Many-Sorted Structures

A $\Sigma_\mathcal{T}$-algebra is a quadruple

\[ \mathcal{A} = (U_\mathcal{A}, \{ (f_\mathcal{A} : (T_1)_\mathcal{A} \times \ldots \times (T_n)_\mathcal{A} \to S_{\mathcal{A}})_{f \in \Omega}, 
\quad (p_\mathcal{A} \subseteq (S_1)_\mathcal{A} \times \ldots \times (S_m)_\mathcal{A} \}_{p \in \Pi}, 
\quad (T_\mathcal{A} \subseteq U_\mathcal{A})_{T \in \mathcal{T}}) \]

where $\text{arity}(f) = n$, $\text{arity}(p) = m$, $\nu(f) = T_1 \ldots T_n S$, $\nu(p) = S_1 \ldots S_m$, $T_\mathcal{A} \neq \emptyset$, $U_\mathcal{A} \neq \emptyset$ is a set, called the universe of $\mathcal{A}$.

The rest of the semantics is identical to the unsorted case, except that valuations respect the sort information.

7 SUP(LA)

Superposition Modulo Linear Arithmetic

- Consider the base specification $\mathcal{SP} = (\Sigma_{\mathcal{LA}}, \mathcal{A}_{\mathcal{LA}})$, where $\Sigma_{\mathcal{LA}} = (\mathbb{Q} \cup \{+, -, \ast\}, \{\geq, \leq, >, <\})$ see Section 2.
- The hierarchic extension of $\mathcal{SP}$ is $\mathcal{SP}' = (\Sigma', N')$, where $\Sigma_{\mathcal{LA}} \subseteq \Sigma'$ and $N'$ is a set of $\Sigma'$ clauses.
- We consider a many-sorted setting, consisting of a base sort, containing all terms of $\Sigma_{\mathcal{LA}}$ plus potentially extension terms from $\Sigma' \setminus \Sigma_{\mathcal{LA}}$, and a general sort containing all other terms.
- A term (a clause) consisting only of $\Sigma_{\mathcal{LA}}$ symbols and base sort variables, is called a base term (base clause).
- For the following results, we need that $\mathcal{A}_{\mathcal{LA}}$ is term-generated, i.e., for any $a \in U_{\mathcal{LA}} (= \mathbb{Q})$ there is a ground term $t \in T_{\Sigma_{\mathcal{LA}}}$ with $\mathcal{A}_{\mathcal{LA}}(t) = a$. This is obvious, because $\mathbb{Q} \subseteq \Sigma_{\mathcal{LA}}$.
- Furthermore, we need that $\mathcal{SP} = (\Sigma_{\mathcal{LA}}, \mathcal{A}_{\mathcal{LA}})$ is compact.
- A model of $\mathcal{A}'$ of $\mathcal{SP}'$, i.e., $\mathcal{A}' \models N'$, is called hierarchic if $\mathcal{A}' \models_{\mathcal{LA}} = \mathcal{A}_{\mathcal{LA}}$.
- A substitution is called simple if it maps variables of the base sort to base terms.
Hierarchic Clauses

A clause \( C = \Lambda \parallel C' \) is called hierarchic if \( \Lambda \) only contains base terms and base literals \((\Sigma_{LA})\) and all base terms in \( C' \) are variables. The semantics of \( C \) is \( \land \Lambda \rightarrow C' \).

Any clause can be equivalently transformed into a hierarchic clause: whenever a sub-term \( t \) whose top symbol is a base theory symbol occurs immediately below a non-base operator symbol, it is replaced by a new base sort variable \( x \) ("abstracted out") and the equation \( x \approx t \) is added to \( \Lambda \). Analogously, if a subterm \( t \) whose top symbol is not a base theory symbol occurs immediately below a base operator symbol, it is replaced by a general variable \( y \) and the disequation \( y \not\approx t \) is added to \( C' \). This transformation is repeated until the clause is hierarchic.

Superposition Modulo LA

**Pos. Superposition:**

\[
\frac{\Lambda_1 \parallel D' \lor t \approx t' \quad \Lambda_2 \parallel C' \lor s[u] \approx s'}{(\Lambda_1, \Lambda_2 \parallel D' \lor C' \lor s[t'] \approx s')\sigma}
\]

where \( \sigma = \text{mgu}(t, u) \) and simple and \( u \) is not a variable.

**Neg. Superposition:**

\[
\frac{\Lambda_1 \parallel D' \lor t \approx t' \quad \Lambda_2 \parallel C' \lor s[u] \not\approx s'}{(\Lambda_1, \Lambda_2 \parallel D' \lor C' \lor s[t'] \not\approx s')\sigma}
\]

where \( \sigma = \text{mgu}(t, u) \) and simple and \( u \) is not a variable.

**Equality Resolution:**

\[
\frac{\Lambda \parallel C' \lor s \not\approx s'}{(\Lambda \parallel C')\sigma}
\]

where \( \sigma = \text{mgu}(s, s') \) and simple.

**Equality Factoring:**

\[
\frac{\Lambda \parallel C' \lor s' \approx t' \lor s \approx t}{(\Lambda \parallel C' \lor t \not\approx t' \lor s \approx t')\sigma}
\]

where \( \sigma = \text{mgu}(s, s') \) and simple.

**Constraint Refutation:**

\[
\frac{\Lambda_1 \parallel \Box \ldots \Lambda_n \parallel \Box}{\Box}
\]

where \( \neg(\land \Lambda_1) \land \ldots \land \neg(\land \Lambda_n) \) is inconsistent in \( A_{LA} \).
Redundancy

A clause $C \in N$ is called redundant if for all simple ground instances $C'$ of $C$ there are simple ground instances $C'_1, \ldots, C'_n$ from $N$ such that $C'_1, \ldots, C'_n \models C''$ and $C'_i \prec C''$ for all $i$.

A hierarchic clause $\Lambda \parallel C$ is called a tautology if $C$ is a tautology or the existential closure of $\bigwedge \Lambda$ is unsatisfiable in $A_{LA}$.

A hierarchic clause $\Lambda_1 \parallel C_1$ subsumes a hierarchic clause $\Lambda_2 \parallel C_2$, if there is a simple matcher $\sigma$ such that $C_1\sigma \subset C_2$ and the universal closure of $\bigwedge \Lambda_2 \rightarrow \bigwedge \Lambda_1\sigma$ holds in $A_{LA}$.

Sufficient Completeness

A set $N$ of clauses is called sufficiently complete with respect to simple instances, if for every model $A'$ of the set of simple ground instances from $N$ and every ground non-base term $t$ of the base sort there exists a ground base term $t'$ such that $t' \approx t$ is true in $A'$.

Completeness of SUP(LA)

The hierarchic superposition calculus modulo LA is refutationally complete for all sets of clauses that are sufficiently complete with respect to simple instances.

The End