

SORT LOGIC AND FOUNDATIONS OF MATHEMATICS

Jouko Väänänen

*Department of Mathematics and Statistics
University of Helsinki, Finland
Institute for Logic, Language and Computation
University of Amsterdam, The Netherlands
jouko.vaananen@helsinki.fi*

I have argued elsewhere [8] that second order logic provides a foundation for mathematics much in the same way as set theory does, despite the fact that the former is second order and the latter first order, but second order logic is marred by reliance on ad hoc *large domain assumptions*. In this chapter I argue that sort logic, a powerful extension of second order logic, provides a foundation for mathematics without any ad hoc large domain assumptions. The large domain assumptions are replaced by ZFC-like axioms. Despite this resemblance to set theory sort logic retains the structuralist approach to mathematics characteristic of second order logic. As a model-theoretic logic sort logic is the strongest logic. In fact, every model class definable in set theory is the class of models of a sentence of sort logic. Because of its strength sort logic can be used to formulate particularly strong reflection principles in set theory.

1. Introduction

Sort logic, introduced in [7], is a many-sorted extension of second order logic. In an exact sense it is the strongest logic that there is. In this paper sort logic is suggested as a foundation of mathematics and contrasted to second order logic and to set theory. It is argued that sort logic solves the problem of second order logic that existence proofs of structures rely on ad hoc large domain assumptions.

The new feature in sort logic over and above what first and second order logics have is the ability to “look outside” the model, as for a group to be the multiplicative group of a field requires reference to a zero element outside the group, or for a Turing machine, defined as a finite set of quadruples, to

halt requires reference to a tape potentially much bigger than the Turing machine itself.

In computer science it is commonplace to regard a database as a many-sorted structure. Each column (attribute) of the database has its own range of values, be it a salary figure, gender, department, last name, zip code, or whatever. In fact, it would seem very unnatural to lump all these together into one domain which has a mixture of numbers, words, and strings of symbols. To state that a new column can be added to a database, e.g. a salary column, involves stating that new elements, namely the salary values, can be added to the overall set of objects referred to in the database.

In a sense ordinary second order logic also “looks outside” the model as well as one can think of the bound second order variables as first order variables ranging over the domain of all subsets and relations on the original domain. In fact, one of the best ways to understand second order logic is to think of it as a two-sorted first order logic in which one sort—the sort over which the second order variables range—is assumed to consist of *all* subsets and relations of the other sort. When “all subsets and relations” is replaced by “enough subsets and relations to satisfy the Comprehension Axioms”, we get semantics relative to which there is a Completeness Theorem of Henkin [2]. The same is true of sort logic.

To get a feeling of sort logic, let us consider the following formulation of the field axioms in a many-sorted first order logic with two sorts of variables. We use variables x, y and z for the sort of the multiplicative group, and u, v and w for the sort of the additive group. The function \cdot and the constant 1 are of the first sort and the function $+$ and the constant 0 of the second sort:

$$\varphi = \begin{cases} \forall x \forall y \forall z ((x \cdot y) \cdot z = x \cdot (y \cdot z)) \\ \forall x (x \cdot 1 = 1 \cdot x = x) \\ \forall x \forall y (x \cdot y = y \cdot x) \\ \forall x \exists y (x \cdot y = 1) \end{cases} \quad \psi = \begin{cases} \forall x \forall y \forall z ((x + y) + z = x + (y + z)) \\ \forall x (x + 0 = 0 + x = x) \\ \forall x \forall y (x + y = y + x) \\ \forall x \exists y (x + y = 0) \\ \forall x \forall y \forall z (x \cdot (y + z) = x \cdot y + x \cdot z) \\ \forall x \exists u (x = u) \wedge \forall u \exists x (u = 0 \vee u = x) . \end{cases} \quad (1.1)$$

We have separated the multiplicative group into the first sort and the ad-

ditive group in the second sort. With this separation of the group and the bigger field part we can ask questions such as:

What kind of groups are the multiplicative group of a field?

And the answer is: exactly the groups that satisfy

$$\text{For some } + \text{ and for some } 0: \varphi \wedge \psi. \quad (1.2)$$

The truth of the sentence (1.2) in a given group means that there is something out there outside the group, in this case the element 0, which together with the new function + defines a field.

For a different type of example, suppose

$$\varphi \quad (1.3)$$

is a finite second order axiomatization of some mathematical structure in the vocabulary $\{R_1, \dots, R_n\}$. Suppose we want to say that φ has a model. So let us take a new unary predicate P and consider the sentence

$$\exists P(\exists R_1 \dots \exists R_n \varphi)^{(P)}, \quad (1.4)$$

where $\varphi^{(P)}$ means the relativization of φ to the unary predicate P . What (1.4) says in a model is that there are a subset P and relations R_1, \dots, R_n on P such that $(P, R_1, \dots, R_n) \models \varphi$. So in any model which is big enough to include a model of φ the sentence (1.4) says that there indeed is such a model. But in smaller models (1.4) is simply false, even though φ may have models. So (1.4) does not really express the existence of a model for φ . The situation would be different if we allowed “ $\exists P \exists R_1 \dots \exists R_n$ ” to refer to outside the model. In sort logic, which we will introduce in detail below, the meaning of the sentence

$$\tilde{\exists} R_1 \dots \tilde{\exists} R_n \varphi, \quad (1.5)$$

is that there is a *new* domain of objects with new relations R_1, \dots, R_n such that φ holds. Thus (1.5) expresses the semantic consistency of φ independently of the model where it is considered.

In algebra concepts such as a module P being projective, a group F being free, etc, are defined by reference to arbitrary modules M and arbitrary groups G with no concern as to whether such modules M can be realized inside P , or whether such groups G could be realized inside F . Even if it turned out that they could be so realized, the original concepts certainly referred to quite arbitrary objects N and G in the universe of all mathe-

mathematical objects. Lesson: Apparently second order concepts in mathematics sometimes refer to outside the structure being considered.

Reference outside is, of course, most blatant in set theory where objects are defined by reference to the entire universe of sets. In practice one can in most cases limit the reference to some smaller part of the universe, but very often not to the elements or to the power-set of the object being defined.

2. Sort Logic

Many-sorted logic has several domains, and variables for each domain, much like vector spaces have a scalar-domain and a vector-domain and different variables for each, or as geometry has different variables for points and lines. It seems to have been first considered by Herbrand, and later by Schmidt, Feferman [1], and others.

2.1. Basic concepts

A (*many-sorted*) *vocabulary* is any set L of predicate symbols P, Q, R, \dots . We leave function and constant symbols out for simplicity of presentation. We use natural numbers as names for sorts.

Each vocabulary L has an *arity-function*

$$\mathbf{a}_L : L \rightarrow \mathbb{N}$$

which tells the arity of each predicate symbol, and a *sort-function*

$$\mathbf{s}_L : L \rightarrow \bigcup_n \mathbb{N}^n, \mathbf{s}_L(R) \in \mathbb{N}^{\mathbf{a}_L(R)},$$

which tells what are the sorts of the elements of the tuple in a relation. Thus if $P \in L$, then P is an $\mathbf{a}_L(P)$ -ary predicate symbol for a relation of $\mathbf{a}_L(P)$ -tuples of elements of sorts n_1, \dots, n_k , where $(n_1, \dots, n_k) = \mathbf{s}(P)$. So we can read off from every n -ary predicate symbol what the sorts of the elements are in the n -tuples of the intended relation. In other words, we do not have symbols for abstract relations between elements of arbitrary sorts (except identity $=$).

2.2. Syntax

The syntax of sort logic is very close to the syntax of second order logic. In effect we just add a new form of formula $\exists P\varphi$ with the intuitive meaning that there is a predicate P of *new* sorts of elements such that φ .

Suppose L is a vocabulary. Variable symbols for individuals are x, y, z, \dots with indexes x_0, x_1, \dots when necessary, and for relations X, Y, Z, \dots with indexes X_0, X_1, \dots . Each individual variable x has a sort $\mathfrak{s}(x) \in \mathbb{N}$ associated to it, so it is a variable for elements of sort $\mathfrak{s}(x)$. Each relation variable X has an arity $\mathfrak{a}(X)$ and a sort $\mathfrak{s}(X) \in \mathbb{N}^{\mathfrak{a}(X)}$ associated to it, so it is a relation variable for a relation between elements of the sorts n_1, \dots, n_k , where $\mathfrak{s}(X) = (n_1, \dots, n_k)$.

The *logical symbols* of sort logic of the vocabulary L are $\approx, \neg, \wedge, \vee, \forall, \exists, (,), x, y, z, \dots, X, Y, Z, \dots$. *L-equations* are of the form $x = y$ where x and y can be variables of any sorts. *L-atomic formulas* are either *L-equations* or of the form $Rx_1 \dots x_k$, where $R \in L$, $\mathfrak{s}_L(R) = (n_1, \dots, n_k)$, and x_1, \dots, x_k are individual variables such that $\mathfrak{s}(x_i) = n_i$ for $i = 1, \dots, k$. A *basic formula* is an atomic formula or the negation of an atomic formula. *L-formulas* are of the form:

- (1) $x = y$.
- (2) $R(x_1, \dots, x_n)$, when $\mathfrak{s}_L(R) = (\mathfrak{s}(x_1), \dots, \mathfrak{s}(x_n))$.
- (3) $X(x_1, \dots, x_n)$, when $\mathfrak{s}(X) = (\mathfrak{s}(x_1), \dots, \mathfrak{s}(x_n))$.
- (4) $\neg\varphi$.
- (5) $(\varphi \vee \psi)$.
- (6) $\exists x\varphi$.
- (7) $\exists X\varphi$.
- (8) $\exists X\varphi$. **New Sort Condition:** If $\mathfrak{s}(X) = (n_1, \dots, n_k)$, then φ has no free variables or symbols of L , other than X , of a sort n_i or of the sort (m_1, \dots, m_l) with $\{m_1, \dots, m_l\} \cap \{n_1, \dots, n_k\} \neq \emptyset$.

The reason for the New Sort Condition is that the domains of the elements referred to by the free variables of φ are fixed already so they should not be altered by the \exists -quantifier.

We treat $\varphi \wedge \psi$, $\varphi \rightarrow \psi$, $\forall x\varphi$ and $\forall X\varphi$ as shorthands obtained from disjunction and existential quantification by means of negation.

The concept of a free occurrence of a variable in a formula is defined as in first order logic. As a new concept we have the concept of a *free occurrence of a sort* in a formula. We define it as follows, following the intuition that if a sort occurs “free” in a formula, either as the sort of an individual variable, relation variable or predicate symbol, then to understand the meaning of the formula in a model we have to fix the domain of elements of that sort. Respectively, if a sort has only “bound” occurrences in a formula, we can understand the meaning of the formula in a model without fixing the domain of elements of that sort, rather, while evaluating the meaning of

the formula in a model we most likely try different domains of elements of that sort.

The *free sorts* $\mathfrak{fs}(\varphi)$ of a formula are defined as follows:

- (1) $\mathfrak{fs}(x = y) = \{\mathfrak{s}(x), \mathfrak{s}(y)\}$.
- (2) $\mathfrak{fs}(Rx_1 \dots x_n) = \{\mathfrak{s}(x_1), \dots, \mathfrak{s}(x_n)\}$.
- (3) $\mathfrak{fs}(Xx_1 \dots x_n) = \{\mathfrak{s}(x_1), \dots, \mathfrak{s}(x_n)\}$.
- (4) $\mathfrak{fs}(\neg\varphi) = \mathfrak{fs}(\varphi)$.
- (5) $\mathfrak{fs}(\varphi \vee \psi) = \mathfrak{fs}(\varphi) \cup \mathfrak{fs}(\psi)$.
- (6) $\mathfrak{fs}(\exists x\varphi) = \mathfrak{fs}(\varphi) \cup \{\mathfrak{s}(x)\}$.
- (7) $\mathfrak{fs}(\exists X\varphi) = \mathfrak{fs}(\varphi) \cup \{n_1, \dots, n_k\}$, if $\mathfrak{s}(X) = (n_1, \dots, n_k)$.
- (8) $\mathfrak{fs}(\tilde{\exists}X\varphi) = \mathfrak{fs}(\varphi) \setminus \{n_1, \dots, n_k\}$, if $\mathfrak{s}(X) = (n_1, \dots, n_k)$.

2.3. Axioms

Below $\varphi(y/x)$ means the formula obtained from φ by replacing x by y in its free occurrences. Substitution should respect sort.

Definition 2.1: The axioms of sort logic are as follows:

Logical axioms:

- Tautologies of propositional logic.
- Identity axioms: $x = y, x = y \rightarrow y = x, (x_1 = y_1 \wedge \dots \wedge x_n = y_n \wedge \varphi) \rightarrow \varphi(y_1 \dots y_n / x_1 \dots x_n)$, for atomic φ
- Quantifier axioms:
 - $\varphi(y/x) \rightarrow \exists x\varphi$, if y is free for x in φ in the usual sense.
 - $\varphi(Y/X) \rightarrow \exists X\varphi$, if Y is free for X in φ in the usual sense.
 - $\varphi(Y/X) \rightarrow \tilde{\exists}X\varphi$, if Y is free for X in φ in the usual sense.

The rules of proof:

- Modus Ponens $\{\varphi, \varphi \rightarrow \psi\} \models \psi$
- Generalization
 - $\{\Sigma, \varphi \rightarrow \psi\} \models \exists x\varphi \rightarrow \psi$, if x is not free in $\Sigma \cup \{\psi\}$
 - $\{\Sigma, \varphi \rightarrow \psi\} \models \exists X\varphi \rightarrow \psi$, if X is not free in $\Sigma \cup \{\psi\}$
 - $\{\Sigma, \varphi \rightarrow \psi\} \models \tilde{\exists}X\varphi \rightarrow \psi$, if no free sorts of ψ occur in $\mathfrak{s}(X)$.

First Comprehension Axiom:

$$\exists X \forall y_1 \dots \forall y_m (X y_1 \dots y_m \leftrightarrow \psi)$$

for any formula ψ not containing X free, whenever $\mathfrak{s}(X) = (\mathfrak{s}(y_1), \dots, \mathfrak{s}(y_m))$.

Second Comprehension Axiom:

$$\tilde{\exists}X\forall y_1\dots\forall y_m(Xy_1\dots y_m \leftrightarrow \psi)$$

for any formula ψ not containing X free, whenever $\mathfrak{s}(X) = (\mathfrak{s}(y_1), \dots, \mathfrak{s}(y_m))$.

The logical axioms and the rules of proof are clearly indispensable and are directly derived from corresponding axioms and rules of first order logic. The difference between the axioms $\varphi(Y/X) \rightarrow \exists X\varphi$ and $\varphi(Y/X) \rightarrow \tilde{\exists}X\varphi$ is the following: Both take $\varphi(Y/X)$ as a hypothesis. The conclusion $\exists X\varphi$ says of the current sorts that a relation X satisfying φ exists, namely, Y . If $\mathfrak{s}(X) = (n_1, \dots, n_k)$, then the conclusion $\tilde{\exists}X\varphi$ says of the sorts other than n_1, \dots, n_k that domains for the sorts n_1, \dots, n_k exists so that in the combined structure of the old and new domains a relation X satisfying φ exists, namely, Y . The Comprehension Axiom is the traditional (impredicative) axiom schema which gives second order logic, and in our case sort logic, the necessary power to do mathematics [3]. In individual cases less comprehension may be sufficient but this is the general schema. The difference between the First and the Second Comprehension Axioms is that the former stipulates the existence of a relation X defined by ψ in the structure consisting of the existing sorts, while the latter says that this is even true if the sorts of elements and relations that ψ

If we limit ourselves to just one sort, for example 0, we get exactly the classical second order logic.

2.4. Semantics

We now define the semantics of sort logic. This is very much like the semantics of second order logic, except that we have to take care of the new domains that may arise from interpreting quantifiers of the form $\tilde{\exists}$ and $\tilde{\forall}$.

Definition 2.2: An *L-structure* (or *L-model*) is a function \mathcal{M} defined on L with the following properties:

- (1) If $R \in L$ and $\mathfrak{s}(R) = (n_1, \dots, n_k)$ then $n_i \in \text{dom}(\mathcal{M})$ and $M_{n_i} =_{df} \mathcal{M}(n_i)$ is a non-empty set for each $i \in \{1, \dots, k\}$.
- (2) If $R \in L$ is a k -relation symbol and $\mathfrak{s}(R) = (n_1, \dots, n_k)$, then $\mathcal{M}(R) \subseteq M_{n_1} \times \dots \times M_{n_k}$.

We usually shorten $\mathcal{M}(R)$ to $R^{\mathcal{M}}$. If no confusion arises, we use the notation

$$\mathcal{M} = (M_{n_1}, \dots, M_{n_l}; R_1^{\mathcal{M}}, \dots, R_m^{\mathcal{M}})$$

for a many-sorted structure with universes M_{n_1}, \dots, M_{n_l} and relations $R_1^{\mathcal{M}}, \dots, R_m^{\mathcal{M}}$ between elements of some of the universes. A vector space with scalar field F and vector group V would be denoted according to this convention (taking functions and constant relationally):

$$(V, F; \cdot, 1, +, 0).$$

Definition 2.3: An *assignment* into an L -structure \mathcal{M} is any function s the domain of which is a set of individual variables, relation variables and natural numbers such that

- (1) If $x \in \text{dom}(s)$, then $\mathfrak{s}(x) \in \text{dom}(\mathcal{M})$ and $s(x) \in M_{\mathfrak{s}(x)}$.
- (2) If $X \in \text{dom}(s)$ with $\mathfrak{s}_L(X) = (n_1, \dots, n_k)$, then $n_1, \dots, n_k \in \text{dom}(\mathcal{M})$ and $s(X) \subseteq M_{n_1} \times \dots \times M_{n_k}$.

A *modified assignment* is defined as follows:

$$s[a/x](y) = \begin{cases} a & \text{if } y = x \\ s(y) & \text{otherwise.} \end{cases}$$

$$s[A/X](Y) = \begin{cases} A & \text{if } Y = X \\ s(Y) & \text{otherwise.} \end{cases}$$

Suppose $\mathfrak{s}(X) = (n_1, \dots, n_k)$. A model \mathcal{M}' is an X -*expansion* of a model \mathcal{M} if $\{n_1, \dots, n_k\} \cap \text{dom}(\mathcal{M}) = \emptyset$, $\text{dom}(\mathcal{M}') = \text{dom}(\mathcal{M}) \cup \{n_1, \dots, n_k\}$, and $\mathcal{M}' \upharpoonright \text{dom}(\mathcal{M}) = \mathcal{M}$.

Definition 2.4: The *truth* of L -formulas in \mathcal{M} under s is defined as follows:

- (1) $\mathcal{M} \models_s R(x_1, \dots, x_n)$ if and only if $(s(x_1), \dots, s(x_n)) \in \mathcal{M}(R)$,
- (2) $\mathcal{M} \models_s x = y$ if and only if $s(x) = s(y)$,
- (3) $\mathcal{M} \models_s \neg\varphi$ if and only if $\mathcal{M} \not\models_s \varphi$,
- (4) $\mathcal{M} \models_s (\varphi \vee \psi)$ if and only if $\mathcal{M} \models_s \varphi$ or $\mathcal{M} \models_s \psi$,
- (5) $\mathcal{M} \models_s \exists x\varphi$ if and only if $\mathcal{M} \models_{s[a/x]} \varphi$ for some $a \in M_{\mathfrak{s}(x)}$,
- (6) $\mathcal{M} \models_s \exists X\varphi$ if and only if $\mathcal{M} \models_{s[A/X]} \varphi$ for some $A \subseteq M_{n_1} \times \dots \times M_{n_k}$, where $\mathfrak{s}(X) = (n_1, \dots, n_k)$,
- (7) $\mathcal{M} \models_s \exists\tilde{X}\varphi$ if and only if $\mathcal{M}' \models_{s[A/X]} \varphi$ for some X -expansion \mathcal{M}' of \mathcal{M} and some $A \subseteq M'_{n_1} \times \dots \times M'_{n_k}$, where $\mathfrak{s}(X) = (n_1, \dots, n_k)$.

Since (7) of the above truth definition involves unbounded quantifiers over sets, the definition has to be given separately for formulas of quantifier-rank at most a fixed natural number n . When n increases, the definition itself gets more complicated in the sense of the quantifier rank.

As in second order logic, there is a looser concept of a model, one relative to which we can prove a Completeness Theorem. This concept permits also a uniform definition.

Definition 2.5: A *Henkin L -structure* (or *Henkin L -model*) is a triple $(\mathcal{M}, \mathcal{U}, \mathcal{G})$, where \mathcal{M} is an L -structure, \mathcal{U} is a set such that $\emptyset \notin \mathcal{U}$ and \mathcal{G} is a set of relations between elements of the domains of \mathcal{M} and the sets in \mathcal{U} . We assume that the First and the Second Comprehension Axioms are satisfied by $(\mathcal{M}, \mathcal{U}, \mathcal{G})$ in the sense defined below.

The idea is that \mathcal{U} gives a set of possible domains for the new sorts needed for the truth conditions of the \exists -quantifiers, and \mathcal{G} gives a set of possible relations needed for the truth conditions of the \exists -quantifiers. Since \mathcal{U} is not the class of all sets (as it is a set) and \mathcal{G} need not be the set of *all* relevant relations, the structures $(\mathcal{M}, \mathcal{U}, \mathcal{G})$ are more general than the structures \mathcal{M} . The original structures \mathcal{M} are called *full*.

An *assignment* and a *modified assignment* for a Henkin L -structure $(\mathcal{M}, \mathcal{U}, \mathcal{G})$ is defined as for ordinary structures. Suppose $\mathfrak{s}(X) = (n_1, \dots, n_k)$. A model \mathcal{M}' is an *X -expansion in \mathcal{U}* of a model \mathcal{M} if $\{n_1, \dots, n_k\} \cap \text{dom}(\mathcal{M}) = \emptyset$, $\text{dom}(\mathcal{M}') = \text{dom}(\mathcal{M}) \cup \{n_1, \dots, n_k\}$, $\mathcal{M}' \upharpoonright \text{dom}(\mathcal{M}) = \mathcal{M}$, and $\mathcal{M}'(n_i) \in \mathcal{U}$ for all $i = 1, \dots, k$.

Definition 2.6: The *truth* of L -formulas in $(\mathcal{M}, \mathcal{U}, \mathcal{G})$ under s is defined as follows:

- (1) $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s R(x_1, \dots, x_n)$ if and only if $(s(x_1), \dots, s(x_n)) \in \mathcal{M}(R)$,
- (2) $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s x = y$ if and only if $s(x) = s(y)$,
- (3) $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s \neg\varphi$ if and only if $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \not\models_s \varphi$,
- (4) $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s (\varphi \vee \psi)$ if and only if $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s \varphi$ or $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s \psi$,
- (5) $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s \exists x\varphi$ if and only if $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_{s[a/x]} \varphi$ for some $a \in M_{\mathfrak{s}(x)}$,
- (6) $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s \exists X\varphi$ if and only if $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_{s[A/X]} \varphi$ for some $A \in \mathcal{P}(M_{n_1} \times \dots \times M_{n_k}) \cap \mathcal{G}$, where $\mathfrak{s}(X) = (n_1, \dots, n_k)$,
- (7) $(\mathcal{M}, \mathcal{U}, \mathcal{G}) \models_s \exists X\varphi$ if and only if $(\mathcal{M}, \mathcal{U}, \mathcal{G})' \models_{s[A/X]} \varphi$ for some X -expansion \mathcal{M}' of \mathcal{M} in \mathcal{U} and some $A \in \mathcal{P}(M'_{n_1} \times \dots \times M'_{n_k}) \cap \mathcal{G}$, where $\mathfrak{s}(X) = (n_1, \dots, n_k)$.

The following characterization of provability in sort logic is proved as the corresponding result for type theory [2]:

Theorem 2.7: (Completeness Theorem) *The following conditions are equivalent for any sentence φ of sort logic and any countable theory T of sort logic:*

- (1) $T \models \varphi$.
- (2) Every Henkin model of T satisfies φ .
- (3) Every countable Henkin model of T satisfies φ .

This characterization shows that our axioms for sort logic capture the intuition of sort logic in a perfect manner, at least if our Henkin semantics does. Our Henkin semantics is very much like that of second order logic.

3. Sort Logic and Set Theory

In this chapter we look at sort logic from the point of view of set theory.

Definition 3.1: We use Δ_n to denote the set of formulas of sort logic which are (semantically) equivalent both to a Σ_n -formula of sort logic, and to a Π_n -formula of sort logic.

Theorem 3.2: [7][6] *The following conditions are equivalent for any model class K and for any $n > 1$:*

- (1) K is definable in the logic Δ_n .
- (2) K is Δ_n -definable in the Levy-hierarchy.

Proof: We give the proof only in the case $n = 2$. The general case is similar. Suppose L is a finite vocabulary and \mathcal{A} is a second order characterizable L -structure. Suppose σ is the conjunction of a large finite part of ZFC. Let us call a model (M, \in) of θ *supertransitive* if for every $a \in M$ every element and every subset of a is in M . Let $\text{Sut}(M)$ be a Π_1 -formula which says that M is supertransitive. Let $\text{Voc}(x)$ be the standard definition of “ x is a vocabulary”. Let $\text{SO}(L, x)$ be the set-theoretical definition of the class of second order L -formulas. Let $\text{Str}(L, x)$ be the set-theoretical definition of L -structures. Let $\text{Sat}(\mathcal{A}, \varphi)$ be an inductive truth-definition of the Σ_2 -fragment of sort logic written in the language of set theory. Let

$$P(z, x, y) = \text{Voc}(z) \wedge \text{Str}(z, x) \wedge \text{SO}(z, y) \wedge \exists M(z, x, y \in M \wedge \sigma^{(M)} \wedge \text{Sut}(M) \wedge (\text{Sat}(z, x, y))^{(M)}).$$

Now if L is a vocabulary, \mathcal{A} an L -structure, then $\mathcal{A} \models \varphi \iff P(L, \mathcal{A}, \varphi)$. This shows that $\mathcal{A} \models \varphi$ is a Σ_2 property of \mathcal{A} and L .

For the converse, suppose the predicate Φ is a Σ_2 property of L -structures. There is a Σ_2 -sentence φ of sort logic such that for all \mathcal{M} , $\mathcal{M} \in K$ iff $\mathcal{M} \models \varphi$.

Suppose $\Phi = \exists x \forall y P(x, y, \mathcal{A})$, a Σ_2 -property of \mathcal{A} . Let ψ be a sort logic sentence the models of which are, up to isomorphism, exactly the models \mathcal{A} for which there is (V_α, \in) , with $\alpha = \beth_\alpha$, $\mathcal{A} \in V_\alpha$, and $(V_\alpha, \in) \models \exists x \forall y P(x, y, \mathcal{A})$. If $\exists x \forall y P(x, y, \mathcal{A})$ holds, we can find a model for ψ by means of the Levy Reflection principle. On the other hand, suppose ψ has a model \mathcal{A} . W.l.o.g. it is of the form (V_α, \in) with $\mathcal{A} \in V_\alpha$. Let $a \in V_\alpha$ such that $(V_\alpha, \in) \models \forall y P(a, y, \mathcal{A})$. Since in this case $H_\alpha = V_\alpha$, $(H_\alpha, \in) \models \forall y P(a, y, \mathcal{A})$, where H_α is the set of sets of hereditary cardinality $< \alpha$. By another application of the Levy Reflection Principle we get $(V, \in) \models \forall y P(a, y, \mathcal{A})$, and we have proved $\exists x \forall y P(x, y, \mathcal{A})$. \square

By a *model class* we mean a class of structures of the same vocabulary, which is closed under isomorphisms. In the context of set theory classes are referred to by their set-theoretical definitions.

The following consequence was mentioned in [5] without proof:

Corollary 3.3: [7][6] *Every model class is definable in sort logic. Sort logic is therefore the strongest logic.*

The logics Δ_n , $n = 2, 3, \dots$, provide a sequence of stronger and stronger logics. Their model theoretic properties can be characterized in set theoretical terms as the following results indicate:

Theorem 3.4: [7] *The Hanf-number of the logic Δ_n is δ_n . The Löwenheim number of the logic Δ_n is σ_n . The decision problem of the logic Δ_n is the complete Π_n -definable set of natural numbers.*

Theorem 3.5: [4] *The LST-number of Δ_2 is the first supercompact cardinal.*

Theorem 3.6: *The LST-number of Δ_3 is at least the first extendible cardinal.*

Theorem 3.7: *The decision problem of Δ_n is the complete Π_n -set of natural numbers.*

4. Sort Logic and Foundations of Mathematics

We suggest that sort logic can provide a foundation of mathematics in the same way as second order logic, with the strong improvement that it does not depend on the ad hoc Large Domain Assumptions of set theory.

We will now think of foundations of mathematics from the point of view of sort logic. Propositions of mathematics are—according to the sort logic view—either of the form

$$\mathcal{A} \models \varphi,$$

where \mathcal{A} is a structure, characterizable in sort logic, and φ is a sentence of sort logic, or else of the form

$$\models \varphi,$$

where again φ is a sentence of sort logic. Thus a proposition of mathematics either states a *specific* truth, truth in a specific structure, or a *general* truth, truth in all structures. The specific truth can be reduced to the general truth as follows. Suppose $\theta_{\mathcal{A}}$ is a sort logic sentence which characterizes \mathcal{A} up to isomorphism. Then

$$\mathcal{A} \models \varphi \iff \models \theta_{\mathcal{A}} \rightarrow \varphi.$$

Curiously, and quite unlike the case of second order logic, the converse holds, too. Suppose φ is a sort logic sentence in which the predicates P_1, \dots, P_k occur only. Let X_1, \dots, X_k be new unary predicate variables of sorts $\mathfrak{s}(P_1), \dots, \mathfrak{s}(P_k)$ respectively. Then

$$\models \varphi \iff \mathcal{A} \models \tilde{\forall} X_1 \dots \tilde{\forall} X_k \varphi(X_1 \dots X_k / P_1 \dots P_k)$$

$$\not\models \varphi \iff \mathcal{A} \models \neg \tilde{\forall} X_1 \dots \tilde{\forall} X_k \varphi(X_1 \dots X_k / P_1 \dots P_k).$$

Intuitively this says that a sort logic sentence which talks about the predicates P_1, \dots, P_k in some domains is valid if and only if whatever new domains and interpretations we take for P_1, \dots, P_k , φ is true. Since the general truth is reducible to specific truth we may focus on specific truth only, without loss of generality.

What is the justification we can give to asserting $\mathcal{A} \models \varphi$? We can *prove* from the axioms of sort logic the sentence $\theta_{\mathcal{A}} \rightarrow \varphi$. Of course, we may have to go beyond the standard axioms of sort logic, but much of mathematics can be justified with the sort logic axioms that we have.

What is the justification for asserting that φ has a model, i.e. $\neg\varphi$ is not valid? By the above it suffices to prove from the axioms the sentence $\neg \tilde{\forall} X_1 \dots \tilde{\forall} X_k \varphi(X_1 \dots X_k / P_1 \dots P_k)$. If we compare the situation of sort logic with that of second order logic the difference is that in second order logic we have to make so-called “large domain assumptions” to justify existence of mathematical structures, while in sort logic we can simply prove them from the general Comprehension Axioms. But here comes a moment

of truth. Can we actually prove the existence of the structures necessary in mathematics from the mere Comprehension Axioms?

In fact we need more axioms to supplement the First and the Second Comprehension Axiom.

Definition 4.1: Power Sort Axiom:

$$\begin{aligned} & \exists Y (\forall u \exists z_1 (u = z_1) \wedge \forall z_1 \exists u (u = z_1) \wedge \\ & \forall x \forall y (\forall z_1 \dots \forall z_n (Y x z_1 \dots z_n \leftrightarrow Y y z_1 \dots z_n) \rightarrow x = y) \wedge \\ & \forall X \exists x \forall z_1 \dots \forall z_n (X z_1 \dots z_n \leftrightarrow Y x z_1 \dots z_n)), \end{aligned} \quad (4.1)$$

where

$$\begin{aligned} \mathfrak{s}(X) &= (\mathfrak{s}(z_1), \dots, \mathfrak{s}(z_n)) \\ \mathfrak{s}(Y) &= (\mathfrak{s}(x), \mathfrak{s}(z_1), \dots, \mathfrak{s}(z_n)) \\ \mathfrak{s}(x) &= \mathfrak{s}(y) \\ \mathfrak{s}(z_1) &= \dots = \mathfrak{s}(z_n). \end{aligned}$$

Note that in the Power Sort Axiom only $\mathfrak{s}(u)$ occurs free, so it is an axiom about models with one sort, namely $\mathfrak{s}(u)$. Naturally the models may consist of other sorts as well, this axiom just does not say anything about those sorts. The sort $\mathfrak{s}(z_1)$ is just an auxiliary copy of the sort $\mathfrak{s}(u)$, as the conjunct $\forall u \exists z_1 (u = z_1) \wedge \forall z_1 \exists u (u = z_1)$ stipulates. The sort $\mathfrak{s}(x)$ is a new sort which codes the n -ary relations on the domain $\mathfrak{s}(u)$. The coding is done by means of the predicate Y .

Lemma 4.2: *Every full model satisfies the Power Sort Axiom.*

Proof: Suppose \mathcal{M} is a (full) model and s is an assignment into \mathcal{M} . Let us fix X, Y , and x, y, z_1, \dots, z_n such that $\mathfrak{s}(X) = (\mathfrak{s}(z_1), \dots, \mathfrak{s}(z_n))$, $\mathfrak{s}(Y) = (\mathfrak{s}(x), \mathfrak{s}(z_1), \dots, \mathfrak{s}(z_n))$, $\mathfrak{s}(x) = \mathfrak{s}(y)$ and $\mathfrak{s}(z_1) = \dots = \mathfrak{s}(z_n)$. Let \mathcal{N} be like \mathcal{M} except that there is a new sort $\mathfrak{s}(u)$ (or if this sort existed in \mathcal{M} it is now replaced) with universe $\mathcal{P}(M_{\mathfrak{s}(x)})$ and $\mathcal{N}_{\mathfrak{s}(z_1)} = M_{\mathfrak{s}(u)}$. Let s' be like s except that

$$s'(Y) = \{(a, b_1, \dots, b_n) \in N_{\mathfrak{s}(x)} \times M_{\mathfrak{s}(u)} \times \dots \times M_{\mathfrak{s}(u)} : (b_1, \dots, b_n) \in a\}.$$

Now s' satisfies in \mathcal{N} the formula

$$\begin{aligned} & \forall x \forall y (\forall z_1 \dots \forall z_n (Y x z_1 \dots z_n \leftrightarrow Y y z_1 \dots z_n)) \rightarrow x = y) \wedge \\ & \forall X \exists x \forall z_1 \dots \forall z_n (X z_1 \dots z_n \leftrightarrow Y x z_1 \dots z_n), \end{aligned}$$

Hence s satisfies in \mathcal{M} the formula (4.1). \square

Definition 4.3: Infinite Sort Axiom:

$$\begin{aligned}
& \exists X(\forall x\forall y\forall z((Xxy \wedge Xxz) \rightarrow y = z) \\
& \forall x\forall y\forall z((Xxz \wedge Xyz) \rightarrow x = y) \\
& \forall x\exists yXxy \\
& \exists z\forall x\forall y(Xxy \rightarrow \neg y = z))
\end{aligned} \tag{4.2}$$

where $\mathfrak{s}(X) = (\mathfrak{s}(x), \mathfrak{s}(x))$ and $\mathfrak{s}(x) = \mathfrak{s}(y) = \mathfrak{s}(z)$.

Lemma 4.4: *Every full model satisfies the Infinite Sort Axiom.*

Proof: Suppose \mathcal{M} is a (full) model and s is an assignment into \mathcal{M} . Let us fix X , and x, y, z such that $\mathfrak{s}(X) = (\mathfrak{s}(x), \mathfrak{s}(x))$, and $\mathfrak{s}(x) = \mathfrak{s}(y) = \mathfrak{s}(z)$. Let \mathcal{M}' be like \mathcal{M} except that there is a new sort $\mathfrak{s}(x)$ (or if this sort existed in \mathcal{M} it is now replaced) with universe ω . Let s' be like s except that

$$s'(X) = \{(n, n+1) : n \in \omega\}.$$

Now s' satisfies in \mathcal{M}' the formula

$$\begin{aligned}
& \forall x\forall y\forall z((Xxy \wedge Xxz) \rightarrow y = z) \\
& \forall x\forall y\forall z((Xxz \wedge Xyz) \rightarrow x = y) \\
& \forall x\exists yXxy \\
& \exists z\forall x\forall y(Xxy \rightarrow \neg y = z).
\end{aligned}$$

Hence s satisfies in \mathcal{M} the formula (4.2). \square

The Power Sort Axiom, reminiscent of the Power Set Axiom of set theory, is necessary for arguing about the existence of new sorts of elements. The Infinite Sort Axiom is required for arguing about infinite domains, just as we need the Axiom of Infinity in set theory. Note that the Power Sort Axiom or the Infinite Sort Axiom do not imply that we have only infinite or uncountable models. By the above lemmas these axioms are true in all models, even finite ones.

With the above two new axioms we can construct mathematical structures up to any cardinality $< \aleph_\omega$, as if we were working in Zermelo's set theory. For bigger structures we have to make stronger assumptions, and they probably have great similarity with the Replacement Axiom of set theory. The point is that in second (and higher) order logic we have to make ad hoc Large Domain Assumptions as we go from structure to structure, while in sort logic we need only make general assumptions about domains, as axioms of set theory postulate general properties of sets.

So what is the difference between sort logic and set theory? Despite its proximity to set theory, sort logic is still a logic, like first order logic, second

order logic, infinitary logic, etc. Sort logic treats mathematical structures up to isomorphism only, there is no preference of one construction of a structure over another, and this is in line with common mathematical thinking about structures. In its model theoretic formulation sort logic gives rise to interesting reflection principles via its Löwenheim-Skolem-Tarski properties. Finally, sort logic provides a natural model theoretic forum for investigating complicated set theoretical properties of models, without going into the nuts and bolts of the constructions of specific structures.

As the strongest logic sort logic is an ultimate yard-stick of definability in mathematics. Any property that is isomorphism invariant can be measured by sort logic. The canonical hierarchy Δ_n ($n < \omega$) inside sort logic climbs up the large cardinal hierarchy by reference to Hanf-, Löwenheim- and Skolem-Löwenhem-Tarski-numbers, reaching all the way to Vopenka's Principle. The model classes that are Δ_2 in the Levy-hierarchy are exactly the model classes definable in the Δ -extension of second order logic. Sort logic provides a similar characterization of model classes that are Δ_n in the Levy-hierarchy for $n > 2$. Is this too strong a logic to be useful? For logics as strong as sort logic the main use is in definability theory. But sort logic has also a natural axiomatization, complete with respect to a natural concept of a Henkin model, so we can also write inferences in sort logic. This is an alternative way of looking at mathematics to set theory, one in which definition rather than construction is the focus.

References

1. Solomon Feferman. Applications of many-sorted interpolation theorems. In *Proceedings of the Tarski Symposium (Proc. Sympos. Pure Math., Vol. XXV, Univ. of California, Berkeley, Calif., 1971)*, pages 205–223, Providence, R. I., 1974. Amer. Math. Soc.
2. Leon Henkin. Completeness in the theory of types. *J. Symbolic Logic*, 15:81–91, 1950.
3. David Hilbert and Wilhelm Ackermann. *Grundzüge der theoretischen Logik*. Springer-Verlag, Berlin, 1972. Sixth edition, Die Grundlehren der mathematischen Wissenschaften, Band 27. First edition published 1928.
4. M. Magidor. On the role of supercompact and extendible cardinals in logic. *Israel J. Math.*, 10:147–157, 1971.
5. J. A. Makowsky, Saharon Shelah, and Jonathan Stavi. Δ -logics and generalized quantifiers. *Ann. Math. Logic*, 10(2):155–192, 1976.
6. Juha Oikkonen. Second-order definability, game quantifiers and related expressions. *Soc. Sci. Fenn. Comment. Phys.-Math.*, 48(1):39–101, 1978.
7. Jouko Väänänen. Abstract logic and set theory. I. Definability. In *Logic Colloquium 78 (Mons, 1978)*, volume 97 of *Stud. Logic Foundations Math.*, pages

- 391–421. North-Holland, Amsterdam, 1979.
8. Jouko Väänänen. Second-order logic or set theory? *Bull. Symbolic Logic*, 18(1):91–121, 2012.