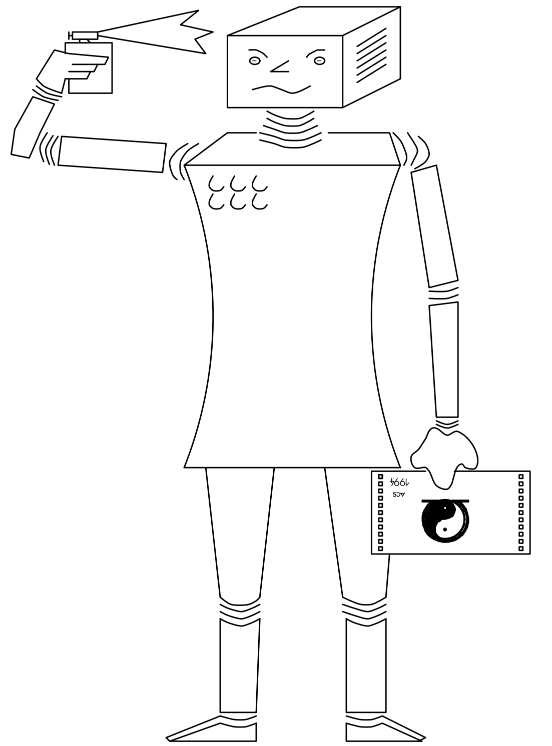

Sound and Complete Translations from Sorted Higher-Order Logic into Sorted First-Order Logic

Manfred Kerber

Published as: In Michael Georgeff, Zhongzhi Shi, Benjamin W. Wah, eds., *Proceedings of PRICAI-94*, 3rd Pacific Rim International Conference on Artificial Intelligence, Beijing, China, 1994, p. 149–154, International Academic Publishers, Beijing, China

SOUND AND COMPLETE TRANSLATIONS FROM SORTED HIGHER-ORDER LOGIC INTO SORTED FIRST-ORDER LOGIC

Manfred Kerber
Fachbereich Informatik, Universität des Saarlandes
Im Stadtwald 15, D-66041 Saarbrücken, Germany
e-mail: kerber@cs.uni-sb.de
Tel.: (+49) 681-302-4628
Fax: (+49) 681-302-4421

Abstract: Extending existing calculi by sorts is a strong means for improving the deductive power of first-order theorem provers. Since many mathematical facts can be more easily expressed in higher-order logic – aside the greater power of higher-order logic in principle –, it is desirable to transfer the advantages of sorts in the first-order case to the higher-order case. One possible method for automating higher-order logic is the translation of problem formulations into first-order logic and the usage of first-order theorem provers. For a certain class of problems this method can compete with proving theorems directly in higher-order logic as for instance with the TPS theorem prover of Peter Andrews or with the Nuprl proof development environment of Robert Constable. There are translations from unsorted higher-order logic based on Church's simple theory of types into many-sorted first-order logic, which are sound and complete with respect to a Henkin-style general models semantics. In this paper we extend corresponding translations to translations of order-sorted higher-order logic into order-sorted first-order logic, thus we are able to utilize corresponding first-order theorem prover for proving higher-order theorems. We do not use any λ-expressions, therefore we have to add so-called comprehension axioms, which a priori make the procedure well-suited only for *essentially first-order* theorems. However, in practical applications of mathematics many theorems are essentially first-order and as it seems to be the case, the comprehension axioms can be mastered too.

1 Introduction

First-order logic is a powerful tool for expressing and proving mathematical facts. Nevertheless higher-order expressions are often better suited for the representation of mathematics and in fact almost all mathematical text books rely on some fragment of higher-order for expressiveness. This fragment can be realized by a higher-order logic itself or by "implementing" parts of it in first-order logic and building it up by a strong set theory. Mathematicians use a technical language, which is relatively informal compared to the formal approaches of logic or set theory. It is much closer to higher-order *sorted* logic augmented by "naive" set theory than to first-order logic. The adjunct of sorts into higher-order logic has the same advantages as their usage in first-order logic: clarity, shorter formulae, and better calculi for proving theorems.

In this paper we show how expressions formulated in higher-order sorted logic can be translated into formulae of first-order sorted logic in order to make use of the strong existing theorem provers for these logics. Of course it would be possible to relativize the sorted higher-order formulae to unsorted ones and then translate these into many-sorted first-order logic, but this would not be optimal, since the expressiveness of order-sorted first-order logic would not be exhausted and the translation would not be structure conserving. An alternative to the translation technique is the usage of higher-order theorem provers like the TPS-system [1] and the extension of Church's simple theory of types to the sorted case [11] or to use a proof development environment like Nuprl [4].

2 Sorted Higher-Order Logics

In this section we introduce formally sorted higher-order logics $\mathcal{L}^n_{\text{sort}}$. To that end, we adopt the notion of sorts for the first-order case of Manfred Schmidt-Schauss [15]. We will follow concept developed by Michael Kohlhase [11] in order to extend higher-order theorem proving to the sorted case.

The Syntax

The syntax of our sorted higher-order logic is similar to the syntax of the unsorted one, that is, Church's simple theory of types [3]. In the sorted case, however, each type (except the truth values) may be structured into subsets. In the following we introduce the basic notions.

$\mathcal{S}$ is the set of *sorts*. $\mathcal{S}$ contains the type symbols ι and o. Whenever $\kappa_1, \ldots, \kappa_m$, and μ are in $\mathcal{S}$ then $(\kappa_1 \times \cdots \times \kappa_m \to \mu)$ is in $\mathcal{S}$. We denote sorts by κ, μ, and ν.

A *simple sort* is a sort that does not contain an arrow "$\to$". All other sorts are called *composed*.

A pair (μ, ν) is called a *subsort declaration* and written as $\mu \dot{\sqsubseteq} \nu$. The *subsort relation* $\dot{\sqsubseteq}$ is the reflexive, transitive closure of the relation $\sqsubseteq$, given by the subsort declarations. We assume $\dot{\sqsubseteq}$ to be closed under covariance in the range sort, that is, for all composed sorts the relations: $(\kappa_1 \times \cdots \times \kappa_m \to \mu) \dot{\sqsubseteq} (\kappa'_1 \times \cdots \times \kappa'_m \to \mu')$ iff $\kappa_1 = \kappa'_1, \ldots, \kappa_m = \kappa'_m$ and $\mu \dot{\sqsubseteq} \mu'$.

A sort μ is called a *top sort* iff for all sorts ν with $\mu \dot{\sqsubseteq} \nu$, ν is equal to μ.

1 Definition (Admissible Subsort Declaration): A subsort declaration is called *admissible* iff the following conditions are fulfilled:

- if $\mu \dot{\sqsubseteq} \nu$ then μ is not equal to ι or o and ν is not equal to o,
- for every simple sort μ there is at least one subsort declaration $\mu \dot{\sqsubseteq} \nu$, so that μ is subsort of a composed sort or of a type symbol (i.e. ι or o).
- in every subsort declaration $\mu \dot{\sqsubseteq} \nu$, μ must be a simple sort and type(μ) = type(ν) (the function type calculates the standard simple type of a sort in the obvious way: type(ι) = ι, type(o) = o, type($\kappa_1 \times \cdots \times \kappa_m \to \mu$) = (type($\kappa_1$) $\times \cdots \times$ type(κ_m) $\to$ type(μ)), and for $\mu \dot{\sqsubseteq} \nu$, type($\mu$) = type($\nu$).)
- there are only finitely many subsort declarations.

Remark: In the following we assume that all subsort declarations are admissible. Therefore we have in particular that every sort has a type and that this type is unique. Furthermore we have that every top sort is either ι or o or it is a composed sort $(\kappa_1 \times \cdots \times \kappa_m \to \mu)$ where μ is a top sort.

2 Definition (Term Declaration): A *term declaration* is a pair (t, κ) usually denoted as $(t : \kappa)$, where t is a well-typed term that is not a variable and κ a sort of the same type as t. If t is a constant $(t : \kappa)$ it is called a *constant declaration* and otherwise a *proper term declaration*. For every constant we require exactly one constant declaration. Variables have fixed sorts, so they cannot be declared. We denote term declarations by δ.

3 Definition (Terms of $\mathcal{L}^{\omega}_{\mathrm{sort}}$): Let Σ be an unsorted signature, $\mathcal{S}$ be a set of sorts, and $\mathfrak{s}$ be a function, which indicates the sorts of variables. Then every variable x with $\mathfrak{s}(x) = \kappa$ is a *term* of sort κ, every term t with term declaration $(t : \kappa)$ and every sort respecting instantiation of t is a *term* of sort κ, and with f is a term of sort $\kappa = (\kappa_1 \times \cdots \times \kappa_m \to \mu)$ and $t_{\kappa_1}, \ldots, t_{\kappa_m}$ are terms of the sorts indicated by their subscripts, $f_\kappa(t_{\kappa_1}, \ldots, t_{\kappa_m})$ is a *term* of sort μ. Furthermore, if t is a term of sort ν and $\nu \dot{\sqsubseteq} \mu$, then t is a *term* of sort μ.

Remark: The sort of a term needs not to be unique in general, only the top sort of a term is unique. For instance if we have a function f of sort $(\mathbb{R} \to \mathbb{N})$ with $\mathbb{N} \dot{\sqsubseteq} \mathbb{R} \dot{\sqsubseteq} \iota$, then it has also the sorts $(\mathbb{R} \to \mathbb{R})$ and $(\mathbb{R} \to \iota)$. The top sort $(\mathbb{R} \to \iota)$ of f is unique. However, a function g of sort $(\mathbb{R} \to \mathbb{N})$ has neither sort $(\mathbb{N} \to \mathbb{N})$ nor sort $(\iota \to \mathbb{N})$.

4 Definition (Signature of $\mathcal{L}^{\omega}_{\mathrm{sort}}$): A *sorted signature* $\Sigma_{\mathcal{S}} = (\Sigma, \mathcal{S}, \mathfrak{s}, \dot{\sqsubseteq}, \boldsymbol{\delta})$ of a logic in $\mathcal{L}^{\omega}_{\mathrm{sort}}$ consist of an unsorted signature Σ, which can be structured in Σ^{const}, the constants, and Σ^{var}, the variables, a set $\mathcal{S}$ of sorts, a function $\mathfrak{s} : \Sigma^{var} \to \mathcal{S}$, such that for every sort $\kappa \in \mathcal{S}$, there exist countably infinitely many variables $x \in \Sigma^{var}$ with $\mathfrak{s}(x) = \kappa$. We write variables x with $\mathfrak{s}(x) = \kappa$ in the form x_κ, a (finite) set of subsort declarations, and a set of term declarations $\boldsymbol{\delta}$.

5 Definition (Admissible Sorted Signature): A sorted signature $\Sigma_{\mathcal{S}}$ is *admissible*, iff each subterm t_i of every well-sorted term $f(t_1, \ldots, t_m)$ is also a well-sorted term. Terms in proper term declarations must be well-sorted with respect to the constant declarations. In the following we will assume that all signatures are admissible (cp. [15, p.16])

6 Definition (Formulae of $\mathcal{L}^{\omega}_{\mathrm{sort}}$): Formulae are defined inductively so that every term of type o is a *formula* and if φ and ψ are formulae and x is a variable of an arbitrary sort κ, then $(\neg\varphi)$, $(\varphi \wedge \psi)$, and $(\forall x_\kappa \varphi)$ are *formulae*. As usual $\vee$, $\Rightarrow$, $\Leftrightarrow$, and $\exists$ can be defined in terms of $\wedge$ and $\forall$.

The order of terms is defined just as in the unsorted case and $\mathcal{L}^{2n}_{\mathrm{sort}}$ is defined to be the subset of $\mathcal{L}^{\omega}_{\mathrm{sort}}$ such that every variable, every constant and every sort declaration is of order less or equal to n, $\mathcal{L}^{2n-1}$ is the subset of $\mathcal{L}^{2n}$ such that no variable of order n is quantified on. By ord(τ) we denote the order of a type.

A signature $\Sigma_{\mathcal{S}}$ is called *many-sorted* iff all subsort relations are of the form $\mu \dot{\sqsubseteq} \nu$ where ν is a top sort. Otherwise it is called *order-sorted*. We denote many-sorted logics by $\mathcal{L}^{n}_{\Lambda}$.

The Semantics

Now we define a set-theoretical semantics for the formulae.

7 Definition (Semantics): Let $\Sigma_{\mathcal{S}}$ be the sorted signature of a logic $\mathcal{L}_{\mathrm{sort}}$. A *frame* corresponding to $\Sigma_{\mathcal{S}} = (\Sigma, \mathcal{S}, \mathfrak{s}, \dot{\sqsubseteq}, \boldsymbol{\delta})$ is a collection $\{\mathcal{D}_\kappa\}_\kappa$ of nonempty sets $\mathcal{D}_\kappa$, one for each sort symbol $\kappa \in \mathcal{S}$, such that $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_{(\kappa_1 \times \cdots \times \kappa_m \to \mu)} \subseteq \mathcal{F}(\mathcal{D}_{\kappa_1}, \ldots, \mathcal{D}_{\kappa_m}; \mathcal{D}_\mu)$ for all sorts κ_i, μ as well as $\mathcal{D}_\kappa \subseteq \mathcal{D}_\mu$ for all sorts κ, μ with $\kappa \dot{\sqsubseteq} \mu$ (By $\mathcal{F}(A_1, \ldots, A_m, B)$ we denote the set of all total functions from $A_1 \times \cdots \times A_m$ to B). The members of $\mathcal{D}_o$ are called *truth values* and the members of $\mathcal{D}_\iota$ are called *individuals*.

An *interpretation* $\mathcal{J}$ in $\{\mathcal{D}_\kappa\}_\kappa$ is a function that maps each constant of sort κ of $\mathcal{L}^{\omega}_{\mathrm{sort}}(\Sigma_{\mathcal{S}})$ to an element of $\mathcal{D}_\kappa$, that is, for every constant declaration $(c : \kappa)$ we have $\mathcal{J}(c) \in \mathcal{D}_\kappa$.

An *assignment* into a frame $\{\mathcal{D}_\kappa\}_\kappa$ is a function ξ on the variables such that for all variables $\xi(x_\kappa) \in \mathcal{D}_\kappa$.

A pair $\mathcal{M} = \langle \{\mathcal{D}_\kappa\}_\kappa, \mathcal{J} \rangle$ is a *weak model* (general model) for $\mathcal{L}^{\omega}_{\mathrm{sort}}(\Sigma_{\mathcal{S}})$ with $\Sigma_{\mathcal{S}} = (\Sigma, \mathcal{S}, \mathfrak{s}, \dot{\sqsubseteq}, \boldsymbol{\delta})$ iff there is a binary function $\mathcal{V}^{\mathcal{M}}$ so that for every assignment ξ and for every term t of sort κ, $\mathcal{V}^{\mathcal{M}}_\xi(t) \in \mathcal{D}_\kappa$, that is, in particular for every term declaration $(t : \kappa)$ in $\boldsymbol{\delta}$, $\mathcal{V}^{\mathcal{M}}_\xi(t : \kappa) = (\mathcal{V}^{\mathcal{M}}_\xi(t) \in \mathcal{D}_\kappa)$ and the usual conditions of homomorphy hold, that is,

1. for all variables x_κ, $\mathcal{V}^{\mathcal{M}}_\xi(x_\kappa) = \xi(x_\kappa)$
2. for all constants c_κ, $\mathcal{V}^{\mathcal{M}}_\xi(c_\kappa) = \mathcal{J}(c_\kappa)$
3. for composed terms $\mathcal{V}^{\mathcal{M}}_\xi(f_{(\kappa_1 \times \cdots \times \kappa_m \to \mu)}(t_{\kappa_1}, \ldots, t_{\kappa_m})) = \mathcal{V}^{\mathcal{M}}_\xi(f_{(\kappa_1 \times \cdots \times \kappa_m \to \mu)})(\mathcal{V}^{\mathcal{M}}_\xi(t_{\kappa_1}), \ldots, \mathcal{V}^{\mathcal{M}}_\xi(t_{\kappa_m}))$
4. $\mathcal{V}^{\mathcal{M}}_\xi(\varphi \wedge \psi) = \mathcal{V}^{\mathcal{M}}_\xi(\varphi) \wedge \mathcal{V}^{\mathcal{M}}_\xi(\psi)$
5. $\mathcal{V}^{\mathcal{M}}_\xi(\neg\varphi) = \neg\mathcal{V}^{\mathcal{M}}_\xi(\varphi)$
6. $\mathcal{V}^{\mathcal{M}}_\xi(\forall x_\kappa \varphi) = \forall d \in \mathcal{D}_\kappa \mathcal{V}^{\mathcal{M}}_{\xi[x_\kappa \leftarrow d]}(\varphi)$

Let φ be a formula of $\mathcal{L}^n_{\text{sort}}(\Sigma_{\mathcal{S}})$, and $\mathcal{M}$ be a weak model of $\mathcal{L}^n_{\text{sort}}(\Sigma_{\mathcal{S}})$. $\mathcal{M}$ is a weak *model* of φ iff for every assignment ξ into $\mathcal{M}$, $\mathcal{V}^{\mathcal{M}}_{\xi}(\varphi) = \mathtt{T}$. We write $\mathcal{M} \models \varphi$.

Remark: In chapter 3 we will give translations of the order-sorted higher-order logics $\mathcal{L}^n_{\text{sort}}$ into the order-sorted logics $\mathcal{L}^1_{\text{sort}}$. That is, the order-sorted logics of ARNOLD OBERSCHELP [13], which have been operationalized by CHRISTOPH WALTHER [16], MANFRED SCHMIDT-SCHAUSS [15], ALAN FRISCH [5] et al., serve as target logics. Hence we can use theorem provers like the Markgraf Karl Refutation Procedure [12] for proving theorems in these logics.

Remark: Normally if we prove mathematical theorems we want to show that a formula set Γ entails a formula φ in the strong sense. Because higher-order logic cannot be completely calculized, weak semantics has been introduced by LEON HENKIN [7], which is in principle a first-order semantics. With respect to weak semantics it is possible to have sound and complete calculi. If we show $\Gamma \vdash \varphi$ then we have $\Gamma \models \varphi$ and by this we can conclude the corresponding strong model relation. Of course to every calculus there are higher-order theorems (with respect to the strong semantics) which cannot be proved at all; they cannot be theorems with respect to the weak semantics. The weak semantics given so far is rather remote from strong semantics and we can approximate it to strong semantics by so-called comprehension axioms, that is, instead of $\Gamma \models \varphi$ we try to prove $\Gamma \cup \Upsilon \models \varphi$, what still entails the corresponding strong model relation. If such axioms are necessary in order to prove a theorem, we call it *truly higher-order*, else *essentially first-order*. The comprehension axioms Υ are:

Υ^f For every term t of sort $\kappa \neq o$ of which the free variables are at most the different variables $x_1, \ldots, x_m, y_1, \ldots, y_k$ of sort $\kappa_1, \ldots, \kappa_m, \mu_1, \ldots, \mu_k$:
$\forall y_1 \ldots \forall y_k \exists f_{(\kappa_1 \times \cdots \times \kappa_m \to \kappa)} \forall x_1 \ldots \forall x_m$
$(f(x_1, \ldots, x_m) \doteq t)$

Υ^p For every formula φ of which the free variables are at most the different variables $x_1, \ldots, x_m, y_1, \ldots, y_k$ of sort $\kappa_1, \ldots, \kappa_m$, $\mu_1, \ldots, \mu_k$:
$\forall y_1 \ldots \forall y_k \exists p_{(\kappa_1 \times \cdots \times \kappa_m \to o)} \forall x_1 \ldots \forall x_m$
$(p(x_1, \ldots, x_m) \Leftrightarrow \varphi)$

In the unsorted case it is possible to represent (for a finite set of occuring types) the a priori infinitely many comprehension axioms by a finite set of axioms for the permutation, projection, and junction of arguments, for nesting and composing formulae (compare [10]). The axioms are of the following form:

PERM In this class are axioms that allow one to permute the arguments of parameters (that are function or predicate symbols) as in $g(x, y) = f(y, x)$.
Let $(\tau_1, \ldots, \tau_n)$ be given and let π be a permutation on $\{1, \ldots, n\}$, such that π permutes two neighbouring elements, that is, $\pi(i) = i + 1$ and $\pi(i+1) = i$ for $i < n$ then the following formula is in **PERM**:

$$\forall f_{\tau_1 \times \cdots \times \tau_n \to \tau'} \exists g_{\tau_{\pi(1)} \times \cdots \times \tau_{\pi(n)} \to \tau} \forall x_{\tau_1} \ldots \forall x_{\tau_n} \; g(x_{\tau_{\pi(1)}}, \ldots, x_{\tau_{\pi(n)}}) \doteq f(x_{\tau_1}, \ldots, x_{\tau_n})$$

We think that the procedure of replacing the infinite set of comprehension axioms by a finite one can be transferred to the sorted case, making the translation method appropriate for truly higher-order theorems too.

3 Translations

In this section we define the concepts of a morphism and introduce the soundness and completeness of such mappings. Furthermore we extend the criterion of the soundness of such morphisms in the unsorted case to the sorted case by applying the corresponding relativizations.

3.1 Logic Morphisms

Now we shall define those concepts that are necessary to describe the translations between formalizations in different logics.

8 Definition (Morphism of Logics): A morphism Θ is a mapping that maps the signature Σ of a logic $\mathcal{F}^1(\Sigma)$ to a signature of a logic $\mathcal{F}^2(\Theta(\Sigma))$ and that maps every formula set in $\mathcal{F}^1(\Sigma)$ to a formula set in $\mathcal{F}^2(\Theta(\Sigma))$.
Let Θ be a morphism from $\mathcal{F}^1$ to $\mathcal{F}^2$. Θ is called sound iff the following condition holds for every formula set Γ in $\mathcal{F}^1$: if Γ has a weak model in $\mathcal{F}^1$ then there is a weak model of $\Theta(\Gamma)$ in $\mathcal{F}^2$.

As expected the soundness of a morphism means: a proof that the translated problem is unsatisfiable entails that the original problem is unsatisfiable. Completeness is defined mutatis mutandis.

3.2 Translations of Higher-Order Sorted Logic

In this section we sketch how to extend the results of translating unsorted higher-order logic into first-order logic to the sorted case. The proof of the soundness theorem is reduced to the corresponding unsorted theorem by using relativizations, which can be done since the translations are structure conserving. The advantage of the proofs via relativizations is that they can be used for other kinds of sort structures too, especially they are easy to transfer to sorted logics where the semantics is defined by the relativization. At first we define relativizations and show them to be sound and complete morphisms. The completeness proof for the standard translation is not lifted, but worked out directly.

Relativizations and Partial Relativizations

For a formula set of sorted logics it is in general possible to state an equivalent formula set of an unsorted logic. In this section we will introduce relativizations and partial relativizations for the logics $\mathcal{L}^n_{\text{sort}}$ to $\mathcal{L}^n$ and $\mathcal{L}^1_{\text{sort}}$ to $\mathcal{L}^1_{\Lambda}$.

9 Definition (Relativization): The *relativization* $\Re$ from $\mathcal{L}^n_{\text{sort}}$ to $\mathcal{L}^n$ is the following morphism:

1. the signature $\Sigma_{\mathcal{S}} = (\Sigma, \mathcal{S}, \mathfrak{s}, \sqsubseteq, \boldsymbol{\delta})$ is mapped to $(\Sigma \cup \mathcal{S} \backslash \{\tau | \tau \text{ is type}\}, \{\tau | \tau \text{ is type}\}, x \mapsto \text{type}(x), \emptyset, \emptyset)$, where the sort symbols κ of type τ are viewed as unary predicate constants of type $(\tau \to o)$.

2. A formula φ is mapped to the formula set $\Re(\varphi)$ consisting of:
   $\{\widehat{\Re}(\varphi)\} \cup$
   $\{\forall x_\tau \ \ \kappa(x) \Rightarrow \mu(x) \mid$
   $\kappa \dot{\sqsubseteq} \mu$, with $\mathrm{type}(\kappa) = \mathrm{type}(\mu) = \tau\} \cup$
   $\{\mathrm{universal\_closure}(\kappa(\widehat{\Re}(t))) \mid (t : \kappa) \in \boldsymbol{\delta}\}$, where $\widehat{\Re}$ is defined as:
   (a) For terms $\widehat{\Re}(t) = t$ and for atomic formulae: $\widehat{\Re}(\varphi) = \varphi$
   (b) For conjunctions and negations: $\widehat{\Re}(\varphi \wedge \psi) = \widehat{\Re}(\varphi) \wedge \widehat{\Re}(\psi)$ and $\widehat{\Re}(\neg\varphi) = \neg\widehat{\Re}(\varphi)$.
   (c) For a quantification over a variable x of sort κ, $\mathrm{type}(\kappa) = \tau$, $\widehat{\Re}(\forall x_\kappa \varphi) = \forall x_\tau \ \kappa(x) \Rightarrow \widehat{\Re}(\varphi)$.

For formula sets Γ we have as usual $\Re(\Gamma) = \bigcup_{\varphi \in \Gamma} \Re(\varphi)$.

10 Definition (Partial Relativization): The *partial relativization* $\partial\Re$ from $\mathcal{L}^1_{\mathrm{sort}}$, where every sort (except ι) has a unique upper sort (shortly called "uus") immediately below ι (with respect to $\dot{\sqsubseteq}$), to $\mathcal{L}^1_\Lambda$ is a morphism defined analogously to a relativization, but we translate expressions of the form $\forall x_\kappa \varphi$ to $(\forall x_{\mathrm{uus}(\kappa)} \kappa(x) \Rightarrow \partial\widehat{\Re}(\varphi))$, that is, we transform an order-sorted formulation to a many-sorted one. For details see [9].

11 Theorem: *The relativizations $\Re$ form $\mathcal{L}^n_{\mathrm{sort}}$ to $\mathcal{L}^n$ are sound, complete, and injective; the partial relativizations $\partial\Re$ from $\mathcal{L}^1_{\mathrm{sort}}$ to $\mathcal{L}^1_\Lambda$ are sound and complete.*

Proof: The proof is straightforward and can be found in [9]. ■

A Sufficient Criterion for Soundness

Now we give a sufficient criterion for the soundness of translations of formulae of $\mathcal{L}^n_{\mathrm{sort}}$ onto formulae of $\mathcal{L}^1_{\mathrm{sort}}$, which is strong enough to cover most requirements.

12 Definition (Quasi-Homomorphism): Let $\mathcal{F}_1(\Sigma^1_{\mathcal{S}})$ and $\mathcal{F}_2(\Sigma^2_{\mathcal{S}})$ be two logics. A morphism Θ from $\mathcal{F}_1$ to $\mathcal{F}_2$ is called a *quasi-homomorphism* from $\mathcal{F}_1(\Sigma^1_{\mathcal{S}})$ to $\mathcal{F}_2(\Sigma^2_{\mathcal{S}})$ iff the following conditions are satisfied:

1. The sorted signature $\Sigma^1_{\mathcal{S}} = (\Sigma_1, \mathcal{S}_1, \mathfrak{s}_1, \dot{\sqsubseteq}_1, \boldsymbol{\delta}_1)$ is mapped to a signature $\Sigma^2_{\mathcal{S}} = (\Sigma_2, \mathcal{S}_2, \mathfrak{s}_2, \dot{\sqsubseteq}_2, \boldsymbol{\delta}_2)$ such that,
   1.1 $\Theta(\Sigma_1) \subseteq \Sigma_2$, variables are mapped onto variables and constants onto constants by Θ,
   1.2 $\Theta(\mathcal{S}_1) \subseteq \mathcal{S}_2$, sort symbols are mapped onto sort symbols by Θ,
   1.3 $\Theta(\mathfrak{s}_1(x)) = \Theta(\mathfrak{s}_1)(\Theta(x)) = \mathfrak{s}_2(x)$ for all variables x in Σ_1^{var},
   1.4 if $\kappa \dot{\sqsubseteq}_1 \mu$, then $\Theta(\kappa) \dot{\sqsubseteq}_2 \Theta(\mu)$, that is, $\Theta(\dot{\sqsubseteq}_1) \subseteq \dot{\sqsubseteq}_2$, and
   1.5 for every term declaration $(t : \kappa)$ in $\boldsymbol{\delta}_1$ we have that $\Theta(t)$ has sort $\Theta(\kappa)$. Especially we have for every constant c of sort κ in Σ_1 that $\Theta(c)$ has sort $\Theta(\kappa)$ in Σ_2.
2. For all composed terms: if $f(t_1, \ldots, t_m)$ is a term of $\mathcal{F}_1(\Sigma_1)$ of sort κ, then $\Theta(f(t_1, \ldots, t_m)) = \boldsymbol{\alpha}^f(\Theta(f), \Theta(t_1), \ldots, \Theta(t_m))$. The $\boldsymbol{\alpha}$ have to be chosen appropriately out of Σ_2: they have to be *new*, that is, there is no element e of $\Sigma^1_{\mathcal{S}}$ with $\boldsymbol{\alpha}^f = \Theta(e)$. ($\boldsymbol{\alpha}$ stands for *apply*.)
3. For all formulae φ_1, φ_2 and for all variables x:
   3.1 $\Theta(\varphi_1 \wedge \varphi_2) = \Theta(\varphi_1) \wedge \Theta(\varphi_2)$
   3.2 $\Theta(\neg\varphi) = \neg\Theta(\varphi)$
   3.3 $\Theta(\forall x \varphi) = \forall \Theta(x) \Theta(\varphi)$

13 Theorem: *If Θ is an injective quasi-homomorphism from $\mathcal{L}^n_{\mathrm{sort}}(\Sigma_{\mathcal{S}})$ to $\mathcal{L}^1_{\mathrm{sort}}(\Sigma'_{\mathcal{S}})$, then Θ is weakly sound.*

Proof: There is a commutative diagram:

$$\begin{tikzcd} \mathcal{L}^n_{\mathrm{sort}} \arrow[r, "\Re"] \arrow[d, "\Theta"'] \arrow[dr, phantom, "\#"] & \mathcal{L}^n \arrow[d, "\dot{\Theta}"] \\ \mathcal{L}^1_{\mathrm{sort}} \arrow[r, "\partial\Re"] & \mathcal{L}^1_\Lambda \end{tikzcd}$$

with a quasi-homomorphism $\dot{\Theta}$ from $\mathcal{L}^n$ to $\mathcal{L}^1_\Lambda$. Since $\partial\Re$ is complete, $\Re$ is sound and $\dot{\Theta}$ is sound by the corresponding unsorted theorem [8], we can conclude that Θ is sound. $\dot{\Theta}$ can be constructed out of Θ using the relation $\dot{\Theta}(x) = \partial\Re \circ \Theta \circ \Re^{-1}(x)$ on basic terms and lifting this property to arbitrary terms. ■

Remark: We can see that the translated formulae are not essentially more complex then the original ones and that proofs found in a translated problem setting $\Theta(\Gamma) \vdash \Theta(\varphi)$ can easily be translated back to a proof for $\Gamma \vdash \varphi$, because the mappings Θ are injective.

The Standard Translation From Sorted Higher-Order Logic to Sorted First-Order Logic

Now we want to define morphisms $\Phi^n_{\mathcal{S}}$ from $\mathcal{L}^n_{\mathrm{sort}}$ to $\mathcal{L}^1_{\mathrm{sort},\doteq}$ (order sorted first-order logic with equality) which are not only sound but also complete. We define the morphisms for odd n, for even n they are obtained as the restriction the next higher odd number, that is, $\Phi^{2n}_{\mathcal{S}} := \Phi^{2n+1}_{\mathcal{S}} \mid_{\mathcal{L}^{2n}_{\mathrm{sort}}}$. The morphisms Φ are defined as $\Phi(\varphi) = \Psi(\varphi) \cup \Xi^{\mathcal{S}}_\alpha$, where $\Psi(\varphi)$ is a quasi-homomorphism and $\Xi^{\mathcal{S}}_\alpha$ are special extensionality axioms for the apply functions $\boldsymbol{\alpha}$.

In the following we have to map the sorts κ injectively to new names. Therefore we use a function $\tilde{\ }$, which can be realized by taking the strings "κ". Often we abbreviate $\tilde{o}$ to o.

14 Definition (Standard Translation $\Phi^{\mathcal{S}}_{2n-1}$): Let $\Sigma^{2n-1}_{\mathcal{S}}$ be the signature of a logic in $\mathcal{L}^{2n-1}_{\mathrm{sort}}$. In order to define a morphism Φ to $\mathcal{L}^1_{\mathrm{sort},\doteq}$, we have to define the signature $\widehat{\Sigma}_{\mathcal{S}}$ of the target logic:
Let $\widehat{\Sigma}_{\mathcal{S}} := (\widehat{\Sigma}, \mathcal{S}, \mathfrak{s}, \dot{\sqsubseteq}, \boldsymbol{\delta})$

1. $\widehat{\Sigma}$ is the union of the following sets:
   (a) $\widehat{\Sigma}^{const}_\iota = \bigcup_\tau \Sigma^{const}_\tau$
   (b) $\widehat{\Sigma}^{const}_{(\underbrace{\iota \times \cdots \times \iota}_{m} \to \iota)} =$
   $\{\boldsymbol{\alpha}^{\tilde{\tau}} \mid \tau = (\tau_1 \times \cdots \times \tau_m \to \sigma), \sigma \neq o\}$

(c) $\widehat{\Sigma}^{const}_{\underbrace{(\iota\times\cdots\times\iota\to o)}_{m}} = \{\boldsymbol{\alpha}^{\tilde\tau} \mid \tau = (\tau_1\times\cdots\times\tau_m\to o)\}$ with elements $\boldsymbol{\alpha}^{\tilde\tau}$ which are new, that is, which do not occur in Σ. In addition, for $m=2$ we have in $\widehat{\Sigma}^{const}_{(\iota\times\iota\to o)}$ the equality sign $\doteq$.

(d) $\widehat{\Sigma}^{var}_{\iota} = \bigcup_{\tau} \Sigma^{var}_{\tau}$.

Φ is defined on the signature as the inclusion mapping $\Sigma_{\mathcal S} \hookrightarrow \widehat{\Sigma}_{\mathcal S}$.

2. $\mathcal S$ is defined as the set $\{\tilde\kappa \mid \mathrm{ord}(\kappa) < n\} \;\cup$ $\{(\tilde\kappa\times\tilde\kappa_1\times\cdots\times\tilde\kappa_m\to\tilde\mu) \mid \kappa = (\kappa_1\times\cdots\times\kappa_m\to\mu)\}$
3. $\mathfrak{s}$ is defined for variables $\Phi(x_\kappa)$, where x_κ is mapped to $\kappa = (\kappa_1\times\cdots\times\kappa_m\to\mu)$ by the corresponding $\mathfrak{s}$-function in higher-order logic, as $\mathfrak{s}(\Phi(x_\kappa)) = \tilde\kappa$
4. $\sqsubseteq$ is defined as
   (a) $\tilde\kappa\sqsubseteq\tilde\mu$ for all $\kappa\sqsubseteq\mu$.
   (b) $\tilde\kappa\sqsubseteq\iota$ for all top sorts κ unequal to o.
5. $\boldsymbol{\delta}$ is defined as the set of all term declarations:
   (a) $(\boldsymbol{\alpha}^{\tilde\tau}(x_{\tilde\kappa}, x_{\tilde\kappa_1},\ldots,x_{\tilde\kappa_m}) : \tilde\mu)$ for all $\kappa = (\kappa_1\times\cdots\times\kappa_m\to\mu))$ with $\mathrm{type}(\kappa)=\tau$,
   (b) for all term declaration $(t:\kappa)$ we have a term declaration $(\Phi(t):\tilde\kappa)$,
   (c) $((x_{\tilde\kappa}\doteq x_{\tilde\kappa}) : \tilde o)$ for all κ with $\mathrm{ord}(\kappa)<n$ and $\kappa\neq o$.

The morphism Ψ is a quasi-homomorphism on that behaves on the signature exactly like Φ, with the only exception that $\doteq$ is not in the image of Ψ.

$\Xi^{\mathcal S}_{\alpha}$ is the set consisting of the following formulae of $\mathcal{L}^1_{\mathcal S,\doteq}$:

$\Xi^{\mathcal S,f}_{\alpha}$ For every function constant $\boldsymbol{\alpha}^{\tilde\tau}$ with $\tau = (\tau_1\times\cdots\times\tau_m\to\sigma)$, $\sigma\neq o$ and for all sorts $\kappa = (\kappa_1\times\cdots\times\kappa_m\to\mu)$ of type τ we have:
$\forall f_{\tilde\kappa}\forall g_{\tilde\kappa}(\forall x^1_{\tilde\kappa_1},\ldots,\forall x^m_{\tilde\kappa_m}$
$\boldsymbol{\alpha}^{\tilde\tau}(f,x_1,\ldots,x_m)\doteq\boldsymbol{\alpha}^{\tilde\tau}(g,x_1,\ldots,x_m)) \Rightarrow f\doteq g$

$\Xi^{\mathcal S,p}_{\alpha}$ For every predicate constant $\boldsymbol{\alpha}^{\tilde\tau}$ with $\tau = (\tau_1\times\cdots\times\tau_m\to o)$ and for all sorts $\kappa = (\kappa_1\times\cdots\times\kappa_m\to o)$ of type τ we have:
$\forall p_{\tilde\kappa}\forall q_{\tilde\kappa}(\forall x^1_{\tilde\kappa_1},\ldots,\forall x^m_{\tilde\kappa_m}$
$\boldsymbol{\alpha}^{\tilde\tau}(p,x_1,\ldots,x_m)\Leftrightarrow\boldsymbol{\alpha}^{\tilde\tau}(q,x_1,\ldots,x_m)) \Rightarrow p\doteq q$

We define $\Phi(\varphi) = \Psi(\varphi)\cup\Xi^{\mathcal S}_{\alpha}$. Analogously for formula sets $\Phi(\Gamma) = \Psi(\Gamma)\cup\Xi^{\mathcal S}_{\alpha}$.

Remark: It is easy to see that Ψ_{2n-1} is an injective quasi-homomorphism from $\mathcal{L}^{2n-1}_{\mathrm{sort}}(\Sigma)$ to $\mathcal{L}^{1}_{\mathrm{sort}}(\Psi(\Sigma))$. Therefore it is not to difficult to prove the soundness of Φ.

15 Theorem: *Φ is weakly complete.*

Proof: In the *first* essential step we define a frame with the help of which we will define a model for Γ. This frame is defined inductively with the induction base $\check{\mathcal D}_\kappa := \mathcal{D}_{\tilde\kappa}$ for all κ of type ι. For top sorts $\kappa = (\kappa_1\times\cdots\times\kappa_m\to\mu)$ we define $\check{\mathcal D}_\kappa$ as subsets of $\mathcal{F}(\check{\mathcal D}_{\kappa_1},\ldots,\check{\mathcal D}_{\kappa_m};\check{\mathcal D}_\mu)$. We cannot take the whole set, because then we would try to obtain strong completeness, which cannot be achieved in general. In a *second* step we define an interpretation function $\check{\mathcal J}$ for $\mathcal{L}^n_{\mathrm{sort}}$ and show that the inclusion relations induced by the subsort relations hold. In a *third* step we show by induction on the construction of terms and formulae that the quasi-homomorphism Ψ is compatible with the model relation. Formally we show $\natural\circ\mathcal{V}^{\mathcal M}_{\xi}\circ\Psi = \mathcal{V}^{\check{\mathcal M}}_{\check\xi}$. In a *fourth* and last step we use this property to show that $\check{\mathcal M} = \langle\{\check{\mathcal D}_\kappa\}_\kappa, \check{\mathcal J}\rangle$ is a model of Γ.

Let Γ be a formula set in $\mathcal{L}^{2n-1}_{\mathrm{sort}}(\Sigma_{\mathcal S})$. Let $\mathcal M$ be a weak model of $\Phi(\Gamma)$. Then $\mathcal M$ is a model of $\Phi(\varphi)$ for every formula φ in Γ. Let $\mathcal M$ be $\langle\{\mathcal D_\kappa\}_\kappa, \mathcal J\rangle$ and ξ be an arbitrary assignment. Then we have $\mathcal{V}^{\mathcal M}_{\xi}(\Phi(\varphi)) = \mathtt{T}$. We want to construct a model $\check{\mathcal M}$ of φ, so that for all assignments $\check\xi$ we have $\mathcal{V}^{\check{\mathcal M}}_{\check\xi}(\varphi) = \mathtt{T}$.

Step 1: In this step we define a frame for $\mathcal{L}^{2n-1}_{\mathrm{sort}}(\Sigma_{\mathcal S})$. Therefore we define $\check{\mathcal D}_\iota := \mathcal D_{\tilde\iota}$ and $\check{\mathcal D}_o = \mathcal D_{\tilde o} = \{\mathtt{T},\mathtt{F}\}$. For all other top sorts κ with $\kappa = (\kappa_1\times\cdots\times\kappa_m\to\mu)$ we have to define $\check{\mathcal D}_\kappa \subseteq \mathcal{F}(\check{\mathcal D}_{\kappa_1},\ldots,\check{\mathcal D}_{\kappa_m};\check{\mathcal D}_\mu)$. We do this by inductively defining injective functions $\natural_\kappa$ from $\mathcal D_{\tilde\kappa}$ to $\mathcal{F}(\check{\mathcal D}_{\kappa_1},\ldots,\check{\mathcal D}_{\kappa_m};\check{\mathcal D}_\mu)$ and setting $\check{\mathcal D}_\kappa := \natural_\kappa(\mathcal D_{\tilde\kappa})$. Hence $\natural_\kappa$ is a bijective function from $\mathcal D_{\tilde\kappa}$ to $\check{\mathcal D}_\kappa$. For the other sorts κ with top sort μ we define: $\check{\mathcal D}_\kappa := \natural_\mu(\mathcal D_{\tilde\kappa})$.

We define $\natural_\kappa$ as bijective functions inductively:

1. $\natural_\iota : \mathcal D_{\tilde\iota}\to\check{\mathcal D}_\iota$ and $\natural_o : \mathcal D_{\tilde o}\to\check{\mathcal D}_o$ as the identity mappings (These functions are obviously bijective).
2. Let $\natural_{\kappa_i}$ and $\natural_\mu$ be already defined. We are going to define a function $\natural_\kappa$ with $\kappa = (\kappa_1\times\cdots\times\kappa_m\to\mu)$, $\mu\neq o$, for $\mathcal D_{\tilde\kappa}$. For all $d\in\mathcal D_{\tilde\kappa}$ let $\natural_\kappa(d)$ be defined by $\natural_\kappa(d)(\check d_1,\ldots,\check d_m) := \natural_\mu(\mathcal{V}^{\mathcal M}_{\xi}(\boldsymbol{\alpha}^{\tilde\tau})(d,\natural^{-1}_{\kappa_1}(\check d_1),\ldots,\natural^{-1}_{\kappa_m}(\check d_m)))$ for all $\check d_1\in\check{\mathcal D}_{\kappa_1},\ldots,\check d_m\in\check{\mathcal D}_{\kappa_m}$
   The following diagram may help to see the involved mappings at a glance:
$$\begin{tikzcd}
\mathcal{V}^{\mathcal M}_{\xi}(\boldsymbol{\alpha}^{\tilde\tau}) : \mathcal D_{\tilde\kappa} \arrow[d, "!\,\natural_\kappa"] & \times & \mathcal D_{\tilde\kappa_1}\times\cdots\times & \mathcal D_{\tilde\kappa_m} \arrow[r] & \mathcal D_{\tilde\mu} \arrow[d, "\natural_\mu"] \\
\check{\mathcal D}_\kappa \hookrightarrow \mathcal{F}(& & \check{\mathcal D}_{\kappa_1}, \ldots, \arrow[u, "\natural^{-1}_{\kappa_1}"] & \check{\mathcal D}_{\kappa_m} ; \arrow[u, "\natural^{-1}_{\kappa_m}"] & \check{\mathcal D}_\mu)
\end{tikzcd}$$
   In order to show the injectivity of $\natural_\kappa$ we use that we have in $\Xi^{\mathcal S,f}_{\alpha}$ the formula
   $\forall f_{\tilde\kappa}\forall g_{\tilde\kappa}(\forall x^1_{\tilde\kappa_1},\ldots,\forall x^m_{\tilde\kappa_m}$
   $\boldsymbol{\alpha}^{\tilde\tau}(f,x^1,\ldots,x^m)\doteq\boldsymbol{\alpha}^{\tilde\tau}(g,x^1,\ldots,x^m)) \Rightarrow$
   $f\doteq g$
   Therefore we have in a model that $\mathcal{V}^{\mathcal M}_{\xi}(\boldsymbol{\alpha}^{\tilde\tau})$ is injective in the first argument, together with the bijectivity of the mappings $\natural_{\kappa_i}$ and $\natural_\mu$ the injectivity can easily be seen. Since the surjectivity is given by definition, we have proved that $\natural_\kappa$ is bijective.
3. The remaining two cases can be proved analogously.

Hence we have defined $\check{\mathcal D}_\kappa$ for all top sorts κ. For all other sorts κ with top sort μ we define $\check{\mathcal D}_\kappa := \natural_\mu(\mathcal D_{\tilde\kappa})$. Thereby we complete our frame $\{\check{\mathcal D}_\kappa\}_\kappa$.

Step 2: In this step we define an interpretation $\check{\mathcal J}$ in order to complete the definition of a model $\langle\{\check{\mathcal D}_\kappa\}_\kappa, \check{\mathcal J}\rangle$. We define $\check{\mathcal J} := \natural\circ\mathcal J\circ\Psi$ on the constants. For all subsort relations $\kappa\sqsubseteq\mu$ (with same top sort ν) we have $\check{\mathcal D}_\kappa\subseteq\check{\mathcal D}_\mu$, because we have $\tilde\kappa\sqsubseteq\tilde\mu$, hence $\mathcal D_{\tilde\kappa}\subseteq\mathcal D_{\tilde\mu}$, consequently $\check{\mathcal D}_\kappa = \natural_\nu(\mathcal D_{\tilde\kappa})\subseteq\natural_\nu(\mathcal D_{\tilde\mu}) = \check{\mathcal D}_\mu$.

Step 3: In this step we have to show that for every assignment $\check{\xi}$ in $\check{\mathcal{M}}$ there is an assignment ξ in $\mathcal{M}$, so that on all terms (and hence all formulae) t we have: $\mathcal{V}_{\check{\xi}}^{\check{\mathcal{M}}} = \natural \circ \mathcal{V}_{\xi}^{\mathcal{M}} \circ \Psi$. The proof is straightforward by induction on the construction of formulae.

<u>Step 4</u>: We notice that the term declarations are correctly interpreted, because the corresponding term declarations hold in the translated case.

Now we are going to show that if $\mathcal{M}$ is a model of $\Phi(\varphi)$, then $\check{M}$ is a model of φ. If $\mathcal{M}$ is model of $\Phi(\varphi)$, then $\mathcal{M}$ is a model of $\Psi(\varphi)$. Let $\check{\xi}$ be an arbitrary assignment and ξ be defined as $\natural^{-1} \circ \check{\xi} \circ \Psi^{-1}$, then we have $\mathcal{V}_{\xi}^{\mathcal{M}}(\Psi(\varphi)) = \mathtt{T}$, because $\mathcal{M}$ is a model of $\Psi(\varphi)$. Hence we have $\mathcal{V}_{\check{\xi}}^{\check{\mathcal{M}}}(\varphi) = \natural(\mathcal{V}_{\xi}^{\mathcal{M}}(\Psi(\varphi))) = \mathtt{T}$. Recall that for truth values $\natural$ is the identity function. ■

Remark: Ψ^{-1} provides a calculus for $\mathcal{L}^n_{\text{sort}}$. If we add rules that enforce that function symbols and predicate symbols are equal if they agree in all arguments, we can transform every sound and complete first-order calculus of $\mathcal{L}^1_{\text{sort},\doteq}$ by Φ to a sound and weakly complete calculus for $\mathcal{L}^n_{\text{sort}}$. We can execute the proof in $\mathcal{L}^1_{\text{sort},\doteq}$ and then lift it to a proof in $\mathcal{L}^n_{\text{sort}}$.

4 Summary

The main goal of this paper was to use the power of an existing order-sorted first-order theorem provers like the MKRP system for proving mathematical theorems. Therefore we presented a whole class of translations from higher-order sorted into first-order sorted logic, which are sound (compare theorem 13). As stated these translations are bidirectional, that is, we can map the first-order proofs back to higher-order logic. In theorem 15 we showed that a particular translation is not only sound, but also complete with respect to a weak semantics. In consequence we can prove *in principle* everything that is provable in higher-order sorted logic via translations into first-order sorted logic. The main drawback is however the need for the so-called comprehension axioms for *truly higher-order theorems*. The distinction between "truly higher-order theorems" and "essentially first-order theorems" made explicit the difference between theorems, which are difficult, *because* they are higher-order, and theorems, which are formulated in a higher-order syntax, but are essentially first-order. Of course, the comprehension axioms can be translated too and indeed they can even be represented by finitely many formulae by adapting methods from [6, 2, 14]. For the unsorted case we have worked it out [10], we think that the result can be tranferred to sorted case too. In tests we have compared the behaviour of the MKRP system using a sorted formulation of a theorem with the corresponding relativized one. The sorted formulation is significantly better.

References

[1] Peter B. Andrews, Sunil Issar, Dan Nesmith, and Frank Pfenning. The TPS theorem proving system. In Mark E. Stickel, editor, *Proceedings of the 10th CADE*, pages 641–642, Kaiserslautern, Germany, 1990. Springer Verlag, Berlin, Germany. LNAI 449.

[2] Robert Boyer, Ewing Lusk, William McCune, Ross Overbeek, Mark Stickel, and Lawrence Wos. Set theory in first-order logic: Clauses for Gödel's axioms. *Journal of Automated Reasoning*, **2**:287–327, 1986.

[3] Alonzo Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, **5**:56–68, 1940.

[4] R.L. Constable, S.F. Allen, H.M. Bromley, W.R. Cleaveland, J.F. Cremer, R.W. Harper, D.J. Howe, T.B. Knoblock, N.P. Mendler, P. Panangaden, J.T. Sasaki, and S.F. Smith. *Implementing Mathematics with the Nuprl Proof Development System.* Prentice Hall, Englewood Cliffs, New Jersey, USA, 1986.

[5] Alan M. Frisch. The substitutional framework for sorted deduction: Fundamental results on hybrid reasoning. *Artificial Intelligence*, **49**:161–198, 1991.

[6] Kurt Gödel. *The Consistency of the Axiom of Choice and of the Generalized Continuum-Hypothesis with the Axioms of Set Theory*, volume 3 of *Annals of Mathematics Studies.* Princeton University Press, Princeton, New Jersey; eighth printing 1970, 1940.

[7] Leon Henkin. Completeness in the theory of types. *Journal of Symbolic Logic*, **15**:81–91, 1950.

[8] Manfred Kerber. How to prove higher order theorems in first order logic. In John Mylopoulos and Ray Reiter, editors, *Proceedings of the 12th IJCAI*, pages 137–142, Sydney, 1991. Morgan Kaufman, San Mateo, California, USA.

[9] Manfred Kerber. *On the Representation of Mathematical Concepts and their Translation into First Order Logic.* PhD thesis, Fachbereich Informatik, Universität Kaiserslautern, Kaiserslautern, Germany, 1992.

[10] Manfred Kerber. On the translation of higher-order problems into first-order logic. In Tony Cohn, editor, *Proceedings of ECAI-94*, Amsterdam 1994. forthcoming.

[11] Michael Kohlhase. Unification in order-sorted type theory. In A. Voronkov, editor, *Proceedings of LPAR*, pages 421–432, Berlin, Germany, 1992. Springer Verlag. LNAI 624.

[12] Karl Mark G Raph. The Markgraf Karl Refutation Procedure. Technical Report Memo-SEKI-MK-84-01, Fachbereich Informatik, Universität Kaiserslautern, Kaiserslautern, Germany, January 1984.

[13] Arnold Oberschelp. Untersuchungen zur mehrsortigen Quantorenlogik. *Mathematische Annalen*, **145**:297–333, 1962.

[14] Art Quaife. Automated deduction in von Neumann-Bernays-Gödel set theory. *Journal of Automated Reasoning*, **8**(1):91–146, 1992.

[15] Manfred Schmidt-Schauß. *Computational Aspects of an Order-Sorted Logic with Term Declarations.* LNAI 395. Springer Verlag, Berlin, Germany, 1989.

[16] Christoph Walther. Ein mehrsortiger Resolutionskalkül mit Paramodulation. Interner Bericht 35/82, Fakultät für Informatik, Universität Karlsruhe, Karlsruhe, Germany, 1982.