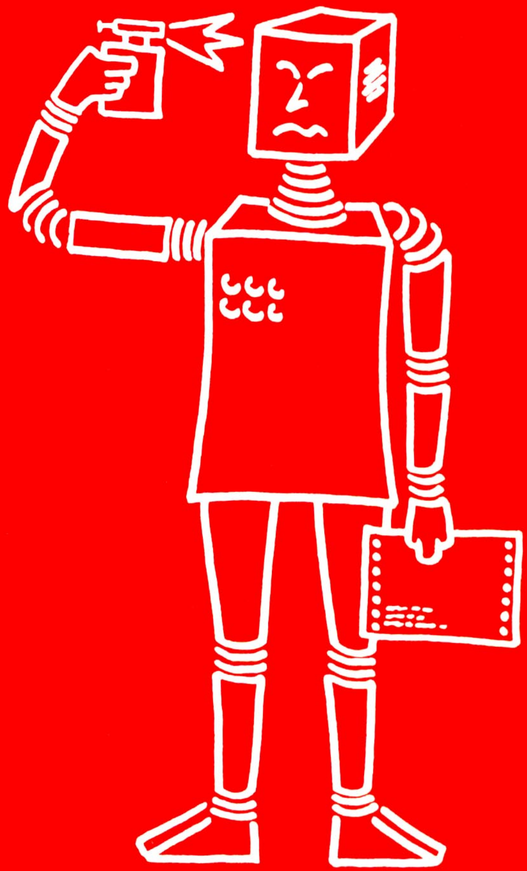
SEKI-PROJEKT

SEKI MEMO

Universität Kaiserslautern
Fachbereich Informatik
Postfach 3049
D-6750 Kaiserslautern
West Germany

Universität Karlsruhe
Institut für Informatik I
Postfach 6380
D-7500 Karlsruhe
West Germany



A MANY-SORTED CALCULUS WITH POLYMORPHIC FUNCTIONS BASED ON RESOLUTION AND PARAMODULATION

M. Schmidt-Schauß

MEMO SEKI-85-II-KL

Interner Bericht 1985

A Many-sorted Calculus with Polymorphic Functions Based on Resolution and Paramodulation.

Manfred Schmidt-Schauss
Fachbereich Informatik
Universität Kaiserslautern

ABSTRACT.

A many-sorted first order calculus, called ΣRP, whose well formed formulas are sorted (typed) clauses and whose inference rules are factorization, resolution, paramodulation and weakening is extended to a many sorted calculus ΣRP* with polymorphic functions (overloading). It is assumed that the sort structure is a finite partially ordered set with a greatest element. It is shown, that this extended calculus is sound and complete, provided the functional reflexivity axioms are present. It is also shown, that unification of terms containing polymorphic functions is in general finitary, i.e. the set of most general unifiers may contain more than one element, but at most finitely many.

We give a natural condition for the signature (the sort structure), such that the set of most general unifiers is always at most a singleton provided this condition holds.

Contents.

Introduction.

The advantages of a many-sorted calculus in automated reasoning systems are well known [Hay71, Hen72, Wa83, GM84, GM85, Co83, CD83, Ob62]. These advantages were also noticed in logical programming [MO84, Mi84]:

In a many-sorted calculus we obtain a shorter refutation of a smaller set of shorter clauses, as compared to the unsorted version.

Our interest is in using a many sorted calculus in a resolution based Automated Theorem Proving-system (ATP). The most desirable properties of such a calculus are:

1.) The unsorted problem and the corresponding sorted problem are equivalent, i.e. the unsorted clause set is unsatisfiable, iff the sorted version is unsatisfiable.

2.) The calculus is complete, i.e. there is a derivation of the empty clause, iff the clause set is unsatisfiable.

3.) The search space in the sorted version of a problem is smaller than the search space of the unsorted version (provided the problem has a sort structure).

4.) The many-sorted calculus has as much expressive power as possible.

5.) The calculus should be based on standard resolution and paramodulation [WR73] possibly augmented by a modified unification algorithm. Hence standard reductions like purity, subsumption, tautology deletion, replacement resolution and incompatibility of unifiers [KM84, Ro64] can still be used. Furthermore resolution based strategies such as Unit resolution and Set-of-support should be applicable.

The ΣRP-calculus of C.Walther [Wa83] essentially satisfies these requirements, but can be improved by the additional incorporation of polymorphic functions (overloading [MO84]). This new extended calculus is called ΣRP*.

In ΣRP* it is possible for example to have a function symbol + (sum) denoting the addition of (complex) numbers with the implicitly stated property, that syntactically, the sum of integers is an integer, the sum of reals is a real and the sum of Gaussian numbers is a Gaussian number. However we do not allow to use the same symbol "+" for say the addition of vectors, since we have the technical restriction, that for every argument position in every function there exists a greatest sort for that argument. Using "+" for numbers and vectors would imply that e.g. 1+(0,0) is well sorted, which does not make sense.

Without this restriction, the ΣRP*-calculus is not complete in general. (This however could be remedied if either ill-sorted terms are allowed during equality deductions or an extended parallel paramodulation rule is used).

The results presented in this paper concern unification of polymorphic terms and completeness of the ΣRP*-calculus. In particular it is shown, that the complete and minimal set of most general unifiers for two polymorphic terms is always finite. Some authors present sort structures, such that the union, the intersection and the complement of sorts are well defined [Co83, CD83]. If such information is used in a deduction, the rules of the ΣRP*-calculus are not sufficient and extra rules would be necessary to ensure completeness. But such extra rules have in general the very unpleasent side effect that the reductions and strategies of a resolution based calculus are no longer applicable.

1. Basic Notions of the ΣRP^*-calculus.

The ΣRP^*-calculus is an extension of the ΣRP-calculus [Wa83] by polymorphic functions.

In the ΣRP-calculus the sort of a term is fully determined by it's outermost function symbol. In contrast, the sort of a term in the ΣRP^*-calculus depends dynamically on the sorts of the arguments of the outermost function symbol.

The following definition of a signature is close to [GM84], but extended to polymorphic functions.

We use in this definition a set of n+1-tuples for the type information of an n-ary function. The first n symbols of a tuple give the sort of the arguments and the $(n+1)^{th}$ symbol gives the sort of the corresponding value. E.g. for the sum (+) of complex numbers we have:

+ : COMPLEX × COMPLEX → COMPLEX ;
COMPLEX × INT → COMPLEX ;
INT × INT → INT ; ...

This is denoted as a set of triples {(COMPLEX,COMPLEX,COMPLEX),, (COMPLEX,INT,COMPLEX), (INT,INT,INT) . . .}.

1.1 Definition.: A polymorphic signature SIG is a triple $(\mathbb{S},\mathbb{F},\mathbb{P})$, where

1) $\mathbb{S}$ is the finite set of sorts, $\leq$ is a partial ordering on $\mathbb{S}$ with the greatest element $\top$. $\leq$ is extended to tuples of sorts in the usual way (componentwise $\leq$).

2) $\mathbb{F}$ is the set of function symbols. $\mathbb{F} = \bigcup \mathbb{F}_W$, where $\mathbb{F}_W$ is a set of function symbols of arity n with $\emptyset \neq W \subseteq \mathbb{S}^{n+1}$.

   If $\mathbb{F}_W \neq \emptyset$, then W satisfies the following conditions:

   - The sort of constants is unique. i.e. $W \subseteq \mathbb{S}$ implies $|W| = 1$.
   - For signatures W of functions, which are not constant:

   a) W contains a unique greatest element $(S_{W,1},S_{W,2},\ldots,S_{W,n+1})$.

   b) For every $(S_1, S_2,\ldots, S_{n+1}) \in W$ and every $(T_1,\ldots,T_n) \in \mathbb{S}^n$, $(T_1,\ldots,T_n) \leq (S_1,\ldots,S_n)$ implies that there exists a unique sort $T_{n+1} \leq S_{n+1}$ such that $(T_1,\ldots,T_{n+1}) \in W$.

   i.e. for every $f \in \mathbb{F}$ the related function

   $f^* : \{(S_1,\ldots,S_n) \mid (S_1,\ldots,S_n) \leq (S_{W,1},\ldots,S_{W,n})\} \rightarrow \mathbb{S}$, where

   $f^*(S_1,\ldots,S_n) = S_{n+1}$, iff $(S_1,\ldots,S_{n+1}) \in W$, is well defined and monotone.

3) $\mathbb{P}$ is the set of predicate symbols. $\mathbb{P}_D$ is the set of predicates with domain D, where $D \in \mathbb{S}^n$. We have $\mathbb{P} = \bigcup \mathbb{P}_D$.

4) For every sort $S \in \mathbb{S}$, there exists a constant c of sort $S_c \leq S$. That means, that every sort is strict or SIG is sensible in the sense of [HO80]. □

We use the following additional notation and abbreviations:

$SO(f) = \mathbb{W}$, iff $f \in \mathbb{F}_{\mathbb{W}}$;

$SO(P) = V$, iff $P \in \mathbb{P}_V$.

$\mathbb{C}$ denotes the set of all constants,

$\mathbb{C}_S$ denotes the set of all constants of sort S.

$\mathbf{V}$ denotes the set of all variables.

$\mathbf{V}_S$ is the (infinite) set of variables of sort S.

$R \sqcap S := \{T \in \mathbb{S} \mid T \leq R \text{ and } T \leq S\}$.

If the set of ranges of a function f, i.e. the set $\{S_{n+1} | (S_1,...,S_{n+1}) \in SO(f)\}$ has more than one element, then f is called a polymorphic function. If no polymorphic function is in $\mathbb{F}$, then SIG corresponds to a signature of the ΣRP-calculus [Wa83].

In an actual implementation, it is not necessary to specify the whole signature of a function explicitely. It is sufficient to give enough information to compute the signature of a function uniquely. For functions, which are not polymorphic, the specification of the maximal domain and maximal range suffices.

The following is the standard definition of a heterogeneous algebra (see e.g. [HO80]) with the additional proviso that the subsort relation is represented as the subset relation.:

1.2 Definition. Let SIG = $(\mathbb{S}, \mathbb{F}, \mathbb{P})$ be a signature. The pair (A, SIG) is called an algebra of type SIG, iff

i) A is a nonempty set.

ii) For every sort $S \in \mathbb{S}$, there is a subset S^A of A such that for all $R,S \in \mathbb{S}$: $R \leq S \Rightarrow R^A \subseteq S^A$. Furthermore $\top^A = A$ for the top sort $\top$.

iii) For $c \in \mathbb{C}_S$ there exists an element $c^A \in A$, such that $c^A \in S^A$.

iv) For $f \in \mathbb{F} \setminus \mathbb{C}$: $f^A: A^n \to A$ is a mapping, such that for every $(S_1,...,S_{n+1}) \in SO(f)$ and every $a_i \in S_i^A$, i=1,...,n, we have $f^A(a_1,...,a_n) \in S_{n+1}{}^A$. □

By the definition of a signature, we have that $S^A \neq \emptyset$ for every $S \in \mathbb{S}$, since every sort contains a constant.

Note that this definiton is independent from the set of predicates $\mathbb{P}$.

We extend the usual notion of a homomorphism to a SIG-homomorphism, which respects the sort structure:

1.3 Definition. Let (A,SIG) and (B,SIG) be two algebras of type SIG. Then a mapping φ: A → B is called a SIG-homomorphism, iff

i) $\varphi(S^A) \subseteq \varphi(S^B)$ for all $S \in \mathbb{S}$.

ii) $\varphi(f^A(a_1,\dots,a_n)) = f^B(\varphi(a_1),\dots,\varphi(a_n))$ for all $f \in \mathbb{F}$ and all $a_i \in S_{f,i}{}^A$, $i=1,\dots,n$ where $(S_{f,1},\dots,S_{f,n})$ is the greatest element of SO(f). □

Obviously, the composition of two SIG-homomorphism is again a SIG-homomorphism.

Let $\mathbf{T} = \mathbf{T}(\mathbb{F},\mathbf{V})$ be the set of all terms (including ill-sorted terms); i.e. $\mathbf{T}$ is the least set with $\mathbf{V} \subseteq \mathbf{T}$, $\mathbf{C} \subseteq \mathbf{T}$, and $f(t_1,\dots,t_n) \in \mathbf{T}$ for all $f \in \mathbb{F}$ and all $t_i \in \mathbf{T}$.

The sort of a term is defined similar to [Wa83], but adapted to polymorphic functions:

1.4 Definition. Let SIG= $(\mathbf{S}, \mathbf{F}, \mathbf{P})$ be a signature.

The sort of a term t, namely [t], is defined by the partial mapping $[\dots]: \mathbf{T} \to \mathbf{S}$:

$$[t] = \begin{cases} S & \text{if } t \in \mathbf{V}_S \text{ or } t \in \mathbf{C}_S \\ S_{n+1} & \text{if } t = f(t_1,\dots,t_n) \text{ and } ([t_1],\dots,[t_n],S_{n+1}) \in SO(f) \\ \text{undefined else.} & \end{cases}$$ □

Example. Let $\mathbb{S}$ = {N, NZ, Z}, where N denotes the nonnegative integers, including 0, NZ denotes the positive integers and Z = {0}. Then N ≥ NZ and N ≥ Z. The function + has the following signature:

SO(+) = {(N,N,N); (N,NZ,NZ); (NZ,N,NZ); (Z,N,N); (N,Z,N); (Z,NZ,NZ); (NZ,Z,NZ); (Z,Z,Z); (NZ,NZ,NZ) }.

Then for example [0+0] = Z and [0+1] = NZ.

The set of all well-sorted terms, called $\mathbf{WST}$, is defined as the domain of [...], i.e the set of elements, where [...] is not undefined.

As usual, ($\mathbf{WST}$, SIG) is an algebra of type SIG with the following (termbuilding) operations:

i) $S^{\mathbf{WST}} = \{t \in \mathbf{WST} \mid [t] \le S\}$.

ii) $f^{\mathbf{WST}}(t_1,\dots,t_n) = f(t_1,\dots,t_n)$, if $t_i \in \mathbf{WST}$ for $i=1,\dots,n$

($\mathbf{WST}$, SIG) is called the free term-algebra of type SIG [Gr79]. The set of well-sorted terms without variables is denoted by $\mathbf{WST}_{gr}$. The algebra ($\mathbf{WST}_{gr}$, SIG) is the initial algebra [Gr79] of type SIG or the Herbrand Universe [CL71,Lo78]. This terminology is justified by the following lemmas:

1.5 Lemma. ($\mathbf{WST}$, SIG) is free:

Let SIG = ($\mathbf{S}$, $\mathbf{F}$, $\mathbf{P}$) be a signature and let (A,SIG) be an algebra.

Let $\psi_0 : \mathbf{V} \to A$ be a partial mapping, such that $\psi_0 x \in [x]^A$. Further let $\mathbf{V}_0$ be the domain of ψ_0.

Then there exists a SIG-homomorphism $\psi: \mathbf{WST} \to A$, such that $\psi|_{\mathbf{V}_0} = \psi_0$.

Moreover if $\mathbf{V} = \mathbf{V}_0$, then ψ is unique.

Proof. Define a mapping $\psi: \mathbf{V} \to A$ with $\psi|_{\mathbf{V}_0} = \psi_0$in the following way:

For every sort S, take a fixed element $d_S \in S^A$ and define $\psi x := d_S$, if $x \in \mathbf{V}_S \setminus \mathbf{V}_0$.

For a term $t = f(t_1,...,t_n)$ we define ψ recursively as $\psi t = f^A(\psi t_1,...,\psi t_n)$. To show, that this homomorphic extension of ψ_0 is a SIG-homomorphism ψ with $\psi|_{\mathbf{V}_0} = \psi_0$ it suffices to show, that $\psi(S^{\mathbf{WST}}) \subseteq S^A$ for all sorts S.

Proof by induction:

Base case. For $x \in \mathbf{V}_S$, we have $\psi x \in S^A$ by assumption and by construction of ψ. For $c \in \mathbf{C}_S$, we have $\psi c = c^A \in S^A$ by defintion 1.2.

Induction step. Let $t = f(t_1,...,t_n) \in S^{\mathbf{WST}}$. Then $t_i \in \mathbf{WST}$ for i=1,...,n . With $S_i = [t_i]$ we have $(S_1,...,S_n,S) \in SO(f)$. From the definiton of an algebra and from the definition of ($\mathbf{WST}$,SIG) it follows, that $f^A(\psi t_1,...,\psi t_n) \in S^A$. Hence $\psi t = f^A(\psi t_1,...,\psi t_n) \in S^A$. □

1.6 Lemma. ($\mathbf{WST}_{gr}$,SIG) is an initial algebra of type SIG, i.e. for every algebra (A,SIG) there exists a unique SIG-homomorphism $\varphi: \mathbf{WST}_{gr} \to A$.

Proof. We define a mapping $\varphi: \mathbf{WST}_{gr} \to A$:

i) $\varphi c = c^A$ for constants c.

ii) $\varphi f(t_1,...,t_n) = f^A(\varphi t_1,...,\varphi t_n)$ for terms $t \in \mathbf{WST}_{gr}$.

The same arguments as in the proof of Lemma 1.5 show, that this mapping is a SIG-homomorphism. Since every SIG-homomorphism must satisfy the conditions i) and ii) it follows by induction, that φ is unique. □

In the following we abbreviate ($\mathbf{WST}$,SIG) to $\mathbf{WST}$ and ($\mathbf{WST}_{gr}$,SIG) to $\mathbf{WST}_{gr}$ if no confusion arises.

1.7 Definition. A mapping $\sigma: \mathbf{WST} \to \mathbf{WST}$ is called a SIG-substitution, iff it is a SIG-endomorphism on $\mathbf{WST}$, which is identical almost everywhere. □

Let $\mathbf{\Sigma}$ denote the set of all SIG-substitutions.

1.8 Lemma. σ is a SIG-substitution, iff the following conditions hold:

i) $\sigma c = c$ for all $c \in \mathbf{C}$.

ii) $\sigma f(t_1,...,t_n)) = f(\sigma t_1,...,\sigma t_n)$ for all $t = f(t_1,...,t_n) \in \mathbf{WST}$.

iii) $[\sigma t] \leq [t]$ for all $t \in \mathbf{WST}$.

iv) $\mathrm{DOM}(\sigma) = \{x \in \mathbf{V} \mid \sigma x \neq x\}$ is finite.

Proof. "⇒": Let $\sigma \in \mathbf{\Sigma}$. Definition 1.3 ii) implies i) and ii) of 1.8. For a SIG-endomorphism σ on **WST** we have $\sigma(S^{\mathbf{WST}}) \subseteq S^{\mathbf{WST}}$ for every $S \in \mathbb{S}$. Hence 1.8 iii) holds. Condition iv) follows from the fact that σ is identical almost everywhere.

"⇐": Let conditions i) - iv) be satisfied. We have to show that σ is a SIG-endomorphism. 1.8 iii) implies $\sigma(S^{\mathbf{WST}}) \subseteq S^{\mathbf{WST}}$ for every $S \in \mathbb{S}$., hence 1.3 i) is satisfied.

For t_i with $[t_i] \leq S_{f,i}$, where $(S_{f,1},...,S_{f,n+1})$ is the greatest element of SO(f), we have $f(t_1,...,t_n) \in \mathbf{WST}$. Now 1.8 i) + ii) implies 1.3 ii). □

The following proposition takes us back to the standard definition of $\mathbf{\Sigma}$ (see e.g. [Wa83, He83]) localizing the test for σ to be a SIG-substitution.

1.9 Proposition. Let $\sigma: \mathbf{WST} \rightarrow \mathbf{WST}$ be a mapping satisfying conditions 1.8 i), ii) and iv) and $[\sigma x] \leq [x]$ for all $x \in \mathbf{V}$.

Then $\sigma \in \mathbf{\Sigma}$.

Proof. By structural induction. □

Atoms are expressions $P(t_1,...,t_n)$, where P is a predicate symbol and the t_i's are well-sorted terms such that $[t_i] \leq S_i$ for $i=1,...,n$ if $SO(P) = (S_1,...,S_n)$. A literal is an expression +A or -A, where A is an atom. We denote the corresponding sets with **L** and **A**. A clause is a set of literals, which stands for the disjunction of it's literals with variables quantified over their domain (i.e. over the domain, which corresponds to the sort of this variable).

A ground atom, literal or clause is one without variables. We denote this with the subscript "gr". Instances of atoms, literals and clauses are the images under a SIG-substitution. where the appliction is defined by $\sigma(\pm P(t_1,...,t_n)) = \pm P(\sigma t_1,...,\sigma t_n)$.

Equality of our object language "$\equiv$" is a distinguished binary predicate with domainsorts $SO(\equiv) = (\top,\top)$. A model that interprets $\equiv$ as the intended equality relation is called an E-model [Lo78].

We define an E-model for a many-sorted logic as in [GM84], i.e.:

1.10 Definition. Let SIG=($\mathbb{S}$, $\mathbf{F}$, $\mathbf{P}$) be a signature. Let CS be a SIG-sorted clause set and let D be a set (the carrier set). The triple (D,SIG,R) is called an E-model of CS, iff the following conditions hold:

i) (D,SIG) is an algebra of type SIG.

ii) R is a set of relations over D. For every $P \in \mathbf{P}$, there is exactly one relation P^D of the same arity.

iii) For every clause $C \in CS$ and every SIG-homomorphism $\varphi: \mathbf{WST} \to D$, there exists a literal $\text{sgn}\ P(t_1,\dots,t_n) \in CS$, such that

$\text{sgn} = +$ and $(\varphi t_1,\dots,\varphi t_n) \in P^D$ or

$\text{sgn} = -$ and $(\varphi t_1,\dots,\varphi t_n) \notin P^D$.

I.e. for every assignment of values in D to variables in C, the resulting disjunction of literals is true if interpreted in (D,SIG,R).

iv) $\equiv$ is interpreted as the identity relation on D.

I.e. $t_1 \equiv t_2$ is true interpreted by a SIG-homomorphism φ, iff $\varphi(t_1) = \varphi(t_2)$ (or equivalently $\varphi(t_1) \equiv^D \varphi(t_2)$ is valid). □

<u>1.11 Example.</u> We give an example for E-models:

$\mathbf{S} = \{\top,\text{NAT},\text{EVEN},\text{ODD}\}$ where $\top \geq \text{NAT} \geq \text{EVEN},\text{ODD}$.

$\mathbf{C} = \{c_O, c_E\}$ with $[c_O] = \text{ODD}$ and $[c_E] = \text{EVEN}$.

$\mathbf{F} = \{*\}$, where $SO(*) = \{$ (NAT,NAT,NAT), (EVEN,NAT,EVEN), (ODD,NAT,NAT),
(NAT,EVEN,EVEN), (NAT,ODD,NAT), (EVEN,EVEN,EVEN),
(EVEN,ODD,EVEN), (ODD,EVEN,EVEN), (ODD,ODD,ODD) $\}$.

$\mathbf{P} = \{\equiv\}$

$CS = \{ ((x*y)*z) \equiv (x*(y*z)) \}$, where $x,y,z \in \mathbf{V}_{NAT}$.

To construct an E-model (D,SIG,R) of CS, we make the following defintions:

- D := NAT, the natural numbers.
- $\equiv^D$ is the identity relation on NAT.
- $c_O{}^D = 1$, $c_E{}^D = 2$,
- $\text{EVEN}^D = \{n \mid n \in \text{NAT}$ and n is even $\}$
- $\text{ODD}^D = \{n \mid n \in \text{NAT}$ and n is odd $\}$
- $*^D$ is the ordinary product on NAT.

Now for every SIG-homomorphism $\varphi: \mathbf{WST} \to D$ we have:

$\varphi(x*(y*z)) = \varphi x\ \varphi y\ \varphi z$ and

$\varphi((x*y)*z) = \varphi x\ \varphi y\ \varphi z$

Obviously (D,SIG,R) is an E.model for CS.

It is easy to see, that the following model constitutes an E-model of CS as well:

Let $D := \{0\}$; $*^D(0,0) = 0$; $c_O{}^D = c_E{}^D = 0$; $\text{EVEN}^D = \text{ODD}^D = D$.

It is a well-known fact, that ground terms are sufficient for building an E-model (the Herbrand or Skolem-model), i.e. we can remove such elements from D, which are not images of ground terms.
Furthermore, if the equality sign is not in $\mathbf{P}$, we can choose $\mathbf{WST}_{gr}$ as the carrier of an E-model (Herbrand sets).
Therefore we define an equivalent notion of an E-model for a many-sorted calculus, namely the HE-model (Herbrand E-model [LO78,CL71]). The difference to an E-model is purely technical. In later proofs we use both definitions. The HE-model has the advantage, that we can work with a fixed set of ground literals, which we cannot do in an arbitrary E-model, whereas the E-model has the advantage, that the substutivity for equality holds without restriction, which is not the case in (many-sorted) HE-models.

We use position vectors (occurrences in [HO80]) to select or substitute subterms of a given term or literal.
$t_{|pos}$ denotes the subterm at position pos.
$t[pos \leftarrow s]$ denotes the term constructed from t by replacing the subterm at position pos with s.

1.12 Definition. Let SIG be a signature and let CS be a SIG-sorted clause set.
M is a HE-model for CS, iff the following conditions hold:
i) M is a maximal set of well-sorted ground literals, not containing complementary literals.
ii) For every $t \in \mathbf{WST}_{gr}$, $t \equiv t \in M$.
iii) If $s \equiv t \in M$, $L \in M$, $L_{|pos} = s$ for an appropriate position pos and $L[pos \leftarrow t]$ is a well-sorted literal, then $L[pos \leftarrow t] \in M$.
iv) Every ground instance of every clause C in CS contains a literal, which is in M.

A clause set is said to be satisfiable, if an E-model exists for CS, respectively unsatisfiable, if no E-model for CS exists. The following theorem shows, that E-(un)satisfiable means exactly the same as HE-(un)satisfiable, since E-models and HE-models are equivalent.

1.13 Theorem. Let SIG = $(\mathbf{S}, \mathbf{F}, \mathbf{P})$ be a signature and let CS be a SIG-sorted clause set. Then:
CS has an HE-model $\Longleftrightarrow$ CS has an E-model.
Proof.
"$\Rightarrow$": Let M be an HE-model of CS. Then the relation ~, where $s \sim t$, iff $s \equiv t \in M$, is a congruence relation on $\mathbf{WST}_{gr}$:
- from the defintion of an HE-model it follows, that ~ is an equivalence relation.
- Let $s_i \sim t_i$, $i=1,\dots,n$, for ground terms s_i, t_i and let $f(s_1,\dots,s_n)$ and $f(t_1,\dots,t_n)$ be well-sorted.We have to show, that $f(s_1,\dots,s_n) \sim f(t_1,\dots,t_n)$.

As SO(f) has a greatest element, repeated application of the substitution rule (substituting s_i for t_i) on $f(t_1,\dots,t_n) \equiv f(t_1,\dots,t_n)$ yields well-sorted literals and finally $f(t_1,\dots,t_n) \equiv f(s_1,\dots,s_n) \in M$. Hence $f(t_1,\dots,t_n) \sim f(s_1,\dots,s_n)$. The same arguments show, that $\pm P(s_1,\dots,s_n)$ is in M , if $\pm P(t_1,\dots,t_n)$ is in M.

Now we construct an E-model for CS:

Let $D := \mathbf{WST}_{gr} / \sim$ and let the representations of constants, sorts and functions be defined canonically. (i.e. $f^D(t_1/\sim,\dots, t_n/\sim) = f(t_1,\dots,t_n)/\sim$ and $S^D = \{ t/\sim \mid t \in \mathbf{WST}_{gr}$ and $[t] \le S \}$.)

For $t_1,\dots,t_n \in \mathbf{WST}_{gr}$, let $P^D(t_1/\sim,\dots,t_n/\sim)$ be valid, iff $P(t_1,\dots,t_n) \in M$.

This is well defined, since $\sim$ is a congruence relation.

The algebra D of type SIG and the defined relation constitute an E-model:

We show, that for every SIG-homomorphism φ, every clause is valid.

Let φ: $\mathbf{WST} \to D$ be a SIG-homomorphism and let C be a clause in CS.

Let $\mathbf{V}(C) = \{\mathbf{x}_1,\dots,\mathbf{x}_m\}$. There exist ground terms t_i , $i=1,\dots,n$ such that $\varphi\mathbf{x}_i = t_i/\sim$ and $[t_i] \le [\mathbf{x}_i]$. Consider the ground SIG-substitution $\sigma = \{\mathbf{x}_i \leftarrow t_i \mid i=1,\dots,n \}$. There exists a literal L of σC, which is in M, since M is a HE-model. We have $\varphi C = \sigma C/\sim$, hence $L/\sim$ is valid. Thus φC is valid.■

"⇐":

Let (D,SIG,R) be an E-model of CS. We can assume, that $\varphi(\mathbf{WST}_{gr}) = D$ for every SIG-homomorphism φ. We construct a HE-model M in the following way: Let φ_0: $\mathbf{WST}_{gr} \to D$ be the unique SIG-homomorphism. M is defined as the set of all well-sorted ground literals, which are true with respect to φ_0.

We show all properties for an HE-model (see 1.12):

i) and ii) are trivial.

iii) Let $s \equiv t \in M$, $L \in M$, $L_{|pos} = s$ and let $L[pos \leftarrow t]$ be a well-sorted literal.
We have $\varphi_0(s) = \varphi_0(t)$. Hence $\varphi_0(L) = \varphi_0(L[pos \leftarrow t])$, since φ_0 is a SIG-homomorphism. Now $L \in M$ implies $L[pos \leftarrow t] \in M$.

iv) Let $C \in CS$, and let σ be a ground SIG-substitution.
$\varphi_0 \circ \sigma$ is a SIG-homomorphism, hence $\varphi_0 \circ \sigma$ (C) is valid in (D,SIG,R). This means, that there exists a literal L in C, such that $\varphi_0 \circ \sigma$ (L) is valid.
By definition of M, $\sigma(L)$ is in M, hence iv) is satisfied. □

2. Unification in ΣRP*.

In this chapter it is shown, that the unification of terms containing polymorphic functions is of type finitary [Si84], that is, for any pair of given terms in ΣRP*, there exists a minimal and finite set of most general unifiers (mgu). We give an example to demonstrate that the set of mgu's can become exponentially large. An algorithm **SUNIFY** is presented, which computes a complete and finite set of mgu's for a given pair of polymorphic terms.

2.1 Basic Notions for Unification of Polymorphic Terms.

We use the following notation:

DOM (σ)	$= \{x \in \mathbf{V} \mid \sigma x \neq x\}$.
COD (σ)	$= \{\sigma x \mid x \in$ DOM $(\sigma)\}$.
$\mathbf{V}$ $(O_1,\dots,O_n)$	= the set of variables occuring in the objects $O_1,\dots,O_n$
VCOD (σ)	$= \mathbf{V}$ (COD (σ)).
ε	denotes the identical substitution.
$\mathbf{\Sigma}^*$	$= \{\sigma \in \mathbf{\Sigma} \mid \sigma \circ \sigma = \sigma\}$, i.e. the set of idempotent SIG-subsititutions.
$\langle s = t \rangle$	denotes the problem to unify the terms s and t.

The definitions and lemmata, which we need for polymorphic terms, are adapted from [He83, Si84, Fa83, Hu76], which treat unification of unsorted terms. For unification of sorted terms see also [Wa84].

The next two lemmas are taken from [He83] and can easiliy be generalized to polymorphic terms.

2.1.1 Lemma. Let $\sigma \in \mathbf{\Sigma}$. Then:

$$\sigma \in \mathbf{\Sigma}^* \iff \mathrm{DOM}(\sigma) \cap \mathrm{VCOD}(\sigma) = \emptyset$$

2.1.2 Lemma. Let $\sigma,\tau \in \mathbf{\Sigma}^*$. Then:

$$\mathrm{DOM}(\tau) \cap \mathrm{VCOD}(\sigma) = \emptyset \Rightarrow \sigma \circ \tau \in \mathbf{\Sigma}^*.$$

2.1.3 Definition. A SIG-substitution $\varrho \in \mathbf{\Sigma}^*$ is a renaming substitution, iff the following conditions hold:

i) $\mathrm{COD}(\varrho) \subseteq \mathbf{V}$.

ii) $\forall x,y \in \mathbf{V}$: $x,y \in \mathrm{DOM}(\varrho) \wedge x \neq y \Rightarrow \varrho x \neq \varrho y$,
i.e. ϱ is injective on $\mathrm{DOM}(\varrho)$.

iii) $\forall x \in \mathbf{V}$: $[\varrho x] = [x]$. i.e. ϱ is type conform.

2.1.4 Definition. Let $s,t \in \mathbf{WST}$.

i) $s \leq t$, iff there exists a $\lambda \in \mathbf{\Sigma}$, such that $s = \lambda t$.

ii) $s \approx t$, iff $s \leq t$ and $t \leq s$.

Note that $\leq$ is a reflexive and transitive relation on **WST** and that $\approx$ is an equivalence relation.

We generalize these relations to SIG-substitutions:

2.1.5 Definition. Let $W \subseteq \mathbf{V}$ and let $\sigma,\tau \in \mathbf{\Sigma}$.

i) $\sigma = \tau \ [W]$, iff $\sigma x = \tau x$ for all $x \in W$.

ii) $\sigma \leq \tau \ [W]$, iff there exists a $\lambda \in \mathbf{\Sigma}$, such that $\sigma = \lambda \circ \tau \ [W]$.

iii) $\sigma \approx \tau \ [W]$, iff $\sigma \leq \tau \ [W]$ and $\tau \leq \sigma \ [W]$.

Obviously $\leq [W]$ is a reflexive and transitive relation on SIG-substitutions and $\approx [W]$ is an equivalence relation.

2.1.6 Definition. A SIG-substitution ξ is a permutation, iff
there exists a SIG-substitution ξ^-, such that $\xi \circ \xi^- = \varepsilon$.

In the following lemma we state without proof some facts about permutations:

2.1.7 Lemma. Let ξ be a permutation. Then:

i) ξ^- is a permutation.

ii) $(\xi^-)^- = \xi$

iii) $\mathrm{DOM}(\xi) = \mathrm{VCOD}(\xi)$. □

Every renaming substitution ϱ corresponds to a permutation $\hat{\varrho}$, where $\hat{\varrho}$ is defined as follows:

$$\hat{\varrho}(x) = \begin{cases} \varrho(x) & \text{if } x \in \mathrm{DOM}(\varrho) \\ y & \text{if } x \in \mathrm{VCOD}(\varrho) \text{ and } y \in \mathrm{DOM}(\varrho) \text{ is the unique variable with } \varrho y = x \\ x & \text{else} \end{cases}$$

$\hat{\varrho}$ is well defined since ϱ is idempotent.
Obviously $\hat{\varrho}\,\hat{\varrho} = \varepsilon$ and $\varrho = \hat{\varrho}\ [\mathrm{DOM}(\varrho)]$.
For example if $\varrho = \{u \leftarrow v\}$, then $\hat{\varrho} = \{u \leftarrow v, v \leftarrow u\}$.

2.1.8 Lemma. Let $s,t \in$ **WST** and let $W \subseteq \mathbf{V}$.
Then $s \approx t$, iff there exists a permutation ξ such that $s = \xi t$.
Proof. "$\Leftarrow$": is trivial.
"$\Rightarrow$": The proof in [Hu76] can be generalized to polymorphic terms.

2.1.9 Lemma. Let $\sigma,\tau \in \mathbf{\Sigma}$ and let $W \subseteq \mathbf{V}$.
Then $\sigma \approx \tau \ [W]$, iff there exists a permutation ξ such that $\sigma = \xi \circ \tau \ [W]$.
Proof. "$\Leftarrow$": is trivial.
"$\Rightarrow$": Follows from 2.1.8 if we take the terms $s = h(\sigma x_1,\ldots,\sigma x_n)$ and
$t = h(\tau x_1,\ldots,\tau x_n)$, where $W = \{x_1,\ldots,x_n\}$. □

2.1.10 Lemma. Let $W \subseteq \mathbf{V}$ and let $\sigma,\tau \in \mathbf{\Sigma}$.
For a renaming substitution ϱ, the following holds:

i) If $\mathrm{DOM}(\varrho) = \mathbf{V}(\tau(W))$, then:

$$\sigma \leq \tau \ [W] \iff \sigma \leq \varrho\circ\tau \ [W].$$

ii) If $\mathrm{DOM}(\varrho) = \mathbf{V}(\sigma(W))$, then:

$$\sigma \leq \tau \ [W] \iff \varrho\circ\sigma \leq \tau \ [W].$$

iii) If $\mathrm{DOM}(\varrho) = \mathbf{V}(\sigma(W))$, then:

$$\sigma \approx \tau \ [W] \iff \varrho\circ\sigma \approx \tau \ [W].$$

Proof.

i) "⇒": Let $\sigma \leq \tau \ [W]$. Then $\sigma = \lambda\circ\tau \ [W]$ for some $\lambda \in \mathbf{\Sigma}$. We have

$$\begin{aligned} \sigma &= \lambda\circ\hat{\varrho}\circ\hat{\varrho}\circ\tau \ [W] && (\hat{\varrho}\circ\hat{\varrho} = \varepsilon) \\ &= \lambda\circ\hat{\varrho}\circ\varrho\circ\tau \ [W] && (\mathrm{DOM}(\varrho) = \mathbf{V}(\tau(W)) \text{ and } \varrho = \hat{\varrho}\ [\mathrm{DOM}(\varrho)]). \end{aligned}$$

Hence $\sigma \leq \varrho\circ\tau \ [W]$.

"⇐": trivial

ii) "⇒": trivial.

"⇐": From $\varrho\circ\sigma = \lambda\circ\tau \ [W]$ for some $\lambda \in \mathbf{\Sigma}$ we have:

$\sigma = \hat{\varrho}\circ\hat{\varrho}\circ\sigma = \hat{\varrho}\circ\varrho\circ\sigma \ [W]$, since $\mathrm{DOM}(\varrho) = \mathbf{V}(\sigma(W))$ and $\varrho = \hat{\varrho}\ [\mathrm{DOM}(\varrho)]$.

Now $\sigma = \hat{\varrho}\circ\varrho\circ\sigma = \hat{\varrho}\circ\lambda\circ\tau \ [W]$ implies $\sigma \leq \tau \ [W]$.

iii) Combination of i) and ii). □

2.1.11 Definition. Let $s,t \in \mathbf{WST}$ and let $\sigma \in \mathbf{\Sigma}^*$.
Then σ is called a SIG-unifier of s and t iff $\sigma s = \sigma t$.

In general there will be more than one unifier for a given pair of terms s and t, hence we have to consider sets of unifiers. In order to simplify proofs, it is often convenient, that the following technical conditions are satisfied for such sets. This can always be achieved by applying appropriate renaming substitutions.

2.1.12 Definition. Let $U \subseteq \mathbf{\Sigma}$ be a set of SIG-substitutions and let $W \subseteq \mathbf{V}$.
We say U is separated on W iff the following conditions hold:

i) $\forall \sigma \in U$: $\quad \mathrm{DOM}(\sigma) = W$.

ii) $\forall \sigma \in U$: $\quad \mathrm{VCOD}(\sigma) \cap W = \emptyset$

iii) $\forall \sigma,\tau \in U$: $\quad \sigma \neq \tau \Rightarrow \mathrm{VCOD}(\sigma) \cap \mathrm{VCOD}(\tau) = \emptyset$ □

Note that the conditions of 2.1.12 imply, that $U \subseteq \mathbf{\Sigma}^*$.
A set of unifiers generated by an actual unification algorithm is not separated in general. (see e.g [Ro65, He83])

Since two polymorphic terms may have more than one most general unifier, we define a set of most general unifiers as the minimal set of solutions of $\langle s=t\rangle$ in $(\mathbf{WST}, \mathrm{SIG})$:

2.1.13 Definition. Let $s,t \in \mathbf{WST}$ and let $W \subseteq \mathbf{V}$ with $\mathbf{V}(s,t) \subseteq W$.
A complete set of most general unifiers for s and t, $CU\Sigma(s,t)$ is defined as a subset of $\mathbf{\Sigma}^*$, which is separated on W and satisfies the following conditions:

i) $\forall \sigma \in CU\Sigma(s,t)$: $\sigma s = \sigma t$ (correctness)

ii) $\forall \delta \in \mathbf{\Sigma}$: $\delta s = \delta t \Rightarrow (\exists \sigma \in CU\Sigma(s,t)$: $\delta \leq \sigma$ **[W]**) (completeness).

The set of most general unifiers is called minimal, if in addition:

iii) $\forall \sigma,\tau \in CU\Sigma(s,t)$: $\sigma \leq \tau$ **[W]** $\Rightarrow \sigma = \tau$ (minimality)

A minimal set of most general unifiers for s and t is denoted as $\mu U\Sigma(s,t)$. □

There may exist more than one set $\mu U\Sigma(s,t)$ for a given pair $s,t \in \mathbf{WST}$, but if so, they are equivalent under permutations [Fa83, Hu76].

2.1.14 Lemma. Let $W \subseteq \mathbf{V}$.
For every finite $U \subseteq \mathbf{\Sigma}$ there exists an $U' \subseteq \mathbf{\Sigma}^*$ and a mapping $\varphi: U \to U'$ such that the following conditions hold:

i) $\varphi: U \to U'$ is a bijection.

ii) $\varphi(\sigma) \approx \sigma$ **[W]**

iii) U' is separated on W.

Proof. For every $\sigma \in U$ we take a renaming ϱ_σ with $DOM(\varrho_\sigma) = \mathbf{V}(\sigma(W))$. It is possible to choose ϱ_σ in such a way, that $VCOD(\varrho_\sigma) \cap VCOD(\varrho_\tau) = \emptyset$ for different $\sigma,\tau \in U$. Now we define $\varphi: U \to U'$ as $\varphi(\sigma) - \varrho_\sigma \circ \sigma$ and U' is defined as $\varphi(U)$. Lemma 2.1.10 implies $\varrho_\sigma \circ \sigma \approx \sigma$ **[W]** . U' is separated by construction. □

We define a weakening substitution (coercing) [Wa84,CD83, GM85] essentially like a renaming substitution, except that it maps variables to variables of lesser sorts. During the computation of most general unifiers, weakening substitutions are used to solve the unification problem $\langle x=t \rangle$, where $[x]$ is not a subsort of $[t]$

.

2.1.15. Definition. Let $\sigma \in \mathbf{\Sigma}^*$. We say σ is a weakening substitution, iff

i) $COD(\sigma) \subseteq \mathbf{V}$.

ii) $\forall x,y \in DOM(\sigma)$: $x \neq y \Rightarrow \sigma x \neq \sigma y$, i.e. σ is injective on $DOM(\sigma)$.

iii) $\forall x \in \mathbf{V}$: $[\sigma x] \leq [x]$.

The set of all weakening substitutions is called $\mathbf{W\Sigma}$.

2.1.16 Lemma. Let $W \subseteq \mathbf{V}$ and let $\sigma,\tau \in \mathbf{W\Sigma}$.
If $W \subseteq DOM(\tau)$ then :

$\sigma \leq \tau$ **[W]** $\iff \forall x \in W$: $[\sigma x] \leq [\tau x]$.

Proof. "$\Rightarrow$": There exists a $\lambda \in \mathbf{\Sigma}$, such that $\sigma = \lambda \circ \tau$ **[W]**. Hence we have

$[\sigma x] = [\lambda \circ \tau x] = [\lambda \circ (\tau x)] \leq [\tau x]$.

"$\Leftarrow$": We define a $\lambda \in \mathbf{\Sigma}$, such that $\sigma = \lambda \circ \tau$ $[W]$:

$\lambda y := \sigma x$, iff $y = \tau x$ for some $x \in W$. τx and σx are variables and all τx are different for $x \in W$, since $\tau(W) \cap W = \emptyset$ and τ is injective on W. From $[\sigma x] \leq [\tau x]$ and from the fact that λ moves at most finitely many variables, we conclude that $\lambda \in \mathbf{\Sigma}$. □

If U is a set of unifiers, we are in general only interested in a minimal subset of U, i e. a subset, which contains one representative for every maximal element of U. Thus we define:

2.1.17 Definition. Let U be a set ordered by a reflexive and transitive relation $\leq$. Then $MAX_{\leq}(U)$ is a set which satisfies the following conditions:

i) $MAX_{\leq}(U) \subseteq U$.

ii) $\forall u \in U: \exists v \in MAX_{\leq}(U): u \leq v$.

iii) $\forall u,v \in MAX_{\leq}(U): u \leq v \Rightarrow u = v$. □

Such a set exists if either U is finite or every chain $\sigma_1 < \sigma_2 < \ldots$ is finite , since the relation is reflexive and transitive, but is not unique in general. The cardinality of $MAX_{\leq}(U)$ is uniquely determined, since it equals the number of equivalence classes of maximal elements of U.
For example we have $\mu U\Sigma(s,t) = MAX_{\leq[W]}(CU\Sigma(s,t))$ □

The set of most general weakening substitutions for a given term is defined in a similar fashion to the definition of $\mu U\Sigma$. Note that t may be a term that is ill-sorted.

2.1.18 Definition. Let $t \in \mathbf{T}(\mathbf{F},\mathbf{V})$, $\mathbf{V}(t) \subseteq W \subseteq \mathbf{V}$ and let $S \in \mathbf{S}$.
The set of most general weakening substitutions for t and S, $\mu W\Sigma_S(t)$, is a set of weakening substitutions, which is separated on W and satisfies the following conditions:

i) $\forall \sigma \in \mu W\Sigma_S(t)$: $[\sigma t] \leq S$ (correctness)

ii) $\forall \delta \in \mathbf{\Sigma}$: $[\delta t] \leq S \Rightarrow \exists \sigma \in \mu W\Sigma_S(t): \delta \leq \sigma$ $[W]$. (completeness)

iii) $\forall \sigma,\tau \in \mu W\Sigma_S(t)$: $\sigma \leq \tau$ $[W] \Rightarrow \sigma = \tau$. (minimality)

This definition is of course useful only for terms, which are not well-sorted or for terms, whose outermost function symbol is polymorphic, since otherwise $[\sigma t] = [t]$ for any σ. If the signature contains only one sort, then $\mu W\Sigma_S(t) = \{\varrho\}$, where ϱ is a renaming of W. The same holds if $S \geq [t]$.

2.1.19 Theorem. Let $t \in \mathbf{WST}$ and let $S \in \mathbf{S}$.
If there exists a $\theta \in \mathbf{\Sigma}$, such that $[\theta t] \le S$, then $\mu W\Sigma_S(t)$ exists, it is not empty and it is always finite.

Proof. Let $U_0 := \{\sigma \mid \sigma \in \mathbf{\Sigma}^* \text{ and } [\sigma t] \le S\}$ and let $W := \mathbf{V}(t)$. For every (admissable) combination of sort assignments to variables in W choose one $\sigma \in U_0$, which makes this assignment. Let U_1 be the set of all σ thus chosen. U_1 contains a σ_0 which makes the same assignments as θ. The number of different combinations of sort assignments to variables in W is finite. Hence U_1 is not empty and finite. Lemma 2.1.14 gives a set U_2 which is separated on W. Now the set $MAX_{\le[\![W]\!]}(U_2)$ satisfies all properties of $\mu W\Sigma_S(t)$, hence we define $\mu W\Sigma_S(t) := MAX_{\le[\![W]\!]}(U_2)$. □

2.1.20 Corrollary. Let $x \in \mathbf{V}$, $t \in \mathbf{WST}$, such that x and t are $\mathbf{S}$-unifiable.
Then $\mu U\Sigma(x,t)$ exists, is not empty and finite.

Proof. Let $S = [x]$ and let $W = \{x\} \cup \mathbf{V}(t)$. Then the set $\mu W\Sigma_S(t)$ is not empty and finite by theorem 2.1.19. We have
$\mu U\Sigma(x,t) = \{\, \{x \leftarrow \sigma t\} \circ (\sigma|_{\mathbf{V}(t)}) \mid \sigma \in \mu W\Sigma_S(t)\, \}$. □

The following example demonstrates, that for a unification problem $\langle s=t \rangle$, the minimal set of most general unifiers can grow exponentially.

2.1.21 Example. Consider the sort structure $\mathbf{S} = \{N, NZ, Z\}$, where N, NZ and Z have the same meaning as in the example of chapter 1. Let $x \in \mathbf{V}_{NZ}$ and $x_i \in \mathbf{V}_N$. The signature of the function "*" (product) is:
$SO(*) = \{(N,N,N), (NZ,N,N), (N,NZ,N), (Z,N,Z), (N,Z,Z), (Z,NZ,Z), (NZ,Z,Z), (Z,Z,Z), (NZ,NZ,NZ)\}$. The unification problem $\langle x = (x_1+x_2) * \ldots * (x_{2n-1}+x_{2n}) \rangle$ produces 2^n unifiers, since for every factor there are two independent solutions $\{x_{2i-1} \leftarrow y_{2i-1}\}$ and $\{x_{2i} \leftarrow y_{2i}\}$, where $[y_j] = NZ$. These solutions have to be combined independently. □

2.2 A Unification Algorithm for Polymorphic Terms.

2.2.1 Definition. Let $s,t \in \mathbf{WST}$ with $s \neq t$. We compare s and t symbolwise from left to right. The pair of well formed subterms, starting with the first symbols, which disagree, is called the first disagreement pair of s and t [Ro65].

The following unification algorithm **S**UNIFY for polymorphic terms takes two terms as input and returns a finite set of mgu's, if a unifier exists (empty otherwise). A similar algorithm for terms in a many-sorted signature without polymorphic functions is given in [Wa83].

The polymorphic unification algorithm is defined as:

2.2.2 Definition.: **S**UNIFY : $\mathbf{WST} \times \mathbf{WST} \to \mathrm{POW}(\mathbf{\Sigma}^*)$

```
INPUT: $s,t \in \mathbf{WST}$.
  $U_{OLD} := \{\varrho\}$,   where $\varrho$ is a renaming of $\mathbf{V}(s,t)$, such that $\{\varrho\}$ is separated
                      on $\mathbf{V}(s,t)$.
  WHILE ( $\sigma s \neq \sigma t$ for some $\sigma \in U_{OLD}$ ) DO:
      $U_{NEW} := \emptyset$.
      FOR ALL $\sigma \in U_{OLD}$ DO:
          IF $\sigma s = \sigma t$ THEN $U_{NEW} := U_{NEW} \cup \{\sigma\}$.
          ELSE DO Let $(d,e)$ be the first disagreement pair for $(\sigma s, \sigma t)$
                  IF d or e is a variable THEN
                     $U_{NEW} := U_{NEW} \cup \{\tau \circ \sigma \mid \tau \in \mu U\Sigma(d,e)\}$
                  ELSE $U_{NEW} := U_{NEW}$.  ( $\sigma s$ and $\sigma t$ are not unifiable )
               END.
      END (FOR ALL)
      Make $U_{NEW}$ to be separated on $\mathbf{W} - \mathbf{V}(s,t)$.
      $U_{OLD} := U_{NEW}$.
  END (WHILE)
  RETURN $\mathrm{MAX}_{\leq[\mathbf{W}]}(U_{OLD})$. □
```

This algorithm exploits the existence and the properties of $\mu U\Sigma(x,t)$. Lemma 2.1.14 justifies the separation of U_{NEW} on $\mathbf{W}$; i.e. to restrict the substitutions σ in U_{NEW} to $\mathbf{V}(s,t)$ and afterwards to rename all variables in $\sigma(\mathbf{W})$.

2.2.3 Lemma. The algorithm **S**UNIFY terminates for every pair (s,t) of well-sorted terms.

Proof. Every $\sigma \in U_{NEW}$, which satisfies $\sigma s = \sigma t$, is transmitted (unchanged) to the next U_{NEW}. Hence it suffices to consider substitutions $\sigma \in U_{NEW}$ with $\sigma s \neq \sigma t$.

In the FOR ALL -loop:

If neither d nor e is a variable, then σ is removed from the set U_{NEW}.

Otherwise the number of variables in $\mathbf{V}(\tau \circ \sigma s, \tau \circ \sigma t)$ is exactly $|\mathbf{V}(\sigma s, \sigma t)| - 1$. Hence for every $\sigma \in U_{NEW}$ the number of successive steps until either an inherited substitution is removed or a unifier for s,t is bounded by $|\mathbf{V}(\sigma s, \sigma t)|$. Thus the algorithm halts. □

2.2.4 Theorem. Let s,t be SIG-unifiable.

Then the algorithm $\mathbb{S}$UNIFY returns a set of mgu's: $\mu U\Sigma(s,t)$.

Proof. The returned set is separated on $W - VAR(s,t)$.

i) correctness: follows from the WHILE -condition.

iii) minimality: follows from the fact, that the returned set is minimized (see 2.1.17).

ii) completeness: Let $\theta \in \mathbf{\Sigma}$, such that $\theta s - \theta t$

We show by induction, that the following statement remains true during the WHILE-loop:

WPR.: $\exists \sigma \in U_{OLD} : \theta \le \sigma\ [W]$.

Base case. $\theta \le \varrho\ [W]$ holds before the WHILE-loop due to Lemma 2.1.10.

Induction step. Assume that WPR is true before the WHILE-loop.

Case 1: If $\sigma s = \sigma t$ then $\sigma \in U_{NEW}$.

Case 2: If $\sigma s \neq \sigma t$, then let (d,e) be the first disagreement pair of $(\sigma s, \sigma t)$.

There exists a $\lambda \in \mathbf{\Sigma}$, such that $\theta = \lambda \circ \sigma\ [W]$ and $DOM(\lambda) \subseteq \mathbf{V}(\sigma W)$.

λ unifies σs and σt, hence $\lambda d = \lambda e$. Either d or e is a variable, since the first symbols disagree.

By Corollary 2.1.20 there exists a $\tau \in \mu U\Sigma(d,e)$ such that $\lambda \le \tau\ [\mathbf{V}(d,e)]$.

We have $DOM(\lambda) \cap \tau(\mathbf{V}(d,e)) - \emptyset$, since $VCOD(\tau)$ consists of new variables.

There exists a $\mu \in \mathbf{\Sigma}$ with $DOM(\mu) \subseteq VCOD(\tau)$ such that $\lambda = \mu \circ \tau\ [\mathbf{V}(d,e)]$.

Furthermore we have $\lambda = \mu \circ \lambda \circ \tau\ [\mathbf{V}(\sigma W)]$:

For $x \in \mathbf{V}(d,e)$:

$\mu \circ \lambda \circ \tau x =$

$\mu \circ \tau x =$ $\quad (DOM(\lambda) \cap \tau(\mathbf{V}(d,e)) = \emptyset)$

λx $\quad (\lambda = \mu \circ \tau\ [\mathbf{V}(d,e)].\ )$

For $x \notin \mathbf{V}(d,e)$:

$\mu \circ \lambda \circ \tau x = \mu \circ \lambda x$ $\quad (DOM(\tau) = (\mathbf{V}(d,e)\ ).$

Case 1: $x \in DOM(\lambda)$:

$\mu \circ \lambda x = \lambda x$ $\quad (VCOD(\lambda) \cap DOM(\mu) = \emptyset)$

Case 2: $x \notin DOM(\lambda)$:

$\mu \circ \lambda x = \mu x = x = \lambda x$ $\quad (x \notin DOM(\lambda)$; $DOM(\mu) \subseteq VCOD(\tau)$ and ($VCOD(\tau)$ consists of new variables)

Finally, $\theta = \lambda \circ \sigma = \mu \circ \lambda \circ \tau \circ \sigma\ [W]$, hence $\theta \le \tau \circ \sigma\ [W]$.

$\tau \circ \sigma$ is in U_{NEW}, hence WPR is true after the WHILE-loop.

The technical manipulation to make the set U_{NEW} being separated on W does not affect the the validity of WPR (see Lemma 2.1.14).
The transitivity of $\leq [\![W]\!]$ guarantees that there exists a $\sigma \in MAX_{\leq[\![W]\!]}(U_{OLD})$ such that $\theta \leq \sigma\ [\![W]\!]$. □

2.2.5 Example. The minimizing step of **S**UNIFY is necessary for the minimality of the returned unifier set:
Let $\mathbf{S} := \{\top, A, B\}$ with $\top \geq A \geq B$.
Let f,g,h be functions with the signatures:
$SO(f) = \{(A,A,A);\ (A,B,B);\ (B,A,B);\ (B,B,B)\ \}$
$SO(g) = \{(A,A,A);\ (A,B,A);\ (B,A,A);\ (B,B,B)\ \}$
$SO(h) = \{(A,A,A);\ (A,B,A);\ (B,A,A);\ (B,B,A)\ \}$
Let $s = h(f(x\ y)\ g(x\ y))$ and let $t = h(u\ v)$ with
$x,y \in \mathbf{V}_A$ and $u,v \in \mathbf{V}_B$.
We unifiy s and t using the algorithm **S**UNIFY:
First we get:
u and f(x y) is the first disagreement pair. There are two most general unifiers of u and f(x y).
$U_1 = \{\ \{x \leftarrow x_1, u \leftarrow f(x_1\ y)\ \};\ \{y \leftarrow y_1, u \leftarrow f(x\ \ y_1)\ \}\ \ \}$ where $x_1,y_1 \in \mathbf{V}_B$.
In the next step the first disagreement pairs for the two unifiers in U_1 are $(v, g(x_1\ \ y))$ and $(v, g(x\ \ y_1))$, respectively.
We get:
$U_2 = \{\ \ \{x \leftarrow x_1, y \leftarrow y_2, u \leftarrow f(x_1\ y_2), v \leftarrow g(x_1\ y_2)\ \};$
$\{x \leftarrow x_2, y \leftarrow y_1, u \leftarrow f(x_2\ y_1), v \leftarrow g(x_2\ y_1)\ \}\ \ \}$ where $x_2,y_2 \in \mathbf{V}_B$.
The two unifiers in U_2 are equivalent ($\approx [\![\{x,y,u,v\}]\!]$).

We say $\langle \mathbb{S}, \leq \rangle$ is a semilattice, iff for every $R,S \in \mathbb{S}$ with $R \sqcap S \neq \emptyset$ there exists a unique $T \in \mathbb{S}$ such that: $T \leq R,S$ and for every $T' \in \mathbb{S}$: $T' \leq R,S \;\Rightarrow\; T' \leq T$.
I.e. an infimum of R and S exists, provided these two sorts have a common subsort. This property implies that a supremum of two sorts exists.

We denote the unique greatest element of $R \sqcap S$ as $R \wedge S$ and the supremum of R and S (which always exists) as $R \vee S$.
Furthermore in a semilattice a set of sorts $\{S_1,\ldots,S_n\}$ has a unique infimum (or greatest lower bound, g.l.b.), if a lower bound exists and a unique supremum (least upper bound, l.u.b.),
We could extend the semilattice $\langle \mathbb{S}, \leq \rangle$ into a complete lattice by adding a least element (but we don't).

<u>2.2.6 Definition.</u> Let SIG= $(\mathbb{S}, \mathbb{F}, \mathbb{P})$ be a signature.

We say SIG is a <u>unification unique signature</u>, iff the following conditions hold:

i) $\langle \mathbb{S}, \leq \rangle$ is a semilattice

ii) For all $f \in \mathbb{F}$ and all $S \in \mathbb{S}$, the set $M(f,S) := \{(S_1,\ldots,S_{n+1}) \in SO(f) \mid S_{n+1} \leq S\}$ is either empty or contains a unique greatest element. □

<u>2.2.7 Theorem.</u> Let SIG be a unification unique signature.

Then for every $s,t \in \mathbf{WST}$: $\mu U\Sigma(s,t)$ is at most a singleton.

<u>Proof.</u> If $\mu W\Sigma_S(t)$ is at most a singleton for all $t \in \mathbf{WST}$, then the properties of $\mathbb{S}$UNIFY (Lemma 2.2.3 and theorem 2.2.4) imply that $\mu U\Sigma(s,t)$ is at most a singleton. Thus we show, that $\mu W\Sigma_S(t)$ is at most a singleton for all t and S.

We prove this by induction on the term structure of t:

<u>Base case</u>: If t is a constant, then either $\mu W\Sigma_S(t) = \{\varepsilon\}$ or $= \emptyset$.

If t is a variable, then two cases are possible:

<u>Case 1:</u> $S \sqcap [t] \neq \emptyset$: $\mu W\Sigma_S(t) = \{\{t \leftarrow z\}\}$ where z is a variable with $[z] = S \wedge [t]$

<u>Case 2</u>: $S \sqcap [t] = \emptyset$: Then $\mu W\Sigma_S(t) = \emptyset$.

<u>Induction step.</u> We show this by contradiction.

Let $t = f(t_1,\ldots,t_n)$ and let $\sigma_1, \sigma_2 \in \mu W\Sigma_S(t)$ with $\sigma_1 \neq \sigma_2$.

If $[\sigma_1 x] = [\sigma_2 x]$ for all $x \in W = \mathbb{V}(t)$, then $\sigma_1 \approx \sigma_2$ $[W]$ by Lemma 2.1.16.

Hence there exists a variable $x_0 \in W$ such that $[\sigma_1 x_0] \neq [\sigma_2 x_0]$.

We construct SIG-substitutions τ and τ' by:

$$\tau' y = \begin{cases} \sigma_1 y & \text{if } y \neq x_0 \\ z & \text{if } y = x_0 \text{ ; z is a new variable and } [z] = [\sigma_1 x_0] \vee [\sigma_2 x_0]. \end{cases}$$

$[\sigma_1 x_0] \vee [\sigma_2 x_0]$ exists and is unique, since $\langle \mathbb{S}, \leq \rangle$ is a finite semilattice.

We have $\sigma_1 \leq \tau'$ $[W]$ by Lemma 2.1.16.

Let τ be a renamed variant of τ' such that $\{\tau, \sigma_1, \sigma_2\}$ is separated on W.

Then obviously $\sigma_1 \leq \tau \llbracket W \rrbracket$.

Case 1 $[z] > [\sigma_1 x_0]$.

Then $\sigma_1 < \tau \llbracket W \rrbracket$. We have $[\tau t] \nleq S$, since σ_1 is a maximal element in the set of all substitutions, which weaken t to S.
Hence there exists an index j and a term $t_j \in \{t_1,\ldots,t_n\}$, such that $[\tau t_j] > S_j$ where $(S_1,\ldots,S_{n+1})$ is the greatest element of M(f,S).
The facts $[\sigma_1 t_j] \leq S_j$ and $[\sigma_2 t_j] \leq S_j$ imply that for the unique $\lambda_j \in \mu W\Sigma_{S_j}(t_j)$ we have $\sigma_1 \leq \lambda_j \llbracket W_j \rrbracket$ and $\sigma_2 < \lambda_j \llbracket W_j \rrbracket$, where $W_j = \mathbf{V}(t_j)$. This λ_j exists and is unique by the induction hypothesis.
The variable x_0 must be in W_j because otherwise $[\tau t_j] = [\sigma_1 t_j]$.
We have $[\lambda_j x_0] \geq [\sigma_1 x_0]$ and $[\lambda_j x_0] \geq [\sigma_2 x_0]$, hence $[\lambda_j x_0] \geq [\tau x_0]$.
We conclude that $[\lambda_j y] \geq [\tau y]$ for all $y \in W_j$. This yields the contradiction $[\tau t_j] \leq S_j$. Hence case 1 is not possible. ■

Case 2. $[z] = [\sigma_1 x_0]$. Interchanging the substitutions σ_1 and σ_2 we obtain that the case $[z] > [\sigma_1 x_0]$ is also not possible. Hence $[\sigma_1 x_0] = [z] = [\sigma_2 x_0]$.
This contradicts the properties of the variable x_0 stated above.

All cases are exhausted, so the theorem is proved. □

Since the condition 2.2.6 ii) is true in the ΣRP-calculus of [Wa83], we have the following corrollary, which is a generalization of a Theorem 7.4 in [Wa83], (see also [Wa84]) :

2.2.8 Corrollary: Let $\langle \mathbf{S},\leq \rangle$ be a semilattice. Then for every pair of terms s,t in the ΣRP-calculus: $\mu U\Sigma(s,t)$ is at most a singleton. □

The condition for $\langle \mathbf{S},\leq \rangle$ to be a semilattice is not critical. It is always possible to complete the sort structure, such that the completion is a semilattice and the (un)satisfiability of the corresponding clause set is not affected (see chapter 2.4)

2.3 Paramodulation Unifiers in ΣRP^*.

Paramodulation [WR73] with $t_1 \equiv t_2$ on a literal L is defined as follows: Let t_3 ccour at position pos in L and let σ be a most general unifier of t_1 and t_3. Then a new literal L′ ,the paramodulant, is deduced by replacing σt_3 with σt_2. We have $L' = \sigma L[pos \leftarrow t_2]$. A minimal set of most general unifiers for the paramodulation with $t_1 \equiv t_2$ on L at position pos is the set $\mu U\Sigma(t_1, L_{|pos})$. In a many-sorted calculus, the problem arises, that the paramodulant, generated by such unifiers may not be well-sorted, but an instance of this paramodulant can be well-sorted.

In [Wa83] a weakening rule is introduced to solve this problem. we propose another solution and define paramodulation by a set of unifiers, which is most general in the set of unifiers for t_1 and $L_{|pos}$ and generate a well-sorted paramodulant. This process is defined as $\mathbf{S}$-paramodulation. It includes implicitely the weakening rule of [Wa83].

2.3.1 Definition. Let L be a well-sorted literal and let $t_1 \equiv t_2$ be an equality literal with $t_1, t_2 \in \mathbf{WST}$. Let pos be a position within L and let $s = L_{|pos}$. The set of paramodulation unifiers for L, pos and $t_1 \equiv t_2$, namely $P\Sigma(t_1,t_2,L,pos)$, is a subset of $\mathbf{\Sigma}$, which satisfies the following conditions: $\forall \sigma \in P\Sigma(t_1,t_2,L,pos)$: $\sigma s = \sigma t_1$ and $\sigma L[pos \leftarrow \sigma t_2]$ is well-sorted.

Let $W = \mathbf{V}(L,t_1,t_2)$. Then $\mu P\Sigma(t_1,t_2,L,pos)$ is defined as a minimal and complete subset of $P\Sigma(t_1,t_2,L,pos)$, which is separated on W.

I.e. $\mu P\Sigma(t_1,t_2,L,pos) = MAX_{\leq[W]}(P\Sigma(t_1,t_2,L,pos))$.

The next example demonstrates, that the property of a paramodulant $\theta L[pos \leftarrow \theta t_2]$ to be well-sorted can be influenced by terms of L outside the position pos. Furthermore substitutions in $\mu P\Sigma(t_1,t_2,L,pos)$ may not be most general unifiers of t_1 and $L_{|pos}$.

2.3.2 Example. Let $\mathbf{S} = \{\top, A, B, C, D\}$ with $\top \geq A \geq B \geq C \geq D$.

Let f,g,h be functions with

$SO(f) = \{(B,B,A); (B,C,A); (C,B,A); (C,C,A); ... ; (D,D,B) \}$.

$SO(g) = \{(B,B,A); (B,C,B); (C,B,B); (C,C,B); ... \}$.

$SO(h) = \{(A,A,A); (A,B,B); (B,A,B); (B,B,B); ... \}$.

The first triple is the maximal domain and range of f,g,h.

Let $P \in \mathbf{P}$ be a predicate with $SO(P) = B$.

Let $L = P(h(g(x_B\ x_B)\ x_B))$ and let $t_1 \equiv t_2$ be the literal $y_B \equiv f(y_B\ y_B)$ where $x_B, y_B \in \mathbf{V}_B$.

We have $[\, h(g(x_B\ x_B)\ x_B)] = B$, hence L is a well-sorted literal.

We paramodulate t_1 into the second argument of h in L.

With the substitution $\sigma = \{y_B \leftarrow x_B\}$ we obtain the paramodulant $(\sigma L)[pos \leftarrow \sigma t_2] = P(h(g(x_B\ x_B)\ f(x_B\ x_B)))$, which is not well-sorted, since $[h(g(x_B\ x_B)\ f(x_B\ x_B))] = A$ and $A > B$.

If we try to weaken $f(x_B\ x_B)$ with a τ, such that $\tau f(x_B\ x_B)$ fits at position pos, we get $\tau = \{x_B \leftarrow x_D\}$. But the substitution $\tau' = \{x_B \leftarrow x_C\}$ suffices to make the paramodulant well-sorted, since $[h(g(x_C\ x_C)\ f(x_C\ x_C))] = B$.

<u>2.3.3 Theorem</u>. For all L, pos, $t_1 \equiv t_2$, which constitute a paramodulation problem in SIG, the set $\mu P\Sigma(t_1,t_2,L,pos)$ exists and is finite.

<u>Proof.</u>

Let $U_0 = \{\sigma \circ \tau \mid \tau \in \mu U\Sigma(L_{|pos},t_1)$ and $\sigma \in \mu W\Sigma_T((\tau L)[pos \leftarrow \tau t_2])\,\}$

i.e. all compositions $\sigma \circ \tau$ of substitutions σ,τ such that τ is a most general unifier of $L_{|pos}$ and t_1 , and σ is a most general weakening substitution, such that the instance of the paramodulant , namely $\sigma(\tau L[pos \leftarrow \tau t_2])$ is a well-sorted literal.

By theorems 2.1.19 and 2.2.4 the sets $\mu W\Sigma_T(\tau L[pos \leftarrow \tau t_2])$ and $\mu U\Sigma(L_{|pos},t_1)$ exist and are finite. After application of some renamings and after restricting the compositions to $W = \mathbf{V}(L,t_1,t_2)$, we can assume, that U_0 is separated on W.

We show, that the requirements of Definition 2.3.1 are satisfied for the set $U = MAX_{\le \mathbf{[}W\mathbf{]}}(U_0) = \mu P\Sigma(t_1,t_2,L,pos)$

i) <u>correctness:</u> For all $\delta = \sigma \circ \tau \in U_0$ we have $\delta(L_{|pos}) = \delta t_1$, since $\tau(L_{|pos}) = \tau t_1$. Furthermore $\delta L[pos \leftarrow \delta t_2]$ is a well-sorted literal, since $\delta L[pos \leftarrow \delta t_2] = \sigma \circ \tau L[pos \leftarrow \sigma\tau t_2] = \sigma(\tau L[pos \leftarrow \tau t_2]\,)$.

ii) <u>completeness:</u> Let $\theta \in \mathbf{\Sigma}$ such that $\theta(L_{|pos}) = \theta t_1$ and $\theta L[pos \leftarrow \theta t_2]$ is well-sorted.

By Theorem 2.2.4, there exists a $\tau \in \mu U\Sigma(L_{|pos},t_1)$ with $\theta \le \tau\ \mathbf{[}W\mathbf{]}$, hence we have $\theta = \lambda_1 \circ \tau\ \mathbf{[}W\mathbf{]}$. We have $\theta L[pos \leftarrow \theta t_2] = \lambda_1(\tau L[pos \leftarrow \tau t_2])$.

From Theorem 2.1.19 it follows, that there exists a $\sigma \in \mu W\Sigma_T(\tau L[pos \leftarrow \tau t_2])$ such that $\lambda_1 \le \sigma\ \mathbf{[}\mathbf{V}(\tau W)\mathbf{]}$.

Thus $\lambda_1 = \lambda_2 \circ \sigma\ \mathbf{[}\mathbf{V}(\tau W)\mathbf{]}$ for some $\lambda_2 \in \mathbf{\Sigma}$ and so:

$\theta = \lambda_1 \circ \tau = \lambda_2 \circ \sigma \circ \tau\ \mathbf{[}W\mathbf{]}$. The transitivity of $\le \mathbf{[}W\mathbf{]}$ now shows that there exists a $\mu \in U$, such that $\theta \le \mu\ \mathbf{[}W\mathbf{]}$.

iii) <u>minimality:</u> is trivial □

2.3.4 Corrollary. For a signature SIG, which is unification unique, the set $\mu P\Sigma(t_1,t_2,L,pos)$ is at most a singleton for all t_1,t_2,L,pos.

Proof. Follows from 2.2.7 and 2.3.3. □

In ΣRP^*, the definition of paramodulation is modified: the set $\mu U\Sigma(t_1,L|_{pos})$ is replaced by the set $\mu P\Sigma(t_1,t_2,L,pos)$. In the next chapter it is shown, that (sorted) resolution together with sorted paramodulation are the deduction rules of a complete calculus.

2.4 Embedding of $\langle \mathbb{S},\le \rangle$ into a Semilattice.

In this section it is shown that every sort structure $\langle \mathbb{S},\le \rangle$ can be embedded into a semilattice. This embedding does not affect the (un)satisfiability of a clause set, i.e. the first part of the definition of a unification unique signature (Def 2.2.6) can always be satisfied by adding some sorts to $\mathbb{S}$.

In other words the expressive power of the ΣRP*-calculus respectively the ΣRP-calculus [Wa83] is not changed, if only semilattice sort structures are admissable. For the ΣRP-calculus, this implies that the mgu-sets are always at most singletons. This is of great practical importance, since it allows to change the sort strucure before the clause set is handed over to an automated reasoning system.

2.4.1 Definition. Let $\langle \mathbb{S},\le \rangle$ and $\langle \mathbb{S}',\le' \rangle$ be sort structures with a greatest element each.

$\langle \mathbb{S},\le \rangle$ is embedded into $\langle \mathbb{S}',\le' \rangle$, iff there exists a mapping $\varphi\colon \mathbb{S} \to \mathbb{S}'$ such that the following conditions hold:

i) φ is injective.

ii) For every $R,S \in \mathbb{S}$: $R \le S \;\Rightarrow\; \varphi R \le' \varphi S$

iii) $\forall S' \in \mathbb{S}' (\exists R,S \in \mathbb{S}\colon\; \varphi R \le' S' \le' \varphi S$ □

Condition iii) implies, that $\varphi\top = \top'$ and that if $R,S \in \mathbb{S}$ have no common subsort, then the same holds for φR and φS.

2.4.2 Lemma. Let $\langle \mathbb{S},\le \rangle$ be a sort structure with a greatest element S. Then there exists a sort structure $\langle \mathbb{S}',\le' \rangle$ such that $\langle \mathbb{S},\le \rangle$ is embedded into $\langle \mathbb{S}',\le' \rangle$ and $\langle \mathbb{S}',\le' \rangle$ is a semilattice.

Proof. For $S \in \mathbb{S}$, let $\varphi S := \{R \in \mathbb{S} \mid R \le S\}$. We define $\mathbb{S}'$ as the following set:

$\mathbb{S}' = \{ M \mid M \neq \emptyset$ and $M = \varphi S_1 \cap \ldots \cap \varphi S_n$ for some $S_i \in \mathbb{S}\}$, i.e. $\mathbb{S}'$ is the set $\varphi\mathbb{S}$ extended by all possible intersections, which are not empty. This definition is similar to the definition of the lattice of ideals in lattices [Gr79]. Obviously $\mathbb{S}'$ is finite.

We let the relation $\le'$ on $\mathbb{S}'$ be the subset ordering. To show that $\langle \mathbb{S}',\le' \rangle$ is a semilattice, it suffices to show that for all $R',S' \in \mathbb{S}'$ a greatest lower bound exists, if lower bounds of R',S' exist at all. $R' \cap S'$ is the greatest lower bound, if it is not empty.

We prove, that $\langle \mathbb{S},\le \rangle$ that is embedded into $\langle \mathbb{S}',\le' \rangle$:

i) φ is injective, since $\varphi R = \varphi S$ implies, that $R \in \varphi R$ and $S \in \varphi S$, hence by the rule of antisymmetry: $R = S$.

ii) For $R,S \in \mathbb{S}$, $R \le S$ implies that $\varphi R \subseteq \varphi S$, hence $\varphi R \le' \varphi S$.

iii) $\varphi T = \mathbf{S}$ is the greatest element of $\mathbf{S}'$. Let $S' \in \mathbf{S}'$. S' is not empty, so take some $S \in S'$. We have $S' = \varphi S_1 \cap \ldots \cap \varphi S_n$ for some $S_i \in \mathbf{S}$.
Now $S \in S'$ implies that $S \leq S_i$ for $i=1,\ldots,n$, hence $\varphi S \subseteq S'$ and so $\varphi S \leq' S' \leq \mathbf{S}$. □

<u>2.4.3 Definition.</u> Let SIG = $(\mathbf{S}, \mathbf{F}, \mathbf{P})$ and SIG$'$ = $(\mathbf{S}', \mathbf{F}', \mathbf{P}')$ be signatures.
We say <u>SIG is embedded into SIG$'$</u>, iff the following holds:

i) $\mathbf{S} \subseteq \mathbf{S}'$, $\mathbf{F} = \mathbf{F}'$, $\mathbf{P} = \mathbf{P}'$

ii) $\langle \mathbf{S}, \leq \rangle$ is embedded in $\langle \mathbf{S}', \leq' \rangle$ with a mapping φ which is the identity on $\mathbf{S}$.

iii) $[t] = [t]'$ for all $t \in \mathbf{WST}$.

iv) $\mathbf{WST} \subseteq \mathbf{WST}'$ and for all $S \in \mathbf{S}$: $\mathbf{V}_S{}' = \mathbf{V}_S$, i.e. The subalgebra of $\mathbf{WST}'$ which is generated by all $f \in \mathbf{F}$ and all variables with sorts in $\mathbf{S}$ equals $\mathbf{WST}$.

<u>2.4.4 Lemma.</u> Let SIG = $(\mathbf{S}, \mathbf{F}, \mathbf{P})$ be a signature and let $\langle \mathbf{S}', \leq' \rangle$ be a semilattice such that $\langle \mathbf{S}, \leq \rangle$ is embedded into $\langle \mathbf{S}', \leq' \rangle$ with a mapping φ which is the identity on $\mathbf{S}$.
Then there exists a signature SIG$'$ - $(\mathbf{S}', \mathbf{F}', \mathbf{P}')$ such that SIG is embedded in SIG$'$.

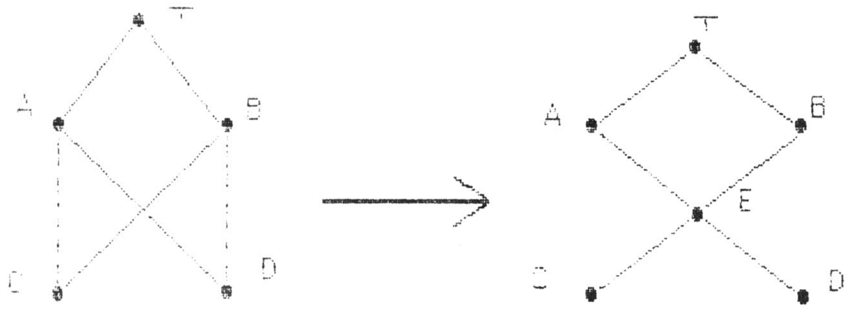
<u>Proof.</u> We define $\mathbf{F}' := \mathbf{F}$, $\mathbf{P}' := \mathbf{P}$.
For $c \in \mathbf{C}$, let $[c]' := [c]$.
For $x \in \mathbf{V}$, let $[x]' := [x]$.
For functions $f \in \mathbf{F}$, SO$'$(f) is defined as the set:

$$\left\{ (S_1{}',\ldots,S_{n+1}{}') \;\middle|\; \begin{array}{l} (S_1{}',\ldots,S_{n+1}{}') \leq (S_{f,1},\ldots,S_{f,n+1}) \text{ where } (S_{f,1},\ldots,S_{f,n+1}) \text{ is} \\ \text{the maximum element of SO(f). } S_{n+1}{}' \text{ is the g.l.b. of} \\ \{S_{n+1} \mid (S_1,\ldots,S_{n+1}) \in \mathrm{SO(f)} \text{ and } (S_1{}',\ldots,S_n{}') \leq (S_1,\ldots,S_n)\,\} \end{array} \right\}$$

The only nontrivial conditions in Definition 1.1 are that for every function f: f^* is a monotone function defined on $\{(S_1{}',\ldots,S_n{}') \mid (S_1{}',\ldots,S_n{}') \leq (S_{f,1},\ldots,S_{f,n})\,\}$ and that SIG$'$is sensible.
Let $(S_{1,1}{}',\ldots,S_{1,n}{}') \leq (S_{2,1}{}',\ldots,S_{2,n}{}')$. $S_{1,n+1}{}'$ is the g.l.b. of the set $\{S_{n+1} \mid (S_1,\ldots,S_{n+1}) \in \mathrm{SO(f)}$ and $(S_{2,1}{}',\ldots,S_{2,n}{}') \leq (S_1,\ldots,S_n)\,\}$,
whereas $S_{2,n+1}{}'$ is the g.l.b. of the set
$\{S_{n+1} \mid (S_1,\ldots,S_{n+1}) \in \mathrm{SO(f)}$ and $(S_{2,1}{}',\ldots,S_{2,n}{}') \leq (S_1,\ldots,S_n)\,\}$.
We have $S_{1,n+1}{}' \leq S_{2,n+1}{}'$, since the first set is a subset of the latter.
SIG$'$ is sensible since for every $S' \in \mathbf{S}'$ there exists a $S \in \mathbf{S}$ with $S \leq S'$. Hence there exists a $c \in \mathbf{C}$ with $[c] \leq S \leq S'$. We have shown that SIG$'$ is a signature according to Definition 1.1. SO(f) $\subseteq$ SO$'$(f) holds for every $f \in \mathbf{F}$, hence $[t] = [t]'$ for all $t \in \mathbf{WST}$. The fact that the greatest elements of SO(f) and SO$'$(f) are equal and the relation $[t] = [t]'$ for $t \in \mathbf{WST}$ imply condition iv) of Definition 2.4.3. □

We give an example for embedded signatures:

2.4.5 Example. The following transformation is a completion of a sort-structure resulting in a semilattice:



For a polymorphic function f with $SO(f) = \{ (\top,\top), (A,A), (B, B), (C,C)\ (D, D) \}$ we obtain $SO'(f) = \{ (\top,\top), (A,A), (B, B),(E,E), (C,C)\ (D, D) \}$, since E is the g.l.b. of $\{A, B,\top\} = \{S |\ (S_1,S) \in SO(f) \text{ and } E \leq S_1\ \}$. □

2.4.6 Theorem. For a given clause set CS and a signature SIG, let SIG′ be a signature such that SIG is embedded into SIG′ and $\langle \mathbf{S}', \leq' \rangle$ is a semilattice.
Then: CS has an E-model w.r.t SIG $\Longleftrightarrow$ CS has an E-model w.r.t. SIG′.

Proof. "⇒": Let (D,SIG,R^D) be an E-model of CS w.r.t SIG.
We construct an E-model (E,SIG',R^E) w.r.t SIG′:
Let $E := D;\ R^E := R^D;\ f^E := f^D.$

Let $S'^E := \ \{d |\ d \in S^D \text{ for all } S \in \mathbf{S} \text{ with } S' \leq S\}$, i.e. S'^E is the intersection of all related subsets in D for greater sorts.
Lemma 2.4.2 shows, that $S'^E \neq \emptyset$ for all $S' \in \mathbf{S}'$.
Now let $\varphi^E : \mathbf{WST}' \to E$ be a SIG-homomorphism. We define a SIG-homomorphism $\varphi^D : \mathbf{WST} \to D$ w.r.t. SIG by $\varphi^D x := \varphi^E x$ for all $x \in \mathbf{V}(SC)$.
Lemma 1.5 yields, that φ^D exists. For all clauses $C \in CS$ we have $\varphi^D C = \varphi^E C$. Thus $\varphi^E C$ is valid for all clauses C. Hence (E,SIG',R^E) is an E-model for CS (w.r.t SIG′).

"⇐": Let (E,SIG',R^E) be an E-model of CS w.r.t SIG′. We can construct an E-model (D,SIG,R^D) in the same way as above by setting
$D := E,\ R^D := R^E,\ f^D := f^E,\ S^D := S^E.$ □

3. Theoretical Properties of the ΣRP*-calculus.

In this chapter the soundness and completeness of the ΣRP*-calculus and the Sortensatz are shown. The proof technique for the soundness and completeness results are taken from [WR73] and are appropriately modified to fit our sort structure.

The functional reflexivity axioms are necessary for the completeness of the ΣRP*-calculus, in contrast to the RP-calculus (without sorts) where they are superfluous [Ri78, Bra75]. In the ΣRP-calculus this is an open problem [Wa83].

3.1 Completeness of the ΣRP*-calculus.

It is a well-known fact, that every formula in a first order predicate calculus can be transformed into a set of clauses. The skolemization, which removes existential quantifiers from the input formula and replaces the quantified variable by a skolem function is the same as in [Wa83]. Note that the generated skolem functions are not polymorphic.
The deduction rules of the ΣRP*-calculus are tailored for clause sets.

3.1.1 Definition. The inference rules of the ΣRP*-calculus are **S**-resolution, **S**-factoring and **S**-paramodulation.

i) **S**-resolution. Let $C \cup \{L\}$ and $D \cup \{-L'\}$ be variable disjoint clauses and let L and L′ be two atoms with the same predicate symbol. Let σ be an SIG-mgu of L and L′.
Then $\sigma C \cup \sigma D$ is an **S**-resolvent of the two clauses.

ii) **S**-factoring. Let C be a clause and let σ be a SIG-mgu of two or more literals of C.
Then σC is an **S**-factor of C.

iii) **S**-paramodulation. Let $C \cup \{L\}$ and $D = \{s \equiv t\}$ be two variable disjoint clauses. Let pos be a position within L and let $L_{|pos} - t_0$. Let σ be a most general paramodulation unifer for L,pos and $s \equiv t$.
Then $\sigma C \cup \sigma D \cup \{\sigma L[pos \leftarrow \sigma t]\}$ is the **S**-paramodulant of the two clauses. □

Note that there may not exist an **S**-resolvent (an **S**-factor, an **S**-paramodulant) for two given clauses although an ordinary resolvent does exist. In the following we drop the **S**- in **S**-resolvent , **S**-factor, **S**-paramodulant if no confusion arises.
We assume that resolvents, factors and paramodulants are renamed before they are added to the original clause set.

3.1.2 Lemma. Let CS be a clause set and let D be a factor, resolvent or paramodulant of clauses in CS.
Then: CS is satisfiable $\Longleftrightarrow$ $CS \cup \{D\}$ is satisfiable.

Proof. The proof for the unsorted case is easily adapted to the many-sorted case. □

3.1.3 Definition. The functional reflexivity axioms are the following axioms:

i) $x \equiv x$ with $x \in \mathbf{V}_T$.

ii) $f(x_1,...,x_n) \equiv f(x_1,...,x_n)$ for every $f \in \mathbf{F} \setminus \mathbf{C}$, where $[x_i] = S_{f,i}$ and $(S_{f,1},...,S_{f,n+1})$ is the greatest element of SO(f) and all variables x_i are different. □

A clauses set, which allows the deduction of all functional reflexivity axioms, is called functionally reflexive.

If the functional reflexivity axioms are present, then it is possible to deduce instances of clauses by paramodulation with these axioms:

For every instance σC_0 of a clause C_0 there exist substitutions σ_{FR} and σ_0, such that $(\sigma_0 \circ \sigma_{FR})C_0 = \sigma C_0$. The instance $\sigma_{FR}C_0$ is deducable from C_0 and the functional reflexivity axioms and has the same term positions as σC_0. The substitution σ_0 does not change the term depth.

However in practical applications the functional reflexivity axioms are not used, since i) they increase the search space enormously and ii) the subsumption rule would delete all of them but the axiom $x \equiv x$.
The instances described above would also be deleted by the subsumption rule.

The next lemma is independent of the rest of this paragraph. It shows, that the presence of the functional reflexivity axioms makes the relation ~ on terms to be a congruence relation, where s ~ t, iff $s \equiv t$ is an instance of a deducable unit equation. This may be false, if the functional reflexivity axioms are absent. The difficulties in proving the completeness of paramodulation arise from this fact.

3.1.4 Lemma. Let CS be a functionally reflexive clause set. Let ~ be the relation on $\mathbf{WST}_{gr}$ defined as:

$s \sim t$, iff there exist s' and $t' \in \mathbf{WST}$ and a $\sigma \in \mathbf{\Sigma}$ such that

$$CS \vdash s' \equiv t' \text{ and } \sigma(s' \equiv t') = (s \equiv t).$$

Then ~ is a congruence relation.

Proof. From the axiom $x \equiv x$ and the paramodulation rule, we obtain that ~ is an equivalence relation.

We show that ~ is a congruence relation:

Let f be a function with n arguments and let $s_i \sim t_i$, $i=1,...,n$ and $s_i,t_i \in \mathbf{WST}_{gr}$. We have to show that $f(s_1,...,s_n) \sim f(t_1,...,t_n)$ if both terms are well-sorted.

There exist s_i', t_i' and σ_i', such that $CS \vdash s_i' \equiv t_i'$ and $\sigma_i(s_i' \equiv t_i') = (s_i \equiv t_i)$ for $i=1,...,n$. Multiple paramodulation with the axiom $f(x_1,...,x_n) \equiv f(x_1,...,x_n)$ yields $CS \vdash f(s_1',...,s_n') \equiv f(t_1',...,t_n')$. We can assume that the substitutions σ_i are variable disjoint. Let $\sigma := \sigma_1 \circ ... \circ \sigma_n$. Then $\sigma f(s_1',...,s_n') = f(s_1,...,s_n)$ and $\sigma f(t_1',...,t_n') = f(t_1,...,t_n)$. □

3.1.5 Example. This example shows, that the paramodulation rule is not complete, if the functional reflexivity axioms are absent.

Let $\mathbf{S} = \{\top,A,B,C,D,E\}$, $\top \geq A,B,C,D,E$.
Let a,b,c,d be constants of sort A,B,C,D respectively.
Let $P \in \mathbf{P}$, with $SO(P) = E$
Let $f \in \mathbf{F}$ with

$$[f(x_R,x_S)] = \begin{cases} E & \text{if } (R \leq A \text{ and } S \leq C) \text{ or } (R \leq B \text{ and } S \leq D) \\ \top & \text{else} \end{cases}$$

The clause set CS is:
$\{ \{ a \equiv b\}; \{c \equiv d\}; \{P(f(a\ c))\}; \{-P(f(b\ d))\}$.
This clause set is unsatisfiable.
Paramodulation is not possible, since $[f(b\ c)] = [f(a\ d)] = \top$. There is no deduction of the empty clause in ΣRP^*. If the functional reflexivity axioms are present, then $f(a\ c) \equiv f(b\ d)$ can be deduced from the axioms $f(x\ y) \equiv f(x\ y)$; $a \equiv b$; $c \equiv d$ by two paramodulations. □

We conjecture, that a "parallel" paramodualtion rule instead of normal paramodulation avoids the functional reflexivity axioms.

3.1.6 Definition. Let SIG be a signature and let CS be a clause set. An interpretation I of CS is a set of well sorted ground literals, such that for each well-sorted atom A, either +A or -A is in I. □

The interpretation I is an HE-model of CS, iff $C_{gr} \cap I \neq \emptyset$ for every ground instance of every clause $C \in CS$.

3.1.7 Theorem. (Maximal Model Theorem [WR73]).
Let CS be a clause set and let I be an interpretation of CS.
If CS has an HE-model, then there exists an HE-model M with the following property:
For every literal $L \in M \cap I$ there exists a clause $C \in CS$, which has a ground instance C_{gr} such that $C_{gr} \cap M = \{L\}$. □

3.1.8 Lemma. (Lifting of factors. Lemma 1 of [WR73]).

Let CS be a clause set, $A \in CS$ and $\tau \in \mathbf{\Sigma}$ such that $A' = \tau A$ is a ground instance of A.

Then there exists a factor F_A of A and a $\mu \in \mathbf{\Sigma}$, such that:

i) $A' = \mu F_A$.

ii) F_A and A' have the same number of literals.

Proof. The SIG-substitution τ partitiones A in congruence classes, if the relation is $L_1 \sim L_2$, iff $\tau L_1 = \tau L_2$. By Theorem 2.2.4 a set of SIG-mgu's exists, which simultaneously unify the literals in every congruence class. There exists an element θ out of this set and a $\mu \in \mathbf{\Sigma}$ with $\tau = \mu \circ \theta$ $[\mathbf{V}(A)]$.

Let $F_A := \theta A$. Then $\mu F_A = \mu \circ \theta A = \tau A = A'$ and the number of literals in F_A and A' are the same. □

3.1.9 Lemma. (Lifting of paramodulation, special case, Lemma 2 of [WR73]).

Let A',B' be ground instances of the clauses A and B, which are in the clause set CS, and let D' be a paramodulant of A' and B', i.e.

$D' = (A' \setminus \{s' \equiv t'\}) \cup (B' \setminus L') \cup \{L'[q \leftarrow t']\}$.

Further assume that the literal $s' \equiv t'$ is in A' and that L' is a literal in B'. Let q be a position vector and let $r' = L'_{|q}$. Let L be a literal in B and $\mu \in \mathbf{\Sigma}$ such that $r = L_{|q}$ exists and that $\mu B = B'$ and $\mu L = L'$.

Then there exists a clause D having D' as an instance such that D is a paramodulant of some factors F_A, F_B of A and B respectively.

Proof. From Lemma 3.1.8 we conclude that factors F_A, F_B of A and B exist and that there exist $\theta, \tau \in \mathbf{\Sigma}$ with $\tau F_A = A'$, $\tau F_B = B'$, $\theta A = F_A$, $\theta B = F_B$ and τ does not change the number of literals in F_A, F_B.

We assume without loss of generality that s' is the argument involved. There exists a literal $s \equiv t$ in A, such that

$\tau\theta(s \equiv t) = s' \equiv t'$ and $\tau\theta s = s' = \tau\theta r$. Now Theorem 2.3.3 states that there exists a most general SIG-unifier σ for paramodulation and a $\lambda \in \mathbf{\Sigma}$ with:

$\tau = \lambda \circ \sigma$ $[\mathbf{V}(F_A, F_B)]$, $\sigma \circ \tau s = \sigma \circ \tau r$ and the paramodulant

$D = \sigma(F_A \setminus \{\theta s \equiv \theta t\}) \cup \{\sigma(F_A \setminus \theta L)\} \cup \{(\sigma \circ \theta L)[q \leftarrow \sigma \circ \theta t]\}$

of F_A and F_B is well-sorted.

Note that $\theta L_{|q}$ exists because $L_{|q}$ exists. Now we have $\lambda D = D'$. □

3.1.10 Lemma.(Lifting of paramodulation, general case, Lemma 2 of [WR73])

Let CS be a functionally reflexive clause set, closed under both paramodulation and factoring. If A' and B' are ground instances of A and B, which are in CS, and if D' is a paramodulant of A' and B' , then there exists a clause D in CS having D' as an instance.

Proof. Let $s' \equiv t' \in A'$, $L' \in B'$ and let q be a position such that $r' = L'_{|q}$. The paramodulation is on the terms s' and r'. Let L be a literal in B and $\mu \in \Sigma$, such that $\mu B = B'$ and $\mu L = L'$.

Case 1. $L_{|q}$ exists. Then Lemma 3.1.9 is applicable.

Case 2: $L_{|q}$ does not exist. We have $\mu = \mu_0 \circ \mu_{FR}$, where $\mu_{FR}(B)$ is deducable by paramodulation with the functional reflexivity axioms and $\mu_{FR}(B)_{|q}$ exists. Now Lemma 3.1.9 is applicable. □

In the following Theorems the lifting lemmas above play a central role. We remark that the application of such lifting lemmas requires that (renamed) copies of clauses are available for resolution and paramodulation.

3.1.11 Theorem. The many-sorted calculus ΣR^* with resolution and factorization as inference rules is sound and complete.
i.e. $\vdash_{\Sigma R^*} \Longleftrightarrow \models$

Proof. (cf. [WR73]). The only modification of the proof given there is to replace mgu by a set of SIG-mgu's. □

3.1.12 Theorem. Let SIG be the signature of a functionally reflexive clause set and let CS^* be the closure of CS under paramodulation and factorization. If no HE-model of CS^* exists, then CS^* has no H-model (where equality is interpreted as a normal predicate).

Proof. Assume by contraposition that CS^* has an H-model. Let I be the set of a well-sorted atoms
and let M be the maximal model of CS^* (see Theorem 3.1.7). We show, that M is an HE-model of CS^*:
Since $\{x \equiv x\} \in CS^*$, we have $t \equiv t \in M$ for all ground terms t. Let $t_1 \equiv t_2 \in M$, $L \in M$ and pos a position vector, such that $t_1 = L_{|pos}$ and $L[pos \leftarrow t_2]$ is a well-sorted literal. By Theorem 3.1.7 there exist clauses $C, D \in CS^*$ and corresponding ground clauses C_{gr}, D_{gr} with $C_{gr} \cap M = \{t_1 \equiv t_2\}$ and $D_{gr} \cap M = \{L\}$. The paramodulant of C_{gr} and D_{gr} is $P_{gr} = (C_{gr} \setminus \{t_1 \equiv t_2\}) \cup (D_{gr} \setminus \{L\}) \cup \{L[pos \leftarrow t_2]\}$. By Lemma 3.1.10 there exist a clause $P \in CS^*$ such that P_{gr} is an instance of P. M is a H-model of CS^*, hence $M \cap P_{gr} \neq \emptyset$. Obviously $M \cap P_{gr} = \{L[pos \leftarrow t_2]\}$. We have shown, that M is an HE-model of CS^*. □

3.1.13 Theorem. The ΣRP*-calculus is sound and complete, if the functional reflexivity axioms are present, i.e.

$$\vdash_{\Sigma RP^*} \iff \models$$

Proof. [WR73] Soundness is trivial.

Completeness: Let CS be a clause set and let CS* be the closure under paramodulation, resolution and factoring. Let the functional reflexivity axioms be in CS*. If CS* has no HE-model, then by Theorem 3.1.12 the set CS* has no H-model. Hence by Theorem 3.1.11 the empty clause is in CS*. The empty clause is the last line of a finite deduction of clauses in CS. Hence a refutation can be found. □

3.2 The Sortensatz.

In this paragraph the important "Sorten"-theorem (Sortensatz [Wa83,Ob62]) is shown to hold for the polymorphic ΣRP^*-calculus. The Sortensatz provides the essential link between a sorted clause set and its unsorted version.

3.2.1 Definiton. Let SIG be a signature for the clause set CS.

We define $CS_{rel} \cup A^{\Sigma}$, the relativized clause set of CS as:

i) The relativized signature SIG_{rel} has only one sort. For every $S \in \mathbf{S}$, there is a unary predicate $P_S \in \mathbf{P}_{rel}$, which is not in $\mathbf{P}$.

ii) The set CS_{rel} is the set of relativized clauses:

$CS_{rel} := \{C_{rel} | C \in CS\}$, where $C_{rel} = -P_{S1}(x_1) \vee \ldots \vee -P_{Sn}(x_n) \vee C$ for a $C \in CS$. $\{x_1,\ldots,x_n\} = \mathbf{V}(C)$ and $S_i = [x_i]$.

iii) The set of sort axioms A^{Σ} consists of:

$-P_{S1}(x) \vee P_{S2}(x)$ for all S_1, $S_2 \in \mathbf{S}$ with $S_1 \le S_2$.

$P_T(x)$

$-P_{S1}(x_1) \vee \ldots \vee -P_{Sn}(x_n) \vee P_{Sn+1}(f(x_1,\ldots, x_n)$ for every $(S_1,\ldots,S_{n+1}) \in SO(f)$.□

3.2.2 Theorem. (Sortensatz).

Let SIG be a signature and let CS be a SIG-sorted clause set. Then:

CS is satisfiable $\Longleftrightarrow$ $CS_{rel} \cup A^{\Sigma}$ is satisfiable.

Proof. "⇒": Let (D,SIG,R) be an E-model of CS.

We have to construct an E-model for $CS_{rel} \cup A^{\Sigma}$ w.r.t. SIG_{rel}.

Let $E := D$ and $f^E := f^D$ for all $f \in \mathbf{F}$. $R_{rel} := R \cup \{P_S{}^E | S \in \mathbf{S}\}$, where $P_S{}^E(d)$ is valid, iff $d \in S^D$.

(E, SIG_{rel}, R_{rel}) is an E-model for $CS_{rel} \cup A^{\Sigma}$:

Let $\varphi_E : \mathbf{WST}_{rel} \to E$ be a SIG-homomorphism.

We show, that all clauses are true under φ_E:

i) C_{rel}: Either one of $\varphi_E(-P_{Si}(x_i))$ is true, or all φ_E $-P_{Si}(x_i)$ are false. In the second case, there exists a SIG-homomorphism φ_D with $\varphi_D(x_i) = \varphi_E(x_i)$, $i=1,\ldots,n$ since $\varphi_E(x_i) \in S_i{}^D$. But then $\varphi_E(C) = \varphi_D(C)$ evaluates to true since (D,SIG,R) is an E-model of CS.

ii) Let $S_1 \le S_2$: then $S_1{}^D \subseteq S_2{}^D$. Hence for every $d \in E$: either $d \notin S_1{}^D$ or $d \in S_2{}^D$, hence $\varphi_E(P_{S1}(x) \vee P_{S2}(x))$ is true.

iii) $\varphi_E(P_T(x))$ is true.

iv) The axiom $-P_{S_1}(x_1) \vee \ldots \vee -P_{S_n}(x_n) \vee P_{S_{n+1}}(f(x_1,\ldots, x_n))$ for every $(S_1,\ldots,S_{n+1}) \in SO(f)$ is true under φ_E , since f^D is a mapping $f^D: S_1{}^D \times \ldots \times S_n{}^D \rightarrow S_{n+1}{}^D$.

"$\Leftarrow$": Let (E, SIG_{rel}, R_{rel}) be an E-model for $CS_{rel} \cup A^{\Sigma}$.

We have to define an E-model for CS.

Let $D := E$ and $f^D := f^E$ for all $f \in \mathbf{F}$. For $S \in \mathbf{S}$, let $S^D := \{d | P_S{}^D(d)$ is valid$\}$.

We verify the conditions of the definition of an algebra w.r.t. SIG:

a) We have $\top^D := D$.

b) Let $S_1, S_2 \in \mathbf{S}$ with $S_1 \leq S_2$. Then $-P_{S_1}(x) \vee P_{S_2}(x)$ is valid in (E, SIG_{rel}, R_{rel}). Hence for $d \in S_1{}^D$, we have that $P_{S_2}{}^D(d)$ is true and so $d \in S_2{}^D$.

c) The condition for constants holds trivially.

d) Let $f \in \mathbf{F} \setminus \mathbf{C}$ and let $(S_1,\ldots,S_n) \in SO(f)$. Then $-P_{S_1}(x_1) \vee \ldots \vee -P_{S_n}(x_n) \vee P_{S_{n+1}}(f(x_1,\ldots, x_n))$ is true in (E, SIG_{rel}, R_{rel}). For elements $d_i \in S_i{}^D$, $P_{S_{n+1}}{}^D(f(d_1,\ldots, d_n))$ is true, hence $f(d_1,\ldots, d_n) \in S_{n+1}{}^D$. □

Let $R := R_{rel} \setminus \{P_S{}^D | S \in \mathbf{S}\}$.

Then (D, SIG, R) is an E-model of C.

The last property to show is that all clauses are true under all SIG-homomorphisms.

Let $\varphi_D : \mathbf{WST} \rightarrow D$ be a SIG-homomorphism and let C be a clause.

Then a SIG-homomorphism $\varphi_E : \mathbf{WST}_{rel} \rightarrow E$ exists with $\varphi_D(x) = \varphi_E(x)$ for all $x \in \mathbf{V}$. φ_E evaluates all literals $P_S(x_i)$ of C_{rel} to false, hence $\varphi_E(C)$ is true.

Therefore C is true under φ_E. □

4. The Lion & Unicorn Example.

4.1 Example. This example is taken from Smullyan: "What is the name of this book?" [SM78], which appears to be a goldmine for theorem proving examples. During a course on automated theorem proving last fall at the Universitiy of Kaiserslautern, our students had to translate these puzzles into first order predicate logic and to solve them with our theorem prover (Markgraf Karl Refutation Procedure, MKRP) [KM84,NN86]. One of these problems (Problem 47) reads as follows:

"When Alice entered the forest of furgetfulness, she did not forget everything, only certain things. She often forgot her name, and the most likely to forget was the day of the week. Now, the lion and the unicorn were frequent visitors to this forest. These two are strange creatures. The lion lies on Mondays, Tuesdays and Wednesdays and tells the truth on the other days of the week. The unicorn, on the other hand lies on Thursdays, Fridays and Saturdays, but tells the truth on the other days of the week.

One day Alice met the lion and the unicorn resting under a tree. They made the following statements:

Lion: Yesterday was one of my lying days.
Unicorn: Yesterday was one of my lying days.

From these statements, Alice who was a bright girl, was able to deduce the day of the week. What was it?"

We use the predicates MO(x), TU(x), ... , SO(x) for saying that x is a Monday, Tuesday etc. Furthermore we need the binary predicate MEMB, indicating set membership and a 3-ary predicate LA. LA(x y z) is true if x says at day y that he lies at day z; ldays(x) denotes the set of lying days of x. The remaining symbols are self explaining. All one-character symbols like u,x,y,z are regarded as universally quantified variables.

Axiomatization of the days of the week:

$$\begin{array}{lcl}
MO(x) & \Leftrightarrow & \neg(TU(x) \vee WE(x) \vee TH(x) \vee FR(x) \vee SA(x) \vee SU(x)\ ) \\
TU(x) & \Leftrightarrow & \neg(WE(x) \vee TH(x) \vee FR(x) \vee SA(x) \vee SU(x) \vee MO(x)\ ) \\
WE(x) & \Leftrightarrow & \neg(TH(x) \vee FR(x) \vee SA(x) \vee SU(x) \vee MO(x) \vee TU(x)\ ) \\
TH(x) & \Leftrightarrow & \neg(FR(x) \vee SA(x) \vee SU(x) \vee MO(x) \vee TU(x) \vee WE(x)\ ) \\
FR(x) & \Leftrightarrow & \neg(SA(x) \vee SU(x) \vee MO(x) \vee TU(x) \vee WE(x) \vee TH(x)\ ) \\
SA(x) & \Leftrightarrow & \neg(SU(x) \vee MO(x) \vee TU(x) \vee WE(x) \vee TH(x) \vee FR(x)\ ) \\
SU(x) & \Leftrightarrow & \neg(MO(x) \vee TU(x) \vee WE(x) \vee TH(x) \vee FR(x) \vee SA(x)\ )
\end{array}$$

Axiomatization of the function yesterday:

$$\begin{array}{lcl}
MO(yesterday(x)) & \Leftrightarrow & TU(x) \\
TU(yesterday(x)) & \Leftrightarrow & WE(x) \\
WE(yesterday(x)) & \Leftrightarrow & TH(x) \\
TH(yesterday(x)) & \Leftrightarrow & FR(x) \\
FR(yesterday(x)) & \Leftrightarrow & SA(x) \\
SA(yesterday(x)) & \Leftrightarrow & SU(x) \\
SU(yesterday(x)) & \Leftrightarrow & MO(x)
\end{array}$$

Axiomatization of the function ldays:

$MEMB(x\ ldays(lion)) \quad \Leftrightarrow MO(x) \vee TU(x) \vee WE(x)$

$MEMB(x\ ldays(unicorn)) \quad \Leftrightarrow TH(x) \vee FR(x) \vee SA(x)$

Axiomatization of the predicate LA:

$\neg MEMB(x\ ldays(u)) \wedge\ LA(u\ x\ y) \quad \Rightarrow\ MEMB(y\ ldays(u))$

$\neg MEMB(x\ ldays(u)) \wedge \neg LA(u\ x\ y) \quad \Rightarrow \neg MEMB(y\ ldays(u))$

$MEMB(x\ ldays(u)) \wedge\ LA(u\ x\ y) \quad \Rightarrow \neg MEMB(y\ ldays(u))$

$MEMB(x\ ldays(u)) \wedge \neg LA(u\ x\ y) \quad \Rightarrow\ MEMB(y\ ldays(u))$

Theorem:

$\exists x\ LA(lion\ x\ yesterday(x)) \wedge LA(unicorn\ x\ yesterday(x))$

The MKRP theorem proving system found a proof for this unsorted version after 183 resolution steps, among them 81 unnecessary steps, hence the final proof was 102 steps long. This proof contains a lot of trivial steps corresponding to common sense reasoning (like: if today is Monday, it is not Tuesday etc.).

Later the sort structure and the signature of the problem at hand was generated automatically by a translator module which accepts an unsorted clause set as input and produces the equivalent many-sorted version together with the corresponding signature [Sch85,OS85].

The sort structure and the signature contain all the relevant information about the relationship of unary predicates (like our days) and the domain-rangesort relation of functions. The sort structure of the subsorts of DAYS in our example is equivalent to the lattice of subsets of {Mo, Tu, We, Th, Fr, Sa, Su} without the empty set, ordered by the subset order. Hence there are 127 (=2^7-1) sorts. The function "yesterday" is a polymorphic function with 127 domain-sort relations. For example: yesterday ({Mo, We}) = {Su, Tu}.

The unification algorithm exploits this information and produces only unifiers, which respect the sort relations, i.e. $\{x \leftarrow t\}$ is syntactically correct, if and only if the sort of the term t is less or equal the sort of the variable x. We give an example for unification: the unifier of $x{:}So{+}Tu$ and $yesterday(y{:}Mo{+}Tu)$ is $\{x \leftarrow yesterday(y_1{:}\ Mo)\ ;\ y \leftarrow y_1{:}Mo\ \}$.

The MKRP theorem-proving system has proved the theorem in the sorted version immediately without any unnecessary steps. The length of the proof is 6. As the following protocol shows, the final substitution into the theorem clause was $\{x \leftarrow y{:}Th\}$. Thus the ATP has found the answer, Thursday, in a very straight forward and humanlike way. Here is the proof protocol:

```
C1     All x:Mo    MEMB (x ldays(lion))
C2     All x:Tu    MEMB (x ldays(lion))
C3     All x:We    MEMB (x ldays(lion))
C4     All x:Th    MEMB (x ldays(unicorn))
C5     All x:Fr    MEMB (x ldays(unicorn))
C6     All x:Sa    MEMB (x ldays(unicorn))
C7     All x,y:Days u:Animal
              MEMB(x ldays(u))  ¬LA(u x y)  MEMB (y ldays(u))
C8     All x,y:Days u:Animal
              MEMB(x ldays(u))  LA(u x y)  ¬MEMB(y ldays(u))
C9     All x,y:Days u:Animal
              ¬MEMB(x ldays(u))  ¬LA(u x y)  ¬MEMB(y ldays(u))
C10    All x,y:Days u:Animal
              ¬MEMB(x ldays(u))  LA(u x y)  MEMB(y ldays(u))
C11    All x:Th+Fr+Sa+Su    ¬MEMB(x ldays(lion))
C12    All x:Tu+We+Su+Mo ¬MEMB(x ldays(unicorn))
Th     All x:Days  ¬LA(lion x yesterday(x))  ¬LA(unicorn x yesterday(x))
```

Proof:

```
C4,1 & C10,1  →   R1:  All x:Th y:Days LA(unicorn x y)
                                 MEMB(y ldays(unicorn))
R1,2 & C12,1  →   R2:  All x:Th y:Tu+We+Su+Mo  LA(unicorn x y)
C3,1 & C8,3   →   R3:  All x:Days y:We   MEMB(x ldays(lion))   LA(lion x y)
R3,1 & C11,1  →   R4:  All x:Th+Fr+Sa+Su y:We  LA(lion x y)
R4,1 & Th,1   →   R5:  All x:Th ¬LA(unicorn x yesterday(x))
R5,1 & R2,1   →   R6:           □
```

Conclusion

A polymorphic calculus is more expressive than a monomorphic one: one function symbol essentially denotes several operations. An alternative formulation with additional equalities and different function symbols surely produces a greater search space.

The ΣRP^*-calculus allows to express this information and to use it directly in the inference mechanism. In other words, the ΣRP^*-calculus has all the advantages of a many-sorted calculus, but with an improved expressive power.

Polymorphic unification is implemented in the MKRP theorem proving system and shows remarkable improvements in particular in combination with a sort generating algorithm [Sch85] (see Example 4.1), which automatically transforms a given problem into it's polymorphic, many-sorted version.

The paramodulation rule, which is used in the ΣRP^*-calculus, shows some deficiencies: the functional reflexivity axioms are necessary for completeness and also there has to be a greatest element in the signature of a function. Moreover, the presence of the functional reflexivity axioms and the subsumption rule are incompatible. Hence some other type of equational reasoning such as rewriting, demodulation, E-resolution or equality-nets [WR67,Lo80,KM84], may be more appropriate in an automated theorem prover based on ΣRP^*.

The ΣRP^*-calculus could advantageously be used as the basis of a typed PROLOG with overloading, where the present (Robinson) unification algorithm would be replaced by the algorithm proposed in this paper.

Acknowledgements.

I would like to thank: my colleagues H.J. Ohlbach, A. Herold H.J. Bürckert, N. Eisinger and Ch. Walther for their support and for their helpful discussions during the preparation of this paper; special thanks go to supervisor J. Siekmann, who encouraged me to write this paper and read all drafts of this paper.

References

Bra75 Brand, D.
Proving Theorems with the Modification Method. SIAM Journal of Computing 4, (1975)

CL73 Chang, C.-L., Lee, R.C.
Symbolic Logic and Mechanical Theorem Proving. Academic Press (1973)

CD83 Cunningham, R.J., Dick, A.J.J.,
Rewrite Systems on a Lattice of Types. Rep. No. DOC 83/7, Imperial College, London SW7 (1983)

Co83 Cohn, A.G.
Improving the Expressiveness of Many- sorted Logic. AAAI-83, Washington (1983)

Fa83 Fages, F.
Formes canonique dans les algèbres booleènnes et application à la dèmonstration automatique en logique de premier ordre. Thèse du 3$^{\text{ème}}$ cycle, Paris, (1983)

GM84 Goguen, J.A., Meseguer, J.
Equality, Types, Modules and Generics for Logic Programming, Journal of Logic Programming, (1984)

GM85 Goguen, J.A., Meseguer, J.
Order Sorted Algebra I. Partial and Overloaded Operators, Errrors and Inheritance. SRI Report (1985)

Gr79 Grätzer, G.
Universal algebra, Springer Verlag, (1979)

Hay71 Hayes, P.
A Logic of Actions. Machine Intelligence 6, Metamathematics Unit, University of Edinburgh (1971)

Hen72 Henschen, L.J.
N-Sorted Logic for Automated Theorem Proving in Higher-Order Logic. Proc. ACM Conference, Boston (1972)

Hu76 Huet, G.
Resolution d'equations dans des languages d'ordere 1,2,...,ω ;
These d'Etat, Univ. de Paris, VII, (1976)

HO80 Huet, G., Oppen, D.C.,
Equations and Rewrite Rules, SRI Technical Report CSL-111, (1980)

KM84 Karl Mark G Raph,
The Markgraf Karl Refutation Procedure, Memo-SEKI-MK-84-01, (1984)

Lo78 Loveland, D.
Automated Theorem Proving, North Holland, (1978)

Mi84 Mishra, P.
Towards a Theory of types in PROLOG. Int. Symp. on Logic Programming (1984)

MO84 Mycroft, A., O'Keefe, R.
A Polymorphic Type System for PROLOG. Artificial Intelligence 23 (1984)

NN86 The computer generated solutions for the whole book are to appear as a SEKI-research report, Univ. Kaiserslautern.

Ob62 Oberschelp, A.
Untersuchungen zur mehrsortigen Quantorenlogik. Mathematische Annalen 145 (1962)

OS85 Ohlbach, H.J., Schmidt-Schauss, M.; Problem corner JAR (1985)

Ro65 Robinson, J.A. A Machine-Oriented Logic
Based on the Resolution Principle. JACM 12 (1965)

Sch85 Schmidt-Schauss, M.
Mechanical Generation of Sorts in Clause Sets. Interner Bericht. Institut für Informatik, Kaiserslautern (forthcoming)

Si84 Siekmann, J. H.
Universal Unification. 7th Int CADE, Napa, California (1984)

Sm78 Smullyan, R.M.
What is the Name of this Book? Prentice Hall
(1978)

Wa83 Walther, C.
A Many-Sorted Calculus Based on Resolution and Paramodulation. Proc. of the 8th IJCAI, Karlsruhe, (1983)

Wa84 Walther, C.
Unification in Many-Sorted Theories. Proc. of the 6th ECAI, PISA, (1984)

WR73 Wos, L. Robinson, G.
Maximal Models and Refutation Completeness: Semidecision Procedures in Automatic Theorem Proving. In "Wordproblems" (W.W. Boone, F.B. Cannonito, R.C. Lyndon, eds.), North-Holland (1973)