On the Semantic Equivalence of Language Syntax Formalisms

Samuele Buro, Isabella Mastroeni

Department of Computer Science, University of Verona
Strada le Grazie 15, 37134 Verona, Italy

Abstract

Several formalisms for language syntax specification exist in literature. In this paper, we prove that longstanding syntactical transformations between *context-free grammars* and *algebraic signatures* are *adjoint functors* and/or *adjoint equivalences* that preserve the *abstract syntax* of the generated terms. The main result is a *categorical equivalence* between the *categories of algebras* (*i.e.*, all the possible semantics) over the objects in these formalisms up to the provided syntactical transformations, namely that all these language specification frameworks are essentially the same from a semantic perspective.

Keywords: syntax specification, context-free grammars, algebraic signatures
2010 MSC: 00-01, 99-00

1. Introduction

Several formalisms for language syntax specification exist in literature [1]. Among them, *formal grammars* [2, 3, 4] and *algebraic signatures* [5, 6, 7] have played and still play a pivotal role. The former are widely used to define syntax of programming languages [8], notably due to compelling results on context-free parsing techniques [3, 9, 10]. The latter provide an algebraic approach to syntax specification, and they are ubiquitous in the fields of universal algebra [5], model theory [11], and logics in general.

Email addresses: `samuele.buro@univr.it` (Samuele Buro), `isabella.mastroeni@univr.it` (Isabella Mastroeni)

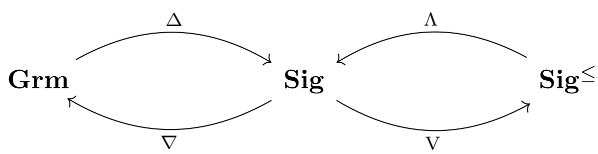


Figure 1: An informal overview of the mappings between the different syntax formalisms.

In this paper, we narrow the focus to three different syntax formalisms: *context-free grammars* (**Grm**), *many-sorted signatures* (**Sig**), and *order-sorted signatures* ($\mathbf{Sig}^{\leq}$). The aim is to provide mappings between these frameworks (see Figure 1) able to translate language syntax specifications from one formalism to another without altering their classes of semantics. Put differently, if $\mathbf{Alg}(\mathscr{X})$ denotes the class of semantics of an object $\mathscr{X}$ (in **Grm**, **Sig**, or $\mathbf{Sig}^{\leq}$) and $\Upsilon\mathscr{X}$ is its conversion to another formalism, we shall prove that $\mathbf{Alg}(\mathscr{X})$ and $\mathbf{Alg}(\Upsilon\mathscr{X})$ are equivalent, meaning that $\mathscr{X}$ and $\Upsilon\mathscr{X}$ are essentially the same from a semantic point of view.

Formally, this requires two constraints: (1) each syntactical transformation Υ shall preserve the *abstract syntax* of terms, and (2) it must exist a *categorical equivalence* between the *categories of algebras*[1] $\mathbf{Alg}(\mathscr{X})$ and $\mathbf{Alg}(\Upsilon\mathscr{X})$.

The mathematical links between these different frameworks have already been partially studied in literature. Goguen et al. [12] provide a definition of $\Delta\colon \mathbf{Grm} \to \mathbf{Sig}$ that yields an isomorphism between the sets of terms (*i.e.*, the *term algebras*) over G and its conversion to many-sorted signature ΔG, and conversely the definition of $\nabla\colon \mathbf{Sig} \to \mathbf{Grm}$ that makes the term algebras over $\mathfrak{S}$ and $\nabla\mathfrak{S}$ isomorphic (the proofs are outlined in detail in [13]). Other results on the subject are given in [7]. The authors provide a definition of $\Lambda\colon \mathbf{Sig}^{\leq} \to \mathbf{Sig}$ that gives rise to an equivalence between the categories of algebras over an order-sorted signature $\mathcal{S}$ and its many-sorted conversion $\Lambda\mathcal{S}$. Both these results of [12, 7] are an instance of the aim of this paper, as we will prove later.

[1] An algebra over an object $\mathscr{X}$ provides a meaning for all the symbols defined by $\mathscr{X}$. Thus, the category of algebras $\mathbf{Alg}(\mathscr{X})$ over $\mathscr{X}$ can be thought as the class of all the possible semantics for $\mathscr{X}$.

In the following sections, we unify and broaden such results in a more general setting. We model **Grm**, **Sig**, and $\mathbf{Sig}^{\leq}$ as the *categories* whose objects are grammars, many-sorted and order-sorted signatures, respectively (Sections 2.1 and 2.2). Arrows between objects in the same category are morphisms preserving the *abstract syntax* [14]. This is a fundamental point: According to [15],

> the essential syntactical structure of programming languages is not that given by their concrete or surface syntax [. . .]. Rather, the deep structure of a phrase should reflect its semantic import.

This viewpoint is also made explicit in [12, 16] where the semantics of a language is defined by the unique homomorphism from the *initial algebra* (*i.e.*, the abstract syntax) to another algebra in the same category.

The mappings from one formalism to another are therefore defined in terms of *functors* between the respective categories. Since the naturality of such constructions, the *adjoint* nature of these functors is then investigated, discussing their semantic implications over the categories of algebras (Sections 3, 4, and 5).

Contributions. The first contribution of this paper is the categorical description of several syntax transformation methods across different formalisms. In particular, we prove that some longstanding syntactical transformations between context-free grammars and many-sorted signatures and between many-sorted signatures and order-sorted signatures give rise to adjoint functors and/or adjoint equivalences that preserve the abstract syntax of the generated terms (Theorems 1 and 2). Moreover, we broaden some already known results of [12, 7, 13] and show that the aforementioned syntactical transformations preserve — up to an equivalence — the categories of algebras over the objects in their respective formalisms (Theorems 3, 4, and 5). The conclusion is twofold: Every categorical property and construction can be shifted between these frameworks (see, for instance, Example 4); and all these formalisms are essentially the same from a semantic perspective.

2. Formalisms for Language Syntax Specification

In this section, we provide a brief presentation of the three syntax formalisms discussed in the rest of the article. Their technical aspects are deferred to the next subsections.

The most popular formalism to specify languages are *context-free grammars*. They enable language designers to easily handle both abstract and concrete aspects of the syntax by combining terminal symbols with syntactic constituents of the language through production rules. Several definitions of context-free grammars exist in literature [13, 17]. Here, we are following [12] (or, the so-called *algebraic grammars* in [17]) and, for the sake of succintness, we sometimes refer to them simply as grammars.

Although grammars are an easy-to-use tool for syntax specification, *signatures* provide a more algebraic approach to language definition. The concept of *many-sorted signature* arose in [6] in order to lift the theory of (full) abstract algebras in case of partially defined operations. From the language syntax perspective, signatures allow the specification of sorted operators, which in turn provide a basis for an algebraic construction of the language semantics. In the rest of the paper, we follow the exposition of [12] and [18] on this subject.

The last formalism considered here are *order-sorted signatures* [7]. They are built upon many-sorted signatures to which they add an explicit treatment of polymorphic operators. Their main aim is to provide a basis on which to develop an algebraic theory to handle several types of polymorphism, multiple inheritance, left inverses of subsort inclusion (retracts), and complete equational deduction.

Basic Notions and Notations. Let S be a *set of sorts* and A a *carrier set*. An S*-sorted set* over A is uniquely determined by a function $\hat{a}\colon S \to \wp(A)$. Given $s \in S$, the A*-component* at s is $A_s = \hat{a}(s)$ and, by abusing the notation, we denote the whole sorted set induced by $\hat{a}$ as $A = \{ A_s \mid s \in S \}$.[2] Conversely, we usually define an S-sorted set A by defining each of its components A_s (assuming the undefined components to be the empty set). We implicitly extend set-theoretic operators and predicates to an S-sorted world in a componentwise fashion. For instance, if A and B are two S-sorted sets, we write $A \subseteq B$ if $A_s \subseteq B_s$ for each $s \in S$, we define the cartesian product $A \times B$ by taking each component $(A \times B)_s = A_s \times B_s$, etc.

Given two S-sorted sets A and B, an *S-sorted function* $h\colon A \to B$ is an S-sorted set $h \subseteq A \times B$ such that $h_s\colon A_s \to B_s$ is a set-theoretic function for each $s \in S$. If $f\colon A \to B$ and $g\colon B \to C$ are two S-sorted functions, one can check that their composition $g \circ f = \{ (g \circ f)_s = g_s \circ f_s \mid s \in S \}$ is still an S-sorted function from A to C. If A is an S-sorted set and $w = s_1 \dots s_n \in S^*$, we denote by A_w the cartesian product $A_{s_1} \times \cdots \times A_{s_n}$ (when $w = \varepsilon$, then $A_w = \{\bullet\}$ is the one-point domain). Likewise, if f is an S-sorted function and $a_i \in A_{s_i}$ for $i = 1 \dots n$, then the function $f_w\colon A_w \to B_w$ is defined by $f_w(a_1, \dots, a_n) = (f_{s_1}(a_1), \dots, f_{s_n}(a_n))$.

If $f\colon A \to B$ is a function defined by cases, we sometimes use the *conditional operator* $f(a) = [\![\, P(a) \;?\; b_1 \;⨾\; b_0 \,]\!]$ as a shorthand for $f(a) = b_1$ if the predicate P holds for a and b_0 otherwise. If A and B are two sets and $f\colon A \to B$ is a function, we denote by $f^*\colon A^* \to B^*$ the unique *monoid homomorphism* induced by the Kleene closure on the sets A and B extending the function f, *i.e.*, $f^*(a_1 \dots a_n) = f(a_1) \dots f(a_n)$. Moreover, if $g\colon A \to B$ is a function, we still use the symbol g to denote the *direct image map of* g (also called the *additive lift* of g), *i.e.*, the function $g\colon \wp(A) \to \wp(B)$ such that $g(X) = \{ g(a) \in B \mid a \in X \}$. Analogously, if $\leq$ is a binary relation on a set A (with elements $a \in A$), we use the same relation symbol to denote its *pointwise extension*, *i.e.*, we write $a_1 \dots a_n \leq a'_1 \dots a'_n$ for $a_1 \leq a'_1, \dots, a_n \leq a'_n$.

[2]The abuse of notation is twofold: Firstly, we are using the same letter A for denoting the carrier set of the sorted set and the sorted set itself; Secondly, we are letting two distinct but internally equal components $A_{s_1} = A_{s_2}$ to coexist in $A = \{ A_s \mid s \in S \}$.

Finally, if A is a set, we denote by $\mathbf{1}_A$ the set-theoretic identity function on A. Similarly, if X is an object in some category C, then $\mathbf{1}_X$ is the identity morphism on X, and $\mathbf{1}_C$ is the identity functor on C (or, the identity morphism on C in the category of categories $\mathbf{Cat}$).

2.1. Context-Free Grammars

A *context-free grammar* [12] (or, a *CF grammar*) is a triple $G = \langle N, T, P \rangle$, where N is the set of *non-terminal symbols* (or, *non-terminals*), T is the set of *terminal symbols* (or, *terminals*) disjoint from N, and $P \subseteq N \times (N \cup T)^*$ is the set of *production rules* (or, *productions*). If (A, β) is a production in P, we stick to the standard notation $A \to \beta$ (although some authors [13] reverse the order and write $\beta \to A$ to match the signature formalism). If $\alpha, \gamma \in (N \cup T)^*$, $B \in N$, and $B \to \beta \in P$, we define $\alpha B \gamma \Rightarrow \alpha\beta\gamma$ the *one-step reduction relation* on the set $(N \cup T)^*$. The language $\mathcal{L}(G)$ generated by G is the union of the N-sorted family $\mathcal{L}_N(G) = \{\, \mathcal{L}_A(G) \mid A \in N \,\}$, *i.e.*, $\mathcal{L}(G) = \bigcup \mathcal{L}_N(G)$, where $\mathcal{L}_A(G) = \{\, t \in T^* \mid A \Rightarrow^* t \,\}$ and $\Rightarrow^*$ is the reflexive transitive closure of $\Rightarrow$. The *non-terminals projection* $\mathrm{nt}\colon N \cup T \to N \cup \{\varepsilon\}$ on G is defined by $\mathrm{nt}(x) = (\!|\, x \in N \mathrel{?} x \mathrel{:} \varepsilon \,|\!)$. In the following, we implicitly characterize the function nt according to the subscript/superscript of G, namely, if G', G_1, etc. are grammars, we denote by nt', nt_1, etc. their non-terminals projections, respectively.

An *abstract grammar morphism* (henceforth *morphism*, when this terminology does not lead to ambiguities) $f\colon G_1 \to G_2$ is a map between two grammars $G_1 = \langle N_1, T_1, P_1 \rangle$ and $G_2 = \langle N_2, T_2, P_2 \rangle$ that preserves the abstract structure of the generated strings. Formally, f is a pair of functions $f_0\colon N_1 \to N_2$ and $f_1\colon P_1 \to P_2$ such that $f_1(A \to \beta) = f_0(A) \to \beta' \in P_2$, where $\mathrm{nt}_2^*(\beta') = (f_0^* \circ \mathrm{nt}_1^*)(\beta)$.

The identity morphism on an object $G = \langle N, T, P \rangle$ is denoted by $\mathbf{1}_G$ and is such that $(\mathbf{1}_G)_0 = \mathbf{1}_N$ and $(\mathbf{1}_G)_1 = \mathbf{1}_P$. The composition of two grammar morphism $f\colon G_1 \to G_2$ and $g\colon G_2 \to G_3$ is obtained by defining $(g \circ f)_0 = g_0 \circ f_0$ and $(g \circ f)_1 = g_1 \circ f_1$.

Proposition 1. *The class of all grammars and the class of all abstract grammar morphisms form the category* **Grm**.

The following section makes clear the semantic implications that a grammar morphism $f\colon G_1 \to G_2$ induces on the categories of algebras over G_1 and G_2. The insight is that preserving the abstract syntax of G_1 into G_2 ensures the possibility to employ G_2-algebras in order to provide meaning to G_1-terms.

2.1.1. Algebras over a Context-Free Grammar

The algebraic approach applied to context-free languages is introduced in [19, 16]. The authors exploit the theory of *heterogeneous algebras* [18] to provide semantics for context-free grammars (see also [12]). The algebraic notions that lead to the category of algebras over a context-free grammar are here summarized.

Let $G = \langle N, T, P \rangle$ be a grammar. A *G-algebra* [19, 12] is a pair $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$, where A is an N-sorted set of *semantic domains* (or, *carrier sets*) and $F_{\mathbb{A}} = \left\{ [\![C \to \delta]\!]_{\mathbb{A}} \colon A_{\mathrm{nt}^*(\delta)} \to A_C \;\middle|\; C \to \delta \in P \right\}$ is a set of *interpretation functions*. A *G-homomorphism* [19, 12] $h\colon \mathbb{A} \to \mathbb{B}$ between two G-algebras $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ and $\mathbb{B} = \langle B, F_{\mathbb{B}} \rangle$ is an N-sorted function $h\colon A \to B$ such that $[\![C \to \delta]\!]_{\mathbb{B}} \circ h_{\mathrm{nt}^*(\delta)} = h_C \circ [\![C \to \delta]\!]_{\mathbb{A}}$ for each production $C \to \delta \in P$.

It is well-known [12, 19] that the class of all G-algebras and the class of all G-homomorphisms form a category, denoted by $\mathbf{Alg}(G)$. The *initial object* in $\mathbf{Alg}(G)$ is the *term algebra* (or, *initial algebra*) and it is denoted by $\mathbb{T}$. Specifically, the carrier sets T_C of $\mathbb{T}$ are inductively defined as the smallest sets such that, if $C \to \delta \in P$ and $\mathrm{nt}^*(\delta) = \varepsilon$, then $C \to \delta \in T_C$, and, if $\mathrm{nt}^*(\delta) = C_1 \dots C_n$ and $t_i \in T_{C_i}$ for $i \in \{1, \dots, n\}$, then $C \to \delta\,\mathtt{(}t_1, \dots, t_n\mathtt{)} \in T_C$.[3] Then, the interpretation functions are obtained by defining $[\![C \to \delta]\!]_{\mathbb{T}} = C \to \delta$, if $\mathrm{nt}^*(\delta) = \varepsilon$, and $[\![C \to \delta]\!]_{\mathbb{T}}(t_1, \dots, t_n) = C \to \delta\,\mathtt{(}t_1, \dots, t_n\mathtt{)}$, if $\mathrm{nt}^*(\delta) = C_1 \dots C_n$ and

[3] The parentheses that occur in terms definition are not to be intended as those for the function application. For this reason, we use the monospaced font to disambiguate these two different situations.

$t_i \in T_{C_i}$ for $i \in \{1, \ldots, n\}$.

The following results spell out the identification of the *abstract syntax* with the *initial algebra* (they are simply the contextualization of basic category theory notions on *initiality*):

Proposition 2 (Goguen et al. [12]). *(i) If S and S' are both initial in a class C of algebras, then S and S' are isomorphic. If S'' is isomorphic to an initial algebra S, then S'' is also initial. (ii) An algebra S is initial in a class C of algebras if and only if for every A in C there exists a unique homomorphism $h\colon S \to A$.*

The first part of the proposition captures the independence from notational variation of the abstract syntax, while the second part ensures the existence of a unique *semantic function* towards each algebra in the category [12].

We now show the semantic effects that grammar morphisms induce on the respective categories of algebras. Let $G_1 = \langle N_1, T_1, P_1 \rangle$ and $G_2 = \langle N_2, T_2, P_2 \rangle$ be two context-free grammars. Suppose that $f\colon G_1 \to G_2$ is a grammar morphism and let $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ be a G_2-algebra. We can make $\mathbb{A}$ into a G_1-algebra $\xi_f \mathbb{A} = \langle \xi_f A, \xi_f F_{\mathbb{A}} \rangle$ by defining

$$(\xi_f A)_C = A_{f_0(C)} \qquad \text{for each } C \in N_1, \text{ and}$$

$$[\![C \to \delta]\!]_{\xi_f \mathbb{A}} = [\![f_1(C \to \delta)]\!]_{\mathbb{A}} \qquad \text{for each } C \to \delta \in P_1$$

Moreover, if $h\colon \mathbb{A} \to \mathbb{B}$ is a G_2-homomorphism, then $(\xi_f h)_C = h_{f_0(C)}$ is G_1-homomorphism from $\xi_f \mathbb{A}$ to $\xi_f \mathbb{B}$.

Proposition 3. *The map $\xi_f\colon \mathbf{Alg}(G_2) \to \mathbf{Alg}(G_1)$ induced by the abstract grammar morphism $f\colon G_1 \to G_2$ is a functor.*

This last proposition suggests to investigate the functorial nature of $\mathbf{Alg}(-)$.

Proposition 4. **Alg** *is a contravariant functor from the category of context-free grammar* **Grm** *to the category of small categories* **Cat***, or, equivalently,* $\mathbf{Alg}\colon \mathbf{Grm} \to \mathbf{Cat}^{\mathrm{op}}$ *is a (covariant) functor.*

Proof. Let $f\colon G_1 \to G_2$ be a grammar morphism and let $\mathbf{Alg}(f) = \xi_f$. Then, for each context-free grammar $G = \langle N, T, P\rangle$ and for each algebra $\mathbb{A} = \langle A, F_{\mathbb{A}}\rangle$ in $\mathbf{Alg}(G)$ holds that

$$(\mathbf{Alg}(\mathbf{1}_G)(A))_C = (\xi_{\mathbf{1}_G} A)_C = A_C \qquad \text{for each } C \in N, \text{ and}$$

$$[\![C \to \delta]\!]_{\mathbf{Alg}(\mathbf{1}_G)(\mathbb{A})} = [\![C \to \delta]\!]_{\xi_{\mathbf{1}_G}\mathbb{A}} = [\![C \to \delta]\!]_{\mathbb{A}} \qquad \text{for each } C \to \delta \in P$$

which imply $\mathbf{Alg}(\mathbf{1}_G) = \mathbf{1}_{\mathbf{Alg}(G)}$. Moreover, given two grammar morphisms $f\colon G_1 \to G_2$ and $g\colon G_2 \to G_3$ we have that

$$\begin{aligned}
\big(\mathbf{Alg}(g \circ f)(A)\big)_C &= (\xi_{(g\circ f)} A)_C \\
&= A_{(g_0 \circ f_0)(C)} \\
&= (\xi_g A)_{f_0(C)} \\
&= \big((\xi_f \circ \xi_g)(A)\big)_C \\
&= \big((\mathbf{Alg}(f) \circ \mathbf{Alg}(g))(A)\big)_C
\end{aligned}$$

for each $C \in N$, and

$$\begin{aligned}
[\![C \to \delta]\!]_{\mathbf{Alg}(g\circ f)(\mathbb{A})} &= [\![C \to \delta]\!]_{\xi_{(g\circ f)}\mathbb{A}} \\
&= [\![(g_1 \circ f_1)(C \to \delta)]\!]_{\mathbb{A}} \\
&= [\![f_1(C \to \delta)]\!]_{\xi_g \mathbb{A}} \\
&= [\![C \to \delta]\!]_{(\xi_f \circ \xi_g)(\mathbb{A})} \\
&= [\![C \to \delta]\!]_{(\mathbf{Alg}(f)\circ\mathbf{Alg}(g))(\mathbb{A})}
\end{aligned}$$

for each $C \to \delta \in P$, and thus $\mathbf{Alg}(g \circ f) = \mathbf{Alg}(f) \circ \mathbf{Alg}(g)$. □

Since functors preserve isomorphisms, we get that isomorphic grammars give rise to isomorphic categories of algebras, implying that f does not lose any (semantic relevant) information.

Corollary 1. *If $f\colon G_1 \to G_2$ is an abstract grammar isomorphism, then*

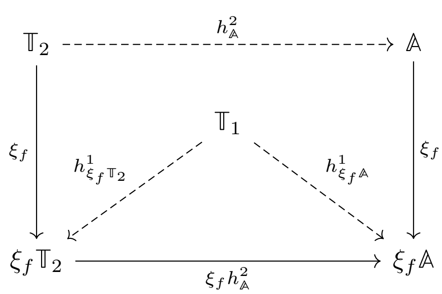
$$\xi_{f^{-1}} \circ \xi_f = \mathbf{1}_{\mathbf{Alg}(G_1)} \qquad \textit{and} \qquad \xi_f \circ \xi_{f^{-1}} = \mathbf{1}_{\mathbf{Alg}(G_2)}$$

Therefore, $\xi_{f^{-1}} = \xi_f^{-1}$ and hence $\mathbf{Alg}(G_1)$ and $\mathbf{Alg}(G_2)$ are isomorphic.

Example 1 (Deriving a Compiler). In this example, we show how a grammar morphism $f\colon G_1 \to G_2$ induces a compiler with respect to the semantic functions in $\mathbf{Alg}(G_2)$. Consider the following grammar specifications $G_1 = \langle N_1, T_1, P_1\rangle$ (left) and $G_2 = \langle N_2, T_2, P_2\rangle$ (right) in the Backus-Naur form:

$$\mathbb{n} ::= \texttt{+}\,\mathbb{n}\,\mathbb{n} \mid \texttt{0} \mid \texttt{1} \mid \texttt{2} \mid \cdots \qquad\qquad \mathbb{p} ::= \texttt{(}\,\mathbb{p}\,\texttt{+}\,\mathbb{p}\,\texttt{)} \mid \texttt{even} \mid \texttt{odd}$$

(Here, we have just specified the productions; terminals and non-terminals can be easily recovered from such specifications assuming no useless symbols in both sets). Let $f\colon G_1 \to G_2$ be the grammar morphism that maps $\mathbb{n}$ to $\mathbb{p}$, $\mathbb{n} \to \texttt{+}\,\mathbb{n}\,\mathbb{n}$ to $\mathbb{p} \to \texttt{(}\,\mathbb{p}\,\texttt{+}\,\mathbb{p}\,\texttt{)}$, and each production $\mathbb{n} \to \bar{n}$ to $\mathbb{p} \to \bar{p}$, where $\bar{n} \in \{\texttt{0}, \texttt{1}, \texttt{2}, \ldots\}$ and $\bar{p} = \texttt{even}$ if $\bar{n}$ represents an even natural number, and $\bar{p} = \texttt{odd}$ otherwise. Suppose that $\mathbb{A} = \langle A, F_{\mathbb{A}}\rangle$ is the G_2-algebra such that $A_{\mathbb{p}} = \{0, 1\}$, $[\![\mathbb{p} \to \texttt{even}]\!]_{\mathbb{A}} = 0$, $[\![\mathbb{p} \to \texttt{odd}]\!]_{\mathbb{A}} = 1$, and $[\![\mathbb{p} \to \texttt{(}\,\mathbb{p}\,\texttt{+}\,\mathbb{p}\,\texttt{)}]\!]_{\mathbb{A}}(p_1, p_2) = (p_1 + p_2) \mod 2$. Let $\mathbb{T}_1$ and $\mathbb{T}_2$ denote the G_1- and G_2-term algebras, respectively. Since $\mathbb{T}_2$ is initial, there is a unique homomorphism $h^2_{\mathbb{A}}\colon \mathbb{T}_2 \to \mathbb{A}$, *i.e.*, the semantics of the language generated by G_2 with respect to $\mathbb{A}$. Applying the functor ξ_f to $h^2_{\mathbb{A}}$ yields the following commutative diagram† (due to the initiality of $\mathbb{T}_1$) in $\mathbf{Alg}(G_1)$:



† $h^1_{\xi_f\mathbb{T}_2}$ and $h^1_{\xi_f\mathbb{A}}$ are the unique homomorphisms leaving $\mathbb{T}_1$.

In this case, the commutativity has an interesting meaning: $h^1_{\xi_f\mathbb{T}_2}$ is the compiler with respect to the semantic function $h^2_{\mathbb{A}}$ induced by the morphism f. Indeed, it is easy to show that for all terms $t \in (T_2)_{\mathbb{p}}$, holds that $\left(h^2_{\mathbb{A}}\right)_{\mathbb{p}}(t) = \left(\xi_f h^2_{\mathbb{A}}\right)_{\mathbb{n}}(t)$ and therefore

$$\xi_f h^2_{\mathbb{A}} \circ h^1_{\xi_f\mathbb{T}_2} = h^2_{\mathbb{A}} \circ h^1_{\xi_f\mathbb{T}_2} = h^1_{\xi_f\mathbb{A}}$$

which is the equation characterizing a compiler [20]. For instance, let `+ 5 3` denotes the $\mathbb{T}_1$-term $\mathbb{n} \to \mathtt{+}\mathbb{n}\mathbb{n}\mathtt{(}\mathbb{n} \to \mathtt{3}, \mathbb{n} \to \mathtt{5}\mathtt{)}$. If we apply the compiler $h^1_{\xi_f \mathbb{T}_2}$ to `+ 5 3`, we obtain a $\mathbb{T}_2$-term which $h^2_{\mathbb{A}}$-semantics agrees with $h^1_{\xi_f \mathbb{A}}$, *i.e.*, $\left(h^1_{\xi_f \mathbb{T}_2}\right)_{\mathbb{n}}(\mathtt{+\ 5\ 3}) = \mathtt{(odd\ +\ odd)}$ where `(odd + odd)` denotes the $\mathbb{T}_2$-term $\mathbb{p} \to \mathtt{(}\,\mathbb{p}\,\mathtt{+}\,\mathbb{p}\,\mathtt{)}\mathtt{(}\mathbb{p} \to \mathtt{odd}, \mathbb{p} \to \mathtt{odd}\mathtt{)}$, and

$$\left(h^2_{\mathbb{A}}\right)_{\mathbb{p}}(\mathtt{(odd\ +\ odd)}) = 0 = \left(h^1_{\xi_f \mathbb{A}}\right)_{\mathbb{n}}(\mathtt{+\ 5\ 3})$$

◁

2.2. Many-Sorted and Order-Sorted Signatures

A *many-sorted signature* [12] (or, an *MS signature*) is a pair $\mathfrak{S} = \langle S, \Sigma \rangle$, where S is a *set of sorts* and Σ is a disjoint[4] family of sets $\Sigma_{w,s}$ such that $w \in S^*$ and $s \in S$. As in the case of context-free grammars, we suppose that $S \cap \bigcup \Sigma = \varnothing$. If $\sigma \in \Sigma_{w,s}$, we call σ an *operator symbol* (or simply, an *operator*), and we write $\sigma \colon w \to s$ as a shorthand. Moreover, if $w = \varepsilon$, we say that σ is a *constant symbol* (or simply, a *constant*) and we write $\sigma \colon s$ instead of $\sigma \colon \varepsilon \to s$. Finally, given $\sigma \colon w \to s$, we define $\mathrm{ar}(\sigma) = w$ the *arity*, $\mathrm{srt}(\sigma) = s$ the *sort*, $\mathrm{rnk}(\sigma) = (w, s)$ the *rank* of σ.

A *many-sorted signature morphism* $f \colon \mathfrak{S}_1 \to \mathfrak{S}_2$ is a map between two many-sorted signatures $\mathfrak{S}_1 = \langle S_1, \Sigma_1 \rangle$ and $\mathfrak{S}_2 = \langle S_2, \Sigma_2 \rangle$ that preserves the underlying graph structure[5] of $\mathfrak{S}_1$ in $\mathfrak{S}_2$, in the following sense: f is a pair of functions $f_0 \colon S_1 \to S_2$ and $f_1 \colon \bigcup \Sigma_1 \to \bigcup \Sigma_2$ such that $f_1(\sigma) \colon f_0^*(w) \to f_0(s)$ in $\mathfrak{S}_2$ for each $\sigma \colon w \to s$ in $\mathfrak{S}_1$.

The identity arrow on $\mathfrak{S} = \langle S, \Sigma \rangle$ is denoted by $\mathbf{1}_{\mathfrak{S}}$ and is such that $(\mathbf{1}_{\mathfrak{S}})_0$ and $(\mathbf{1}_{\mathfrak{S}})_1$ are the set identity functions on their domains, and the composition of two morphisms $f \colon \mathfrak{S}_1 \to \mathfrak{S}_2$ and $g \colon \mathfrak{S}_2 \to \mathfrak{S}_3$ is obtained by defining $(g \circ f)_0 = g_0 \circ f_0$ and $(g \circ f)_1 = g_1 \circ f_1$, which is trivially a morphism from $\mathfrak{S}_1$ to $\mathfrak{S}_3$.

---

[4] Such a condition is not necessary and may be omitted at the cost of defining signature morphisms as sorted functions. We follow [12], and we adopt it to simplify the exposition.

[5] The graph similarity is obtained by considering an operator $\sigma \colon w \to s$ as a σ-labeled edge from w to s.

Proposition 5. *The class of all many-sorted signatures and the class of all many-sorted signature morphisms form the category* **Sig**.

Similarly, we introduce the theory of order-sorted signatures. An *order-sorted signature* [7] (or, an *OS signature*) is a triple $\mathcal{S} = \langle S, \leq, \Sigma \rangle$, where $\langle S, \leq \rangle$ is a poset of sorts and Σ is an $(S^* \times S)$-sorted family of sets $\Sigma_{w,s}$ such that satisfies the following condition: If $\sigma \in \Sigma_{w_1,s_1} \cap \Sigma_{w_2,s_2}$ and $w_1 \leq w_2$, then $s_1 \leq s_2$.

Note that S and Σ play the same role as before, except for the fact that Σ is no more required to be a disjoint family, thus enabling the definition of polymorphic operators. Furthermore, we extend to the order-sorted signatures the terminology that was introduced for the many-sorted case.

An *order-sorted signature morphism* $f\colon \mathcal{S}_1 \to \mathcal{S}_2$, where $\mathcal{S}_1 = \langle S_1, \leq_1, \Sigma_1 \rangle$ and $\mathcal{S}_2 = \langle S_2, \leq_2, \Sigma_2 \rangle$, is formed by the two components f_0 and f_1. The former component f_0 is a function between S_1 and S_2 that does not remove existing constraints, *i.e.*, if $s \leq_1 s'$ in $\langle S_1, \leq_1 \rangle$, then $f_0(s) \leq_2 f_0(s')$ in $\langle S_2, \leq_2 \rangle$. The latter component $f_1 = \{\, f^1_{w,s}\colon \Sigma^1_{w,s} \to \Sigma^2_{f_0^*(w), f_0(s)} \mid w \in S_1^* \land s \in S_1 \,\}$ is a set of functions between the sets of operators that preserve sorts and polymorphism, namely if $\sigma \in \Sigma_{w_1,s_1} \cap \Sigma_{w_2,s_2}$, then $f^1_{w_1,s_1}(\sigma) = f^1_{w_2,s_2}(\sigma)$.

The identity morphism $\mathbf{1}_{\mathcal{S}}$ over an order-sorted signature $\mathcal{S}$ is defined by taking $(\mathbf{1}_{\mathcal{S}})_0$ and each component $(\mathbf{1}_{\mathcal{S}})^1_{w,s}$ of $(\mathbf{1}_{\mathcal{S}})_1$ the set-theoretic identities on their domains. The composition $g \circ f$ of two order-sorted signature morphisms $f\colon \mathcal{S}_1 \to \mathcal{S}_2$ and $g\colon \mathcal{S}_2 \to \mathcal{S}_3$ is obtained by defining $(g \circ f)_0 = g_0 \circ f_0$ and $(g \circ f)^1_{w,s} = g^1_{f_0^*(w), f_0(s)} \circ f^1_{w,s}$.

Proposition 6. *The class of all order-sorted signatures and the class of all order-sorted signature morphisms form the category* $\mathbf{Sig}^{\leq}$.

2.2.1. Algebras over a Signature

In this section, we prove the same results developed in Section 2.1.1 for the classes of algebras over a many-sorted and order-sorted signature. Again, we provide the basic algebraic notions required to build the category of algebras over a given signature, and we redirect the reader to [7] for a thorough exposition of the following concepts.

Many-Sorted Algebra. Let $\mathfrak{S} = \langle S, \Sigma \rangle$ be a many-sorted signature. A *many-sorted* $\mathfrak{S}$*-algebra* [7] is a pair $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$, where A is an S-sorted set of *semantic domains* (or, *carrier sets*) and $F_{\mathbb{A}} = \left\{ [\![\sigma]\!]_{\mathbb{A}} \colon A_w \to A_s \;\middle|\; \sigma \in \Sigma_{w,s} \right\}$ is the set of *interpretation functions* (we use the same terminology adopted for an algebra over a context-free grammar). A *many-sorted* $\mathfrak{S}$*-homomorphism* [7] $h \colon \mathbb{A} \to \mathbb{B}$ between two many-sorted $\mathfrak{S}$-algebras $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ and $\mathbb{B} = \langle B, F_{\mathbb{B}} \rangle$ is an S-sorted function $h \colon A \to B$ such that $[\![\sigma]\!]_{\mathbb{B}} \circ h_w = h_s \circ [\![\sigma]\!]_{\mathbb{A}}$ for each $\sigma \in \Sigma_{w,s}$. The category of all $\mathfrak{S}$-algebras and $\mathfrak{S}$-homomorphisms is denoted by $\mathbf{Alg}(\mathfrak{S})$. The *many-sorted term* $\mathfrak{S}$*-algebra* $\mathbb{T}$ is the *initial algebra* in its category (*i.e.*, the *initial object*) and it is obtained in an analogous way to the term algebra over a grammar [7]. ◁

Order-Sorted Algebra. If $\mathcal{S} = \langle S, \leq, \Sigma \rangle$ is an order-sorted signature, an *order-sorted* $\mathcal{S}$*-algebra* [7] is a pair $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ where A is an S-sorted set and $F_{\mathbb{A}} = \{ [\![\sigma]\!]_{\mathbb{A}}^{w,s} \colon A_w \to A_s \mid \sigma \in \Sigma_{w,s} \}$. Moreover, the following monotonicity conditions must be satisfied:

(i) $\sigma \in \Sigma_{w_1,s_1} \cap \Sigma_{w_2,s_2}$ and $w_1 \leq w_2$ implies $[\![\sigma]\!]_{\mathbb{A}}^{w_1,s_1}(a) = [\![\sigma]\!]_{\mathbb{A}}^{w_2,s_2}(a)$ for each $a \in A_{w_1}$; and

(ii) $s_1 \leq s_2$ implies $A_{s_1} \subseteq A_{s_2}$.

An *order-sorted* $\mathcal{S}$*-homomorphism* $h \colon \mathbb{A} \to \mathbb{B}$ between two $\mathcal{S}$-algebras $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ and $\mathbb{B} = \langle B, F_{\mathbb{B}} \rangle$ is an S-sorted function $h \colon A \to B$ such that

(i) $[\![\sigma]\!]_{\mathbb{B}}^{w,s} \circ h_w = h_s \circ [\![\sigma]\!]_{\mathbb{A}}^{w,s}$ for each $\sigma \in \Sigma_{w,s}$; and

(ii) $s_1 \leq s_2$ implies $h_{s_1}(a) = h_{s_2}(a)$ for each $a \in A_{s_1}$.

In the following, we denote by $\mathbf{Alg}(\mathcal{S})$ the category formed by $\mathcal{S}$-algebras and $\mathcal{S}$-homomorphisms. The *order-sorted term* $\mathcal{S}$*-algebra* $\mathbb{T}$ is guaranteed to be initial only if $\mathcal{S}$ is *regular* (see [7] for the regularity definition and for details on the construction of $\mathbb{T}$ in the order-sorted case). ◁

We now have all the elements to show the semantic effects induced by a many-sorted signature morphism $f\colon \mathfrak{S}_1 \to \mathfrak{S}_2$, where $\mathfrak{S}_1 = \langle S_1, \Sigma_1 \rangle$ and $\mathfrak{S}_2 = \langle S_2, \Sigma_2 \rangle$. As in the case of context-free grammars, we can build a mapping from the category of algebras $\mathbf{Alg}(\mathfrak{S}_2)$ to $\mathbf{Alg}(\mathfrak{S}_1)$, in order to employ $\mathfrak{S}_2$-algebras to provide meaning to $\mathfrak{S}_1$-terms: Let $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ be an $\mathfrak{S}_2$-algebra. We can make $\mathbb{A}$ to a $\mathfrak{S}_1$-algebra $\zeta_f \mathbb{A} = \langle \zeta_f A, \zeta_f F_{\mathbb{A}} \rangle$ by defining

$$(\zeta_f A)_s = A_{f_0(s)} \qquad \text{for each } s \in S_1, \text{ and}$$
$$[\![\sigma]\!]_{\zeta_f \mathbb{A}} = [\![f_1(\sigma)]\!]_{\mathbb{A}} \qquad \text{for each } \sigma \in \bigcup \Sigma_1$$

Moreover, given a $\mathfrak{S}_2$-homomorphism $h\colon \mathbb{A} \to \mathbb{B}$, we can define the $\mathfrak{S}_1$-homomorphism $\zeta_f h\colon \zeta_f \mathbb{A} \to \zeta_f \mathbb{B}$ such that $(\zeta_f h)_s = h_{f_0(s)}$. The very same construction can be applied to the order-sorted case, namely, if $g\colon \mathcal{S}_1 \to \mathcal{S}_2$ is an order-sorted signature morphism, the map $\psi_g\colon \mathbf{Alg}(\mathcal{S}_2) \to \mathbf{Alg}(\mathcal{S}_1)$ is defined analogously to ζ_f.

Proposition 7. *The maps* $\zeta_f\colon \mathbf{Alg}(\mathfrak{S}_2) \to \mathbf{Alg}(\mathfrak{S}_1)$ *and* $\psi_g\colon \mathbf{Alg}(\mathcal{S}_2) \to \mathbf{Alg}(\mathcal{S}_1)$ *induced by the signature morphisms* $f\colon \mathfrak{S}_1 \to \mathfrak{S}_2$ *and* $g\colon \mathcal{S}_1 \to \mathcal{S}_2$, *respectively, are functors.*

Again, we can prove that **Alg** is a contravariant functor (we use the same name for the functor **Alg** both for categories of grammars and signatures):

Proposition 8. **Alg** *is a contravariant functor from the category of many-sorted signatures (order-sorted signatures)* **Sig** *(resp.,* $\mathbf{Sig}^{\leq}$*) to the category of small categories* **Cat**, *or, equivalently,* $\mathbf{Alg}\colon \mathbf{Sig} \to \mathbf{Cat}^{\mathrm{op}}$ *(resp.,* $\mathbf{Alg}\colon \mathbf{Sig}^{\leq} \to \mathbf{Cat}^{\mathrm{op}}$*) is a (covariant) functor.*

Proof. The proof is similar to the proof of Theorem 4. □

This last proposition leads to an equivalent of Corollary 1 for signature morphisms: Isomorphic signatures give rise to isomorphic categories of algebras, entailing that signature isomorphisms do not add or remove any semantic relevant information.

Proposition 9. *If $f\colon \mathfrak{S}_1 \to \mathfrak{S}_2$ and $g\colon \mathcal{S}_1 \to \mathcal{S}_2$ are isomorphism, then*

$$\zeta_{f^{-1}} \circ \zeta_f = \mathbf{1}_{\mathbf{Alg}(\mathfrak{S}_1)} \qquad \psi_{g^{-1}} \circ \psi_g = \mathbf{1}_{\mathbf{Alg}(\mathcal{S}_1)}$$
$$\zeta_f \circ \zeta_{f^{-1}} = \mathbf{1}_{\mathbf{Alg}(\mathfrak{S}_2)} \qquad \psi_g \circ \psi_{g^{-1}} = \mathbf{1}_{\mathbf{Alg}(\mathcal{S}_2)}$$

Therefore, $\zeta_{f^{-1}} = \zeta_f^{-1}$ and $\psi_{g^{-1}} = \psi_g^{-1}$, and thus ζ_f and ψ_g are isomorphisms.

3. Equivalence between MS Signatures and CF Grammars

In this section, we generalize the results of [12] by proving the conversion of a grammar into a signature and vice versa can be extended to functors that give rise to an *adjoint equivalence* between **Grm** and **Sig**. The major benefit of such new development is the preservation of all the categorical properties (such as initiality, limits, colimits, ...) from **Grm** to **Sig**, and vice versa. A concrete example is provided at the end of the section.

The map $\Delta\colon \mathbf{Grm} \to \mathbf{Sig}$ transforms a grammar $G = \langle N, T, P\rangle$ to the signature $\Delta G = \langle S_G, \Sigma_G\rangle$, where $S_G = N$ and $\Sigma^G_{w,s} = \{\, A \to \beta \in P \mid A = s \wedge \mathrm{nt}^*(\beta) = w \,\}$, and a grammar morphism $f\colon G_1 \to G_2$ to the signature morphism Δf such that $(\Delta f)_0 = f_0$ and $(\Delta f)_1 = f_1$.

Proposition 10. *$\Delta\colon$ **Grm** $\to$ **Sig** is a functor.*

Proof. The only non-trivial fact in the proof is checking that Δf satisfies the signature morphism condition: Let $G_1 = \langle N_1, T_1, P_1\rangle$ and $G_2 = \langle N_2, T_2, P_2\rangle$ be two context-free grammars, and let $f\colon G_1 \to G_2$ be a grammar morphism. If $\Delta G_1 = \langle S_{G_1}, \Sigma_{G_1}\rangle$ and $\Delta G_2 = \langle S_{G_2}, \Sigma_{G_2}\rangle$, then given $A \to \beta\colon\ \mathrm{nt}_1^*(\beta) \to A$ in Σ_{G_1} holds that

$$(\Delta f)_1(A \to \beta) = f_1(A \to \beta) = f_0(A) \to \beta'$$

where $(f_0^* \circ \mathrm{nt}_1^*)(\beta) = \mathrm{nt}_2^*(\beta')$, and therefore

$$\begin{aligned}(\Delta f)_1(A \to \beta)\colon&\ \mathrm{nt}_2^*(\beta') \to f_0(A)\\ :&\ (f_0^* \circ \mathrm{nt}_1^*)(\beta) \to f_0(A)\\ :&\ ((\Delta f)_0^* \circ \mathrm{nt}_1^*)(\beta) \to (\Delta f)_0(A)\end{aligned}$$

Hence Δf is a proper signature morphism from ΔG_1 to ΔG_2. ☐

Similarly, we define $\nabla\colon \mathbf{Sig} \to \mathbf{Grm}$ that maps objects and arrows between the specified categories. The conversion of a signature $\mathfrak{S} = \langle S, \Sigma\rangle$ to a grammar $\nabla\mathfrak{S} = \langle N_{\mathfrak{S}}, T_{\mathfrak{S}}, P_{\mathfrak{S}}\rangle$ is obtained by defining $N_{\mathfrak{S}} = S$, $T_{\mathfrak{S}} = \bigcup\Sigma$, and $P_{\mathfrak{S}} = \{\, s \to \sigma w \mid \sigma \in \Sigma_{w,s}\,\}$, while a signature morphism $f\colon \mathfrak{S}_1 \to \mathfrak{S}_2$ is mapped to the grammar morphism ∇f such that $(\nabla f)_0 = f_0$ and $(\nabla f)_1(s \to \sigma w) = f_0(s) \to f_1(\sigma) f_0^*(w)$.

Proposition 11. $\nabla\colon \mathbf{Sig} \to \mathbf{Grm}$ *is a functor.*

Proof. We show that ∇f yields a proper grammar morphism (the remaining part of the proof is trivial): Let $\mathfrak{S}_1 = \langle S_1, \Sigma_1\rangle$ and $\mathfrak{S}_2 = \langle S_2, \Sigma_2\rangle$ be two many-sorted signatures, and let $f\colon \mathfrak{S}_1 \to \mathfrak{S}_2$ be a signature morphism. Also, let $\nabla\mathfrak{S}_1 = \langle N_{\mathfrak{S}_1}, T_{\mathfrak{S}_1}, P_{\mathfrak{S}_1}\rangle$ and $\nabla\mathfrak{S}_2 = \langle N_{\mathfrak{S}_2}, T_{\mathfrak{S}_2}, P_{\mathfrak{S}_2}\rangle$, and let nt_{∇_1} and nt_{∇_2} denote the non-terminals projections on $\nabla\mathfrak{S}_1$ and $\nabla\mathfrak{S}_2$, respectively. Then,

$$(\nabla f)_1(s \to \sigma w) = f^*(s \to \sigma w) = f_0(s) \to f_1(\sigma) f_0^*(w)$$

for each $s \to \sigma w \in P_{\mathfrak{S}_1}$. Since the following chain of equalities holds

$$\mathrm{nt}^*_{\nabla_2}(f_1(\sigma) f_0^*(w)) = f_0^*(w) = f_0^*(\mathrm{nt}^*_{\nabla_1}(\sigma w)) = (f_0^* \circ \mathrm{nt}^*_{\nabla_1})(\sigma w)$$

then ∇f is a grammar morphism from $\nabla\mathfrak{S}_1$ to $\nabla\mathfrak{S}_2$. ☐

As underlined in [13] (and shown in the next example), Δ and ∇ are not isomorphisms. Indeed, in general, $\mathfrak{S} \neq \Delta\nabla\mathfrak{S}$ and $G \neq \nabla\Delta G$, and thus $\Delta\nabla \neq \mathbf{1}_{\mathbf{Sig}}$ and $\nabla\Delta \neq \mathbf{1}_{\mathbf{Grm}}$. However, as we prove in the next two propositions, there are natural isomorphisms η and ϵ^{-1} that transform the identity functors $\mathbf{1}_{\mathbf{Sig}}$ and $\mathbf{1}_{\mathbf{Grm}}$ to $\Delta\nabla$ and $\nabla\Delta$, respectively. It follows that $\mathfrak{S} \cong \Delta\nabla\mathfrak{S}$ and $G \cong \nabla\Delta G$ (where $\cong$ means *is isomorphic to*).

Example 2. Consider the following context-free grammar G (with terminal symbols underlined) for generating natural numbers in Peano's notation:

$$\mathsf{n} ::= \underline{\mathsf{s}}\,\mathsf{n} \mid \underline{\mathsf{0}}$$

Its conversion to signature via Δ and way back to grammar via ∇ is

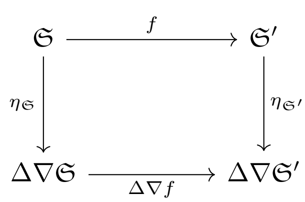
$$\mathbb{n} ::= \underline{\mathbb{n} \to \mathsf{s}\,\mathbb{n}}\,\mathbb{n} \;\big|\; \underline{\mathbb{n} \to \mathsf{0}}$$

Even though G and $\nabla\Delta G$ are different, there is a trivial grammar isomorphism f that maps $\mathbb{n} ::= \underline{\mathsf{s}}\,\mathbb{n}$ to $\mathbb{n} ::= \underline{\mathbb{n} \to \mathsf{s}\,\mathbb{n}}\,\mathbb{n}$ and $\mathbb{n} ::= \underline{\mathsf{0}}$ to $\mathbb{n} ::= \underline{\mathbb{n} \to \mathsf{0}}$. $\triangleleft$

Let $\mathfrak{S} = \langle S, \Sigma \rangle$ be a many-sorted signature. We denote by $\eta_{\mathfrak{S}} \colon \mathfrak{S} \to \Delta\nabla\mathfrak{S}$ the signature morphism defined by $(\eta_{\mathfrak{S}})_0 = \mathbf{1}_S$ and $(\eta_{\mathfrak{S}})_1(\sigma) = \mathrm{srt}(\sigma) \to \sigma\,\mathrm{ar}(\sigma)$. Since in the many-sorted case the arity and the rank are fully determined by the operator (Σ is a disjoint family of sets) the previous function is well-defined.

Proposition 12. $\eta \colon \mathbf{1}_{\mathbf{Sig}} \Rightarrow \Delta\nabla$ *is a natural isomorphism.*

Proof. Let $\mathfrak{S} = \langle S, \Sigma \rangle$ be a many-sorted signature and let $\sigma \colon w \to s$ in $\mathfrak{S}$. Then, $(\eta_{\mathfrak{S}})_1(\sigma) = s \to \sigma w$ has the same rank of σ. Since $(\eta_{\mathfrak{S}})_0$ is the identity on the set of sorts, $\eta_{\mathfrak{S}}$ satisfies the signature morphism condition. Moreover, it is easy to prove that each component $\eta_{\mathfrak{S}}$ is an isomorphism in **Sig** by defining its inverse $\eta_{\mathfrak{S}}^{-1}$ as $(\eta_{\mathfrak{S}}^{-1})_0 = \mathbf{1}_S$ and $(\eta_{\mathfrak{S}}^{-1})_1(s \to \sigma w) = \sigma$. We complete the proof by showing that the following diagram commutes for each signature morphism $f \colon \mathfrak{S} \to \mathfrak{S}'$:



The 0-th components of the morphisms in the diagram trivially commute. As regards the 1-th components, they commute if and only if $(\eta_{\mathfrak{S}'})_1(f_1(\sigma)) = (\Delta\nabla f)_1\big((\eta_{\mathfrak{S}})_1(\sigma)\big)$ for each $\sigma \in \Sigma_{w,s}$:

$$\begin{aligned}(\eta_{\mathfrak{S}'})_1(f_1(\sigma)) &= f_0(s) \to f_1(\sigma) f_0^*(w)\\ &= (\Delta\nabla f)_1(s \to \sigma w)\\ &= (\Delta\nabla f)_1\big((\eta_{\mathfrak{S}})_1(\sigma)\big)\end{aligned}$$

and hence the thesis. $\square$

Similarly, let $G = \langle N, T, P \rangle$ be a context-free grammar. We denote by $\epsilon_G \colon \nabla\Delta G \to G$ the grammar morphism defined by $(\epsilon_G)_0 = \mathbf{1}_N$ and $(\epsilon_G)_1\big(A \to (A, \beta)\,\mathrm{nt}^*(\beta)\big) = A \to \beta$.[6]

Proposition 13. $\epsilon \colon \nabla\Delta \Rightarrow \mathbf{1}_{\mathbf{Grm}}$ *is a natural isomorphism.*

Proof. Let $G = \langle N, T, P \rangle$ be a context-free grammar and let $A \to (A, \beta)\,\mathrm{nt}^*(\beta)$ be a production in $P_{\Delta G}$. Recall that

$$(\epsilon_G)_1(A \to (A, \beta)\,\mathrm{nt}^*(\beta)) = A \to \beta \qquad \text{and} \qquad (\epsilon_G)_0(A) = A$$

and therefore

$$((\epsilon_G)_0^* \circ \mathrm{nt}^*_{\nabla\Delta})((A, \beta)\,\mathrm{nt}^*(\beta)) = \mathrm{nt}^*(\beta)$$

where $\mathrm{nt}_{\nabla\Delta}$ is the non-terminals mapping on $\nabla\Delta G$. Thus, ϵ_G is a proper grammar morphism. Moreover, ϵ_G is an isomorphism in **Grm**: Let ϵ_G^{-1} denotes its inverse defined by

$$(\epsilon_G^{-1})_0 = \mathbf{1}_N \qquad \text{and} \qquad (\epsilon_G^{-1})_1(A \to \beta) = A \to (A, \beta)\,\mathrm{nt}^*(\beta)$$

Now one can check that $\epsilon_G \circ \epsilon_G^{-1} = \mathbf{1}_G$ and $\epsilon_G^{-1} \circ \epsilon_G = \mathbf{1}_{\nabla\Delta G}$. In order to prove the thesis, we show the commutativity of the following diagram for each grammar morphism $f \colon G \to G'$:

$$\begin{array}{ccc} \nabla\Delta G & \xrightarrow{\nabla\Delta f} & \nabla\Delta G' \\ \downarrow{\scriptstyle \epsilon_G} & & \downarrow{\scriptstyle \epsilon_{G'}} \\ G & \xrightarrow[f]{} & G' \end{array}$$

Since $\nabla\Delta f = f$ and $(\epsilon_G)_0$ and $(\epsilon_{G'})_0$ are the identity functions, we can conclude the commutativity of the 0-th components of the diagram. Moreover,

$$\begin{aligned} &(\epsilon_{G'})_1\big((\nabla\Delta f)_1(A \to (A, \beta)\,\mathrm{nt}^*(\beta))\big) \\ &\quad = (\epsilon_{G'})_1\big(f_0(A) \to f_1(A \to \beta)(f_0 \circ \mathrm{nt})^*(\beta)\big) \end{aligned}$$

---

[6] Note that the productions in $P_{\Delta G}$ are formed from those in P, *i.e.*, $P_{\Delta G} = \{\, A \to (A, \beta)\,\mathrm{nt}^*(\beta) \mid A \to \beta \in P \,\}$. Therefore, when considering a general production in $P_{\Delta G}$ derived from $A \to \beta$ in P, we write $A \to (A, \beta)\,\mathrm{nt}^*(\beta)$ (or, $A ::= \underline{A \to \beta}\,\mathrm{nt}^*(\beta)$ when considering a specific production in some example) instead of $A \to A \to \beta\,\mathrm{nt}^*(\beta)$ to avoid any confusion.

for each production rule $A \to (A, \beta)\,\mathrm{nt}^*(\beta)$ in $P_{\Delta G}$. Let $G' = \langle N', T', P' \rangle$. Since f is a grammar morphism and $A \to \beta \in P$, then $f_0(A) \to \beta' \in P'$ for some β' where $(\mathrm{nt}')^*(\beta') = (f_0^* \circ \mathrm{nt}^*)(\beta)$. Therefore, we can continue the previous chain of equalities:

$$\begin{aligned}
&= (\epsilon_{G'})_1\big(f_0(A) \to f_1(A \to \beta)(\mathrm{nt}')^*(\beta')\big) \\
&= f_0(A) \to \beta' \\
&= f_1(A \to \beta) \\
&= f_1\big((\epsilon_G)_1(A \to (A, \beta)\,\mathrm{nt}^*(\beta))\big)
\end{aligned}$$

and the proof is complete. $\square$

Example 3. Consider the context-free grammar G of the previous example. The grammar morphism ϵ_G transforms $\nabla\Delta G$ back to G. Indeed,

$$(\epsilon_G)_1(\mathtt{n} ::= \underline{\mathtt{n} \to \mathtt{s}\,\mathtt{n}}\,\mathtt{n}) = \mathtt{n} \to \underline{\mathtt{s}}\,\mathtt{n} \qquad \text{and}$$

$$(\epsilon_G)_1(\mathtt{n} ::= \underline{\mathtt{n} \to \mathtt{0}}) = \mathtt{n} \to \underline{\mathtt{0}}$$

$\triangleleft$

The previous results suggest to study if ∇ and Δ form an adjunction between the categories **Grm** and **Sig**.

Theorem 1. *∇ is left adjoint to Δ and (ϵ, η) are the counit and the unit of the adjunction $(\nabla, \Delta, \epsilon, \eta)$.*

Proof. We need to prove the following triangle equalities:

The 0-th components of both diagrams trivially commutes. We only prove the commutativity of the 1-th components.

- For each $s \to \sigma w \in P_{\mathfrak{S}}$

$$\begin{aligned}(\epsilon_{\nabla\mathfrak{S}})_1\big((\nabla\eta_{\mathfrak{S}})_1(s \to \sigma w)\big) &= (\epsilon_{\nabla\mathfrak{S}})_1\big((\eta_{\mathfrak{S}})_0(s) \to (\eta_{\mathfrak{S}})_1(\sigma)(\eta_{\mathfrak{S}})_0^*(w)\big) \\ &= (\epsilon_{\nabla\mathfrak{S}})_1(s \to (s, \sigma w)w) \\ &= s \to \sigma w\end{aligned}$$

- For each $A \to \beta \in \Sigma^G_{\mathrm{nt}^*(\beta),A}$

$$\begin{aligned}(\Delta\epsilon_G)_1\big((\eta_{\Delta G})_1(A \to \beta)\big) &= (\Delta\epsilon_G)_1(A \to (A, \beta)\,\mathrm{nt}^*(\beta)) \\ &= (\epsilon_G)_1(A \to (A, \beta)\,\mathrm{nt}^*(\beta)) \\ &= A \to \beta\end{aligned}$$

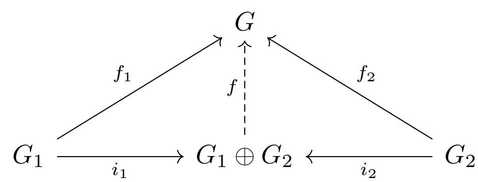
$\square$

Since ∇ is left adjoint to Δ (Theorem 1) and η and ϵ are natural isomorphisms (Propositions 12 and 13), we get the following corollary.

Corollary 2. $(\nabla, \Delta, \epsilon, \eta)$ *is an adjoint equivalence.*

Theorem 1 implies that **Grm** and **Sig** are identical except for the fact that each category may have different numbers of isomorphic copies of the same object. A consequence of this result is that we can move categorical limits between **Grm** and **Sig**. The next example provides a definition of *coproduct* in **Grm** able to recognize the union of two context-free languages. As a consequence of Theorem 1, we achieve for free the same construction in **Sig**.

Example 4 (Coproduct Preservation). Suppose to have the following notion of categorical coproduct in **Grm**: Given two context-free grammars $G_1 = \langle N_1, T_1, P_1\rangle$ and $G_2 = \langle N_2, T_2, P_2\rangle$, the coproduct of G_1 and G_2 is defined by $G_1 \oplus G_2 = \langle N_1 \uplus N_2, T_1 \uplus T_2, P_1 \uplus P_2\rangle$, where $\uplus$ is the disjoint union of sets. The inclusion morphism $i_k\colon G_k \to G_1 \oplus G_2$ for $k \in \{1, 2\}$ are defined by $(i_k)_0 = \mathbf{1}_{N_k}$ and $(i_k)_1 = \mathbf{1}_{P_k}$. Given two morphisms $f_1\colon G_1 \to G$ and $f_2\colon G_2 \to G$, where G is a context-free grammar, one can check that the unique morphism f that makes the following diagram commute



is obtained by defining $f_0(n) = \llbracket n \in N_1 \mathbin{?} (f_1)_0(n) \mathbin{:} (f_2)_0(n) \rrbracket$ and $f_1(A \to \beta) = \llbracket A \to \beta \in P_1 \mathbin{?} (f_1)_1(A \to \beta) \mathbin{:} (f_2)_1(A \to \beta) \rrbracket$. The term algebra over $G_1 \oplus G_2$ carries terms both in G_1 and G_2 and recognizes the (disjoint) union of the languages over G_1 and G_2. Since $(\nabla, \Delta, \epsilon, \eta)$ is an adjoint equivalence, then so is $(\Delta, \nabla, \eta^{-1}, \epsilon^{-1})$. Therefore, Δ is left adjoint to ∇ and hence it preserves colimits. Since a coproduct is a colimit, $\Delta(G_1 \oplus G_2)$ is the coproduct of ΔG_1 and ΔG_2 in **Sig**.

4. Equivalence between MS Signatures and OS Signatures

In this section, we show that similar results of those in Section 3 hold for many-sorted and order-sorted signature transformations Λ and V.

The map $\Lambda\colon \mathbf{Sig}^{\leq} \to \mathbf{Sig}$ converts an order-sorted signature $\mathcal{S} = \langle S, \leq, \Sigma \rangle$ to the many-sorted signature $\mathfrak{S}_{\mathcal{S}} = \langle S_{\mathcal{S}}, \Sigma_{\mathcal{S}} \rangle$ defined by $S_{\mathcal{S}} = S$ and $\Sigma^{\mathcal{S}}_{w,s} = \{ \sigma_{w,s} \mid \sigma \in \Sigma_{w,s} \}$ (such a construction is provided in [7]). The transformation of an order-sorted signature morphism $f\colon \mathcal{S}_1 \to \mathcal{S}_2$ to a many-sorted signature morphism $\Lambda f\colon \Lambda \mathcal{S}_1 \to \Lambda \mathcal{S}_2$ is obtained by defining $(\Lambda f)_0 = f_0$ and $(\Lambda f)_1(\sigma_{w,s}) = \big(f^1_{w,s}(\sigma)\big)_{f_0^*(w), f_0(s)}$.

Proposition 14. $\Lambda\colon \mathbf{Sig}^{\leq} \to \mathbf{Sig}$ *is a functor.*

Similarly, the map $\mathrm{V}\colon \mathbf{Sig} \to \mathbf{Sig}^{\leq}$ maps the many-sorted signature $\mathfrak{S} = \langle S, \Sigma \rangle$ to the order-sorted signature $\mathcal{S}_{\mathfrak{S}} = \langle S_{\mathfrak{S}}, \leq_{\mathfrak{S}}, \Sigma_{\mathfrak{S}} \rangle$, where $S_{\mathfrak{S}} = S$, $\leq_{\mathfrak{S}}$ is the reflexive binary relation on S, and $\Sigma^{\mathfrak{S}}_{w,s} = \Sigma_{w,s}$. Moreover, if $f\colon \mathfrak{S}_1 \to \mathfrak{S}_2$ is a many-sorted signature morphism, then $\mathrm{V}f\colon \mathrm{V}\mathfrak{S}_1 \to \mathrm{V}\mathfrak{S}_2$ defined by $(\mathrm{V}f)_0 = f_0$ and $(\mathrm{V}f)^1_{w,s} = f_1|_{\Sigma_{w,s}}$ is an order-sorted signature morphism.

Proposition 15. $\mathrm{V}\colon \mathbf{Sig} \to \mathbf{Sig}^{\leq}$ *is a functor.*

As before, we can provide natural transformations $\varphi\colon \mathbf{1}_{\mathbf{Sig}} \Rightarrow \Lambda\mathrm{V}$ and $\vartheta\colon \mathrm{V}\Lambda \Rightarrow \mathbf{1}_{\mathbf{Sig}^{\leq}}$. Let $\mathfrak{S} = \langle S, \Sigma\rangle$ be a many-sorted signature. Then, the $\mathfrak{S}$-component $\varphi_{\mathfrak{S}}\colon \mathfrak{S} \to \Lambda\mathrm{V}\mathfrak{S}$ of φ is defined by taking $(\varphi_{\mathfrak{S}})_0 = \mathbf{1}_S$ and $(\varphi_{\mathfrak{S}})_1(\sigma) = \sigma_{\mathrm{ar}(\sigma),\mathrm{srt}(\sigma)}$.

Proposition 16. $\varphi\colon \mathbf{1}_{\mathbf{Sig}} \Rightarrow \Lambda\mathrm{V}$ *is a natural isomorphism.*

Proof. The component $\varphi_{\mathfrak{S}}$ at $\mathfrak{S}$ of φ is trivially an invertible many-sorted signature morphism for each many-sorted signature $\mathfrak{S} = \langle S, \Sigma\rangle$. Thus, we only prove the naturality of φ, *i.e.*, that

$$\begin{array}{ccc}
\mathfrak{S} & \xrightarrow{\ f\ } & \mathfrak{S}' \\
{\scriptstyle \varphi_{\mathfrak{S}}}\downarrow & & \downarrow{\scriptstyle \varphi_{\mathfrak{S}'}} \\
\Lambda\mathrm{V}\mathfrak{S} & \xrightarrow[\ \Lambda\mathrm{V}f\ ]{} & \Lambda\mathrm{V}\mathfrak{S}'
\end{array}$$

commutes for each many-sorted signature morphism $f\colon \mathfrak{S} \to \mathfrak{S}'$. The 0-th component of the diagram commutes because $(\Lambda\mathrm{V}f)_0 = f_0$ and $\varphi_{\mathfrak{S}}$ and $\varphi_{\mathfrak{S}'}$ are the right and left identities for f, respectively. As regards the 1-th component of the diagram, we have that

$$\begin{aligned}
(\Lambda\mathrm{V}f)_1(\sigma_{w,s}) &= \big((\mathrm{V}f)_1(\sigma)\big)_{(\mathrm{V}f)_0^*(w),(\mathrm{V}f)_0(s)} \\
&= (f_1(\sigma))_{f_0^*(w),f_0(s)} \\
&= (\varphi_{\mathfrak{S}'})_1(f_1(\sigma))
\end{aligned}$$

and hence the thesis. □

Conversely, if $\mathcal{S} = \langle S, \leq, \Sigma\rangle$ is an order-sorted signature, the $\mathcal{S}$-component $\vartheta_{\mathcal{S}}\colon \mathrm{V}\Lambda\mathcal{S} \to \mathcal{S}$ of ϑ is obtained by defining $(\vartheta_{\mathcal{S}})_0 = \mathbf{1}_S$ and $(\vartheta_{\mathcal{S}})^1_{w,s}(\sigma_{w,s}) = \sigma$.

Proposition 17. $\vartheta\colon \mathrm{V}\Lambda \Rightarrow \mathbf{1}_{\mathbf{Sig}^{\leq}}$ *is a natural transformation.*

Proof. We prove the naturality of ϑ, *i.e.*, that

$$\begin{array}{ccc}
\mathrm{V}\Lambda\mathcal{S} & \xrightarrow{\ \mathrm{V}\Lambda f\ } & \mathrm{V}\Lambda\mathcal{S}' \\
{\scriptstyle \vartheta_{\mathcal{S}}}\downarrow & & \downarrow{\scriptstyle \vartheta_{\mathcal{S}'}} \\
\mathcal{S} & \xrightarrow[\ f\ ]{} & \mathcal{S}'
\end{array}$$

commutes for each order-sorted signature morphism $f \colon \mathcal{S} \to \mathcal{S}'$. The 0-th component of the diagram trivially commutes. As regards the 1-th component, we have that

$$\begin{aligned}
(\vartheta_{\mathcal{S}'})^1_{f_0^*(w),f_0(s)}\big((\mathrm{V}\Lambda f)^1_{w,s}(\sigma_{w,s})\big) &= (\vartheta_{\mathcal{S}'})^1_{f_0^*(w),f_0(s)}\big((\Lambda f)_1(\sigma_{w,s})\big) \\
&= (\vartheta_{\mathcal{S}'})^1_{f_0^*(w),f_0(s)}\big(f^1_{w,s}(\sigma)_{f_0^*(w),f_0(s)}\big) \\
&= f^1_{w,s}(\sigma) \\
&= f^1_{w,s}\big((\vartheta_{\mathcal{S}})^1_{w,s}(\sigma_{w,s})\big)
\end{aligned}$$

and hence the thesis. □

Again, Λ and V form an adjunction:

Theorem 2. V *is left adjoint to* Λ *and* (ϑ, φ) *are the counit and the unit of the adjunction* $(\mathrm{V}, \Lambda, \vartheta, \varphi)$.

Proof. We prove the following triangle equalities (0-th component trivially commutes):

- For each $\sigma \in \Sigma^{\mathfrak{S}}_{w,s}$

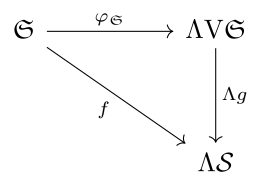
$$\begin{aligned}
(\vartheta_{\mathrm{V}\mathfrak{S}})^1_{w,s}\big((\mathrm{V}\varphi_{\mathfrak{S}})^1_{w,s}(\sigma)\big) &= (\vartheta_{\mathrm{V}\mathfrak{S}})^1_{w,s}\big((\varphi_{\mathfrak{S}})_1(\sigma)\big) \\
&= (\vartheta_{\mathrm{V}\mathfrak{S}})^1_{w,s}(\sigma_{w,s}) \\
&= \sigma
\end{aligned}$$

- For each $\sigma_{w,s} \in \Sigma^{\mathcal{S}}_{w,s}$

$$\begin{aligned}
(\Lambda\vartheta_{\mathcal{S}})_1\big((\varphi_{\Lambda\mathcal{S}})_1(\sigma_{w,s})\big) &= (\Lambda\vartheta_{\mathcal{S}})_1((\sigma_{w,s})_{w,s}) \\
&= \big((\vartheta_{\mathcal{S}})^1_{w,s}(\sigma_{w,s})\big)_{w,s} \\
&= \sigma_{w,s}
\end{aligned}$$

□

The results in this section can be rephrased in terms of *free constructions*. Indeed, the order-sorted signature $\mathrm{V}\mathfrak{S}$ is actually a free object on $\mathfrak{S}$ (together with the morphism $\varphi_{\mathfrak{S}}\colon \mathfrak{S} \to \Lambda\mathrm{V}\mathfrak{S}$). In this context, the functor $\Lambda\colon \mathbf{Sig}^{\leq} \to \mathbf{Sig}$ acts as a *forgetful functor* which forgets the ordering between sorts of the signature, whereas the *free functor* $\mathrm{V}\colon \mathbf{Sig} \to \mathbf{Sig}^{\leq}$ adds the loosest ordering on the set of sorts of a many-sorted signature (*i.e.*, the smallest reflexive relation). Therefore, it follows that, given a many-sorted signature $\mathfrak{S}$, for each order-sorted signature $\mathcal{S}$ and (many-sorted) morphism $f\colon \mathfrak{S} \to \Lambda\mathcal{S}$ there is a unique (order-sorted) morphism $g\colon \mathrm{V}\mathfrak{S} \to \mathcal{S}$ such that the following diagram commutes:



Example 5. In this example, we denote by $\equiv_X$ the smallest reflexive relation on a given set X, *i.e.*, $\equiv_X \,=\, \{\,(x,x) \mid x \in X\,\}$. Let $\mathfrak{S} = \langle S_1, \Sigma_1\rangle$ be the many-sorted signature with only two sorts $\mathbb{a}$ and $\mathbb{b}$, and two operators $\mathtt{a} : \mathbb{a}$ and $\mathtt{b} : \mathbb{b}$. The free order-sorted signature on $\mathfrak{S}$ is $\mathrm{V}\mathfrak{S} = \langle S_1, \equiv_{S_1}, \Sigma_1\rangle$. Now, let $\mathcal{S} = \langle S_2, \leq_2, \Sigma_2\rangle$ be the following order-sorted signature

$$\mathtt{a} : \mathbb{a} \qquad \mathtt{b} : \mathbb{b} \qquad \mathtt{c} : \mathbb{c} \qquad \text{where} \qquad \leq_2 \,=\, \{(\mathbb{a},\mathbb{b}), (\mathbb{a},\mathbb{c}), (\mathbb{b},\mathbb{c})\} \cup \equiv_{S_2}$$

The forgetful functor Λ maps $\mathcal{S}$ to $\Lambda\mathcal{S} = \langle S_2, \widehat{\Sigma}_2\rangle$, where $\widehat{\Sigma}^2_{w,s} = \{\mathtt{a}_{\varepsilon,\mathbb{a}}, \mathtt{b}_{\varepsilon,\mathbb{b}}, \mathtt{c}_{\varepsilon,\mathbb{c}}\}$. Suppose that $f\colon \mathfrak{S} \to \Lambda\mathcal{S}$ is the many-sorted morphism which its first component f_0 acts as the inclusion function from S_1 to S_2, and $f_1(\mathtt{a}) = \mathtt{a}_{\varepsilon,\mathbb{a}}$ and $f_1(\mathtt{b}) = \mathtt{b}_{\varepsilon,\mathbb{b}}$. The unique morphism $g\colon \mathrm{V}\mathfrak{S} \to \mathcal{S}$ which makes the previous diagram commute is the one who mimics the behavior of f, namely $g_0(s) = s$ for each $s \in S_1$ and $g^1_{w,s}(\sigma) = \sigma$ for each $\sigma \in \Sigma^1_{w,s}$.

5. Semantic Equivalence

In this section, we show that the provided syntactical transformations between context-free grammars and many-sorted signatures (Section 3) and between many-sorted and order-sorted signatures (Section 4) give rise to equivalent categories of algebras over the transformed objects.

More specifically, if $\Upsilon \in \{\Delta, \nabla, \Lambda, \mathrm{V}\}$ is a syntactical transformation and $\mathscr{X}$ is a language specification in the domain of Υ, then we prove that $\mathbf{Alg}(\mathscr{X})$ and $\mathbf{Alg}(\Upsilon\mathscr{X})$ are equivalent, namely

$$\begin{array}{ccc} \mathscr{X} & \xrightarrow{\quad\Upsilon\quad} & \Upsilon\mathscr{X} \\ \Big\downarrow{\scriptstyle \mathbf{Alg}} & & \Big\downarrow{\scriptstyle \mathbf{Alg}} \\ \mathbf{Alg}(\mathscr{X}) & \cong & \mathbf{Alg}(\Upsilon\mathscr{X}) \end{array}$$

Some of these equivalences are presented as *isomorphisms of categories*. It is well-known that an isomorphism of categories is a strong notion of categorical equivalence where functors compose to the identity.

5.1. Context-Free Grammars and Many-Sorted Signatures

As mentioned in the introduction, [12] proves an equivalence between the many-sorted term ΔG-algebra $\mathbb{T}_{\Delta G}$ and the initial algebra $\mathbb{T}_G$ over each grammar G. We now extend this result to the whole categories of algebras $\mathbf{Alg}(G)$ and $\mathbf{Alg}(\Delta G)$.

Let $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ be a G-algebra, and recall that the conversion of G to many-sorted signature is denoted by $\Delta G = \langle S_G, \Sigma_G \rangle$. Then, we map $\mathbb{A}$ to the many-sorted ΔG-algebra $\mathbb{A}^{\uparrow} = \langle A^{\uparrow}, F_{\mathbb{A}^{\uparrow}} \rangle$ such that $A^{\uparrow}_N = A_s$ for each $N \in S_G$ and $[\![C \to \delta]\!]_{\mathbb{A}^{\uparrow}} = [\![C \to \delta]\!]_{\mathbb{A}}$ for each $C \to \delta \in \bigcup \Sigma_G$ (operators in ΔG are productions in G). Furthermore, given a G-homomorphism $h\colon \mathbb{A} \to \mathbb{B}$, we define the ΔG-homomorphism $h^{\uparrow}\colon \mathbb{A}^{\uparrow} \to \mathbb{B}^{\uparrow}$ such that $h^{\uparrow}_N = h_N$.

Conversely, let $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ be a ΔG-algebra. Then, we define the inverse construction that maps $\mathbb{A}$ to the G-algebra $\mathbb{A}^{\downarrow} = \langle A^{\downarrow}, F_{\mathbb{A}^{\downarrow}} \rangle$ such that $A^{\downarrow}_N = A_N$ for each non-terminal N and $[\![C \to \delta]\!]_{\mathbb{A}^{\downarrow}} = [\![C \to \delta]\!]_{\mathbb{A}}$ for each production $C \to \delta$. Moreover, if $h\colon \mathbb{A} \to \mathbb{B}$ is a ΔG-homomorphism, then $h^{\downarrow}\colon \mathbb{A}^{\downarrow} \to \mathbb{B}^{\downarrow}$ such that $h^{\downarrow}_s = h_s$ is a proper G-homomorphism.

Theorem 3. *The inverse of* $(\_)^{\uparrow}$ *is* $(\_)^{\downarrow}$*, therefore they form an isomorphism of categories between* $\mathbf{Alg}(G)$ *and* $\mathbf{Alg}(\Delta G)$ *for each context-free grammar* G*.*

Since an isomorphism of categories is a strict notion of categorical equivalence, it preserves the initial objects, and thus, by applying $(\_)^{\uparrow}$ and $(\_)^{\downarrow}$ to the initial algebras, we have the exactly result of [12], *i.e.*, $\mathbb{T}_G^{\uparrow} = \mathbb{T}_{\Delta G}$ and $\mathbb{T}_{\Delta G}^{\downarrow} = \mathbb{T}_G$.

In a similar manner, given a many-sorted signature $\mathfrak{S}$, we can extend the equivalence between the initial algebras $\mathbb{T}_{\mathfrak{S}}$ and $\mathbb{T}_{\nabla\mathfrak{S}}$ to their whole categories of algebras $\mathbf{Alg}(\mathfrak{S})$ and $\mathbf{Alg}(\nabla\mathfrak{S})$.

Let $\mathbb{A} = \langle A, F_{\mathbb{A}}\rangle$ be a many-sorted $\mathfrak{S}$-algebra. We denote by $\nabla\mathfrak{S} = \langle N_{\mathfrak{S}}, T_{\mathfrak{S}}, P_{\mathfrak{S}}\rangle$ the context-free grammar obtained by converting the signature $\mathfrak{S}$. We define the $\nabla\mathfrak{S}$-algebra $^{\uparrow}\mathbb{A} = \langle ^{\uparrow}\mathbb{A}, F_{^{\uparrow}\mathbb{A}}\rangle$, where $^{\uparrow}A_s = A_s$ for each $s \in N_{\mathfrak{S}}$ and $[\![s \to \sigma w]\!]_{^{\uparrow}\mathbb{A}} = [\![\sigma]\!]_{\mathbb{A}}$ for each $s \to \sigma w \in P_{\mathfrak{S}}$. The conversion of a $\mathfrak{S}$-homomorphism $h\colon \mathbb{A} \to \mathbb{B}$ to a $\nabla\mathfrak{S}$-homomorphism $^{\uparrow}h\colon {^{\uparrow}\mathbb{A}} \to^{\uparrow} \mathbb{B}$ is analogous to the previous case.

On the contrary, if $\mathbb{A} = \langle A, F_{\mathbb{A}}\rangle$ is a $\nabla\mathfrak{S}$-algebra and $h\colon \mathbb{A} \to \mathbb{B}$ is a $\nabla\mathfrak{S}$-homomorphism, we can obtain a many-sorted $\mathfrak{S}$-algebra $^{\downarrow}\mathbb{A}$ and an $\mathfrak{S}$-homomorphism $^{\downarrow}h\colon {^{\downarrow}\mathbb{A}} \to^{\downarrow} \mathbb{B}$ by simply inverting the previous construction.

Theorem 4. *The inverse of* $^{\uparrow}(\_)$ *is* $^{\downarrow}(\_)$*, therefore they form an isomorphism of categories between* $\mathbf{Alg}(\mathfrak{S})$ *and* $\mathbf{Alg}(\nabla\mathfrak{S})$ *for each many-sorted signature* $\mathfrak{S}$*.*

Again, the result of [12] is a special case of this last theorem by noting that $^{\uparrow}\mathbb{T}_{\mathfrak{S}} = \mathbb{T}_{\nabla\mathfrak{S}}$ and $^{\downarrow}\mathbb{T}_{\nabla\mathfrak{S}} = \mathbb{T}_{\mathfrak{S}}$.

Example 6 (Example 4 Continued). In the Example 4, we have shown how to preserve categorical constructions between **Grm** and **Sig**. Theorems 3 and 4 can be applied on the top of Theorem 1 to ensure the semantic equivalence of the achieved constructions. For instance, if the $(G_1 \oplus G_2)$-algebra $\mathbb{A}$ provides the semantics of the disjoint union of languages over G_1 and G_2, then $\mathbb{A}^{\uparrow}$ provides the equivalent semantics in the category $\mathbf{Alg}(\Delta(G_1 \oplus G_2))$, as a consequence of Theorem 3.

5.2. Many-Sorted and Order-Sorted Signatures

The forgetful functor Λ transforms an order-sorted signature $\mathcal{S}$ to the many-sorted signature $\Lambda\mathcal{S}$ by forgetting the ordering on the sorts. In [7], the authors prove the categorical equivalence between $\mathbf{Alg}(\mathcal{S})$ and $\mathbf{Alg}(\Lambda\mathcal{S})$. We now extend such a result to its left adjoint V.

Let $\mathbb{A} = \langle A, F_{\mathbb{A}} \rangle$ be a many-sorted $\mathfrak{S}$-algebra and let $\mathrm{V}\mathfrak{S} = \langle S_{\mathfrak{S}}, \leq_{\mathfrak{S}}, \Sigma_{\mathfrak{S}} \rangle$. We define the order-sorted $\mathrm{V}\mathfrak{S}$-algebra $\mathbb{A}_{\uparrow} = \langle A_{\uparrow}, F_{\mathbb{A}_{\uparrow}} \rangle$ such that $(A_{\uparrow})_s = A_s$ for each $s \in S_{\mathfrak{S}}$ and $[\![\sigma]\!]^{w,s}_{\mathbb{A}_{\uparrow}} = [\![\sigma]\!]_{\mathbb{A}}$ for each $\sigma \in \Sigma^{\mathfrak{S}}_{w,s}$. Moreover, if $h \colon \mathbb{A} \to \mathbb{B}$ is an $\mathfrak{S}$-homomorphism, then $h_{\uparrow} \colon \mathbb{A}_{\uparrow} \to \mathbb{B}_{\uparrow}$ is the $\Lambda\mathfrak{S}$-homomorphism defined by $(h_{\uparrow})_s = h_s$. Furthermore, we denote by $(\_)_{\downarrow}$ the inverse functor that maps $\Lambda\mathfrak{S}$-algebras and $\Lambda\mathfrak{S}$-homomorphism to the category $\mathbf{Alg}(\mathfrak{S})$.

Theorem 5. *The inverse of* $(\_)_{\uparrow}$ *is* $(\_)_{\downarrow}$, *therefore they form an isomorphism of categories between* $\mathbf{Alg}(\mathfrak{S})$ *and* $\mathbf{Alg}(\mathrm{V}\mathfrak{S})$ *for each many-sorted signature* $\mathfrak{S}$.

6. Discussion and Related Works

This article is an extension of the conference paper [21] along three different ways: Firstly, we have added examples and detailed proofs for every major theorem in the paper. Secondly, we have studied the (contravariant) functoriality nature of **Alg** (Proposition 4 and Proposition 8) which has lead to a neater presentation of Corollary 1 and Corollary 9. Finally, we have strengthened the definition of an order-sorted morphism $f \colon \mathcal{S}_1 \to \mathcal{S}_2$ by (1) embedding the sort ordering of $\mathcal{S}_1$ into $\mathcal{S}_2$ through f_0 and by (2) preserving polymorphic operators. As a result, we have obtained a free construction in terms of adjoint functors (Theorem 2) — instead of an adjoint equivalence — that nevertheless gives rise to an isomorphism between the respective categories of algebras (Theorem 5).

The work most directly related to ours is [13], where the correspondence between context-free grammars and algebraic signatures is studied, both in the first-order and high-order setting. In particular, the author provides a proof (Proposition 2.15) of the isomorphism between the term algebra over a grammar and over its conversion to a many-sorted signature, and vice versa. Our work can be seen as generalizing these results (for the first-order case but as well including order-sorted specifications) towards a categorical framework, in which also a correspondence between internal transformations (morphisms) and external transformations (functors) is established. Moreover, this more abstract point of view has enabled us to extend the aforementioned isomorphisms to the whole category of algebras, not only to the initial object.

To the best of our knowledge, the first paper presenting the syntactical transformations studied in this work is [12]. More specifically, the authors provided the definitions of ΔG and $\nabla\mathfrak{S}$ with the properties described in [13]. A similar approach is taken in [7], where the definition of Λ is given, along with the proof (Theorem 4.2) that $\mathbf{Alg}(\mathcal{S})$ is equivalent to $\mathbf{Alg}(\Lambda\mathcal{S})$.

7. Conclusion and Further Research

In this paper, we have provided a categorical model of three different syntax formalisms (context-free grammars, many-sorted signatures and order-sorted signatures). We have shown how the extension to functors of already existing syntactical transformations gives rise to adjoint constructions able to preserve the abstract syntax of the generated terms. Finally, we have proved that the categories of algebras over the objects in these formalisms are categorically equivalent up to the provided transformations.

An obvious but important consequence of the underlying categorical model is the compositional nature of the proved results. Indeed, we can get a free equivalence between the category of grammars **Grm** and the category of order-sorted signatures $\mathbf{Sig}^{\leq}$ by simply composing $\mathrm{V}\Delta$ and $\nabla\Lambda$. The algebraic counterpart of the same observation allows us to claim that the composition of the functors $(\_)_{\downarrow} \circ (\_)^{\uparrow}$ gives rise to an isomorphism between $\mathbf{Alg}(G)$ and $\mathbf{Alg}(\mathrm{V}\Delta G)$ (and, of course, the dual result holds).

Further research concern refinements of the syntactical transformations between the formalisms in order to preserve specific properties of the concrete syntax [22]. Among them, *polymorphism* seems the most interesting. Unfortunately, the composition of functors $V\Delta$ and $\nabla\Lambda$ yields non-polymorphic set of operators. Another future work goes in the direction of providing syntactical transformation from **Grm** to $\mathbf{Sig}^{\leq}$ that yields only *regular* (see [7] for *regularity* definition) order-sorted signatures. Then, studying the adjoint of such a transformation could provide an interesting notion of regularity in the category of grammars that may be employed to weaken the standard notion of *ambiguity*.

References

[1] E. Visser, Syntax Definition for Language Prototyping, Ponsen & Looijen, 1997.

[2] N. Chomsky, Three models for the description of language, IRE Transactions on Information Theory 2 (3) (1956) 113–124.

[3] J. Earley, An Efficient Context-Free Parsing Algorithm, Communications of the ACM 13 (2) (1970) 94–102.

[4] D. E. Knuth, Semantics of Context-Free Languages, Mathematical Systems Theory 2 (2) (1968) 127–145.

[5] P. M. Cohn, Universal Algebra, Vol. 159, D. Reidel, Dordrecht, NL, 1981.

[6] P. J. Higgins, Algebras with a Scheme of Operators, Mathematische Nachrichten 27 (1-2) (1963) 115–132.

[7] J. A. Goguen, J. Meseguer, Order-sorted Algebra I: Equational Deduction for Multiple Inheritance, Overloading, Exceptions and Partial Operations, Theoretical Computer Science 105 (2) (1992) 217–273.

[8] M. L. Scott, Programming Language Pragmatics, Morgan Kaufmann Publishers, San Francisco, CA, USA, 2000.

[9] L. G. Valiant, General Context-free Recognition in Less Than Cubic Time, Journal of Computer and System Sciences 10 (2) (1975) 308–315.

[10] L. Lee, Fast Context-free Grammar Parsing Requires Fast Boolean Matrix Multiplication, Journal of the ACM 49 (1) (2002) 1–15.

[11] C. C. Chang, H. J. Keisler, Model Theory, Vol. 73, Elsevier, Essex, UK, 1990.

[12] J. A. Goguen, J. W. Thatcher, E. G. Wagner, J. B. Wright, Initial Algebra Semantics and Continuous Algebras, Journal of the ACM 24 (1) (1977) 68–95.

[13] E. Visser, Polymorphic Syntax Definition, Theoretical Computer Science 199 (1-2) (1998) 57–86.

[14] J. McCarthy, Towards a Mathematical Science of Computation, Springer, New York, NY, USA, 1993, Ch. The Mathematical Paradigm, pp. 35–56.

[15] M. Fiore, G. Plotkin, D. Turi, Abstract Syntax and Variable Binding, in: Proceedings of the 14th Annual IEEE Symposium on Logic in Computer Science, LICS, IEEE Computer Society, Washington, DC, USA, 1999, pp. 193–202.

[16] T. Rus, Context-Free Algebra: A Mathematical Device for Compiler Specification, in: International Symposium on Mathematical Foundations of Computer Science, Springer, New York, NY, USA, 1976, pp. 488–494.

[17] T. Rus, J. S. Jones, Multi-layered Pipeline Parsing from Multi-axiom Grammars, Algebraic Methods in Language Processing 95 (1995) 65–81.

[18] G. Birkhoff, J. Lipson, Heterogeneous Algebras, Journal of Combinatorial Theory 8 (1) (1970) 115 – 133.

[19] W. S. Hatcher, T. Rus, Context-Free Algebras, Cybernetics and System 6 (1-2) (1976) 65–77.

[20] Y. Futamura, K. Nogi, A. Takano, Essence of generalized partial computation, Theoretical Computer Science 90 (1) (1991) 61–79.

[21] S. Buro, I. Mastroeni, On the Semantic Equivalence of Language Syntax Formalisms, in: Proceedings of the 20th Italian Conference on Theoretical Computer Science, ICTCS, CEUR-WS.org, 2019, pp. 34–51.

[22] J. E. Hopcroft, R. Motwani, J. D. Ullman, Introduction to Automata Theory, Languages, and Computation (3rd Edition), Addison-Wesley, Boston, MA, USA, 2006.