4)$: by $(t1-3)$ we have $sX \cap tY \subseteq sX \cap t(sX \cap tY) \subseteq sX \cap ttY = sX \cap tY$.
$(st5)$: analogous to the proof of $(\Box{+}1)$, requiring the Interpolation Theorem. $\Box$

By the definition of Cl in Sect. 2 it is obvious that X is a theory iff it satisfies $X = sX \cap tX$. The definition of the operations on theories introduced in Sect. 2 now reads:

$$T(X) =_{\text{def}} sX \cap t\emptyset$$
$$X \Box Y =_{\text{def}} sX \cap Y$$
$$X + Y =_{\text{def}} s(X \cup Y) \cap t(X \cup Y)$$

Now $(S+), (S\Box), (\Box S), (\Box\Box)$ and $(\Box+)$ follow from the previous Lemma; for $(\Box+)$, use the instance of $(st5)$ obtained by taking $Z = \emptyset$.

6 First-Order Semantics for a Second-Order Fragment

A natural logical interpretation of the notions of combination and hiding is: conjunction and existential quantification. So let us consider the collection $L2$ of second-order formulae, characterised by

$$A ::= \varphi \mid A\&A \mid \exists f A$$

where φ stands for first-order formulae (elements of L) and f for signature elements. We consider the following interpretation:

$$\begin{aligned}
[\![\varphi]\!] &=_{\text{def}} \{\psi \mid \varphi \vdash \psi \text{ and } S(\psi) \subseteq S(\varphi)\} \\
[\![A\&B]\!] &=_{\text{def}} \{\varphi \mid [\![A]\!] \cup [\![B]\!] \vdash \varphi \text{ and } S(\varphi) \subseteq S(A) \cup S(B)\} \\
[\![\exists f A]\!] &=_{\text{def}} \{\varphi \mid \varphi \in [\![A]\!] \text{ and } f \notin S(\varphi)\}
\end{aligned}$$

Validity is defined by

$$\begin{aligned}
A \models B &=_{\text{def}} [\![A]\!] \supseteq [\![B]\!] \\
A \equiv B &=_{\text{def}} A \models B \text{ and } B \models A
\end{aligned}$$

It is obvious that $[\![A]\!]$ is always a theory (i.e. of the form $Cl(\Sigma, X)$) and that

$$\begin{aligned}
[\![A\&B]\!] &= [\![A]\!] + [\![B]\!] \\
[\![\exists f A]\!] &= (S[\![A]\!] - \{f\}) \,\Box\, [\![A]\!]
\end{aligned}$$

Moreover, assuming that L has the substitution property, i.e.

if $\Gamma \vdash \varphi$ then $\Gamma[f := \lambda x.t] \vdash \varphi[f := \lambda x.t]$
(for unary function symbols f; analogously for other signature elements),

we also have

$$[\![A[f := \lambda x.t]]\!] \supseteq [\![A]\!][f := \lambda x.t] \ .$$

Lemma 4. *The following hold:*

$$\begin{aligned}
&(\&L) && A\&B \models A, B \\
&(\&R) && \textit{if } A \models B \textit{ and } A \models C \textit{ then } A \models B\&C \\
&(\exists R) && A[f := \lambda x.t] \models \exists f A \\
&(\exists L) && \textit{if } A \models B \textit{ and } f \textit{ not free in } B \textit{ then } \exists f A \models B \\
&(\&\exists) && \textit{if } f \textit{ not free in } A \textit{ then } A\&\exists f B \equiv \exists f(A\&B)
\end{aligned}$$

Proof (sketch). $(\&L)$: easy, use $S[\![A]\!] = S(A)$.

$(\&R)$: if $\varphi \in [\![B\&C]\!]$ then there are $\psi \in [\![B]\!], \chi \in [\![C]\!]$ with $\psi \wedge \chi \vdash \varphi$. Hence $\psi, \chi \in [\![A]\!]$, but $[\![A]\!]$ is closed, so $\varphi \in [\![A]\!]$.

$(\exists R)$: we only have to show $[\![\exists f A]\!] \subseteq [\![A]\!][f := \lambda x.t]$, which follows directly from the substitution property for L.

$(\exists L)$: straightforward.

$(\&\exists)$: follows from $(\Box+)$. As a consequence, $(\&\exists)$ requires the Interpolation Theorem for L. $\Box$

6.1 Remark.

A natural question is: can the collection of formulae used for the definition of the interpretation of $L2$ not be replaced by a single formula? This is what Pitts actually does in [9] for second-order intuitionistic propositional logic. In order to do so, he proves a uniform interpolation theorem for first-order intuitionistic propositional logic (IpL), viz.

> if A is a formula of IpL and $\Sigma \subseteq S(A)$, then there is an $I = I(A, \Sigma)$ satisfying $A \vdash I$, $S(I) \subseteq \Sigma$ and ($A \vdash B$ and $\Sigma \subseteq S(A) \cap S(B)$ imply $I \vdash B$).

However, uniform interpolation does not hold in predicate logic, as has been observed by Henkin in [6]. As a consequence, it remains a nontrivial question how to extend the interpretation $[\![\cdot]\!]$ to negation and, via negation, to the other logical operations.

References

1. R.M. Burstall and J.A. Goguen, The semantics of CLEAR, a specification language, in: Abstract software specifications, LNCS 86, Springer-Verlag (1980) 292–332
2. J.A. Bergstra, J. Heering, P. Klint, Module algebra, Journal of the ACM **37** (1990) 335–372
3. H. Ehrig and B. Mahr, Fundamentals of algebraic specification 1, Equations and initial semantics, Springer-Verlag, 1985
4. H. Ehrig and B. Mahr, Fundamentals of algebraic specifications 2, Module specifications and constraints, Springer-Verlag, 1990
5. M.-C. Gaudel, Toward structured algebraic specifications, in: Esprit '85: Status report of continuing work, vol. 1, North-Holland (1986) 493–510
6. L. Henkin, An extension of the Craig-Lyndon interpolation theorem, The Journal of Symbolic Logic **28** (1963) 201–216
7. T.S.E. Maibaum and M.R. Sadler, Axiomatising specification theory, in: H.-J. Kreowski (ed.), Recent trends in data type specification, 3rd Workshop on Theory and Applications of Abstract Data Types, Informatik-Fachberichte 116, Springer-Verlag (1985) 171–177
8. D. Pigozzi, The join of equational theories, Colloquium Mathematicum **30** (1974), 15–25
9. A.M. Pitts, On an interpretation of second order quantification in first order intuitionistic propositional logic, to appear in The Journal of Symbolic Logic
10. G.R. Renardel de Lavalette, Modularisation, parametrisation, interpolation, Journal of Information Processing and Cybernetics EIK **25** (1989) 283–292
11. G.R. Renardel de Lavalette, COLD-K^2, the Static Kernel of COLD-K, Report RP/mod-89/8, Software Engineering Research Centre, Utrecht, 1989
12. P.H. Rodenburg, On the ω-completeness of a fragment of module algebra, manuscript (1990)
13. P.H. Rodenburg, A simple algebraic proof of the equational interpolation theorem, Algebra Universalis **28** (1991) 48–51
14. P.H. Rodenburg, Interpolation in conditional equational logic, Fundamenta Informaticae **15** (1991) 80–85
15. P.H. Rodenburg, Interpolation in equational logic, Report P9201, Programming Research Group, University of Amsterdam, 1992

16. M. Wirsing, Structured algebraic specifications: a kernel language, Habilitationsschrift, Institut für Informatik, Technische Universität, München, 1983

This article was processed using the LaTeX macro package with LLNCS style