Algebraic constructions: a simple framework for complex dependencies and parameterisation

Citation for published version:
Marczynski, G, Sannella, D & Tarlecki, A 2015, 'Algebraic constructions: a simple framework for complex dependencies and parameterisation' International Journal of Software and Informatics, vol. 9, no. 2, pp. 117-139.

Int J Software Informatics, Vol.XX, No.XX, XXXXXX 20XX, pp. 1–000
International Journal of Software and Informatics, ISSN 1673-7288

E-mail: ijsi@iscas.ac.cn
http://www.ijsi.org
Tel: +86-10-62661040

Algebraic constructions: a simple framework for complex dependencies and parameterisation

Grzegorz Marczyński[1], Donald Sannella[2] and Andrzej Tarlecki[1]

[1]Institute of Informatics, University of Warsaw

[2]Laboratory for Foundations of Computer Science, University of Edinburgh

Abstract We propose a simple framework of *algebraic constructions* for software specification, modular design and development. Algebraic constructions generalise (parameterised) modules by allowing on one hand a rather arbitrary collection of elements to form the parameter and on the other hand dependencies between the module elements to be spelled out explicitly. Algebraic constructions are specified in a very natural way by means of ordinary algebraic specifications. They are combined using a sum operation which captures as special cases various operations on (parameterised) modules offered by standard specification and development frameworks. We show the expected composability result for the sum of algebraic constructions and of their specifications.

Key words: Algebraic specification, Module, Parameterisation, Hierarchy, Dependency, Algebraic construction

1 Introduction

SPECTRAL [KBS91] was an experiment in specification language design in which both programs (modelled as algebras) and their specifications, both simple and parameterised, were viewed as first-class entities and could be arbitrarily used as parameters for each other. This extended to higher-order parameterisation, where entities like specifications parameterised by programs, specifications parameterised by parameterised programs, programs parameterised by specifications that in turn are parameterised by programs, etc., could be expressed.

The power of SPECTRAL made it possible to present various complex examples in a rather appealing way, but the full ramifications of this power meant that the details of the semantics, including the extent to which static typechecking would be possible, had to be left for subsequent work. Part of this was to be given by the kernel language for higher-order parameterisation presented in [ST91, SST92], amounting to a subset of SPECTRAL, which was equipped with a calculus for reasoning about higher-order parameterised programs and their specifications. But this calculus was later shown to suffer from serious technical problems [Asp97]. A "stratified" version in which these problems are absent is presented in [ST12].

We take a different approach here, somewhat in the spirit of [Gog90]. We enhance parameterised programs [Gog84, SST92] and their specifications by relaxing the requirement that the parameters of a module are, in a sense, complete (i.e., form an algebra over a *parameter signature*) and by introducing an explicit record of the possible dependencies between the various entities in the module. As it turns out, we may in this way capture some complex dependencies between entities within a module, which typically require the use of higher-order parameterisation in more standard approaches.

‡ Corresponding author: A. Tarlecki, E-mail: tarlecki@mimuw.edu.pl

Driven by this idea, we formalise the notion of an algebraic construction signature, of an algebraic construction over such a signature, and of specifications of algebraic constructions. We show that the category of algebraic construction signatures is finitely cocomplete, and that the colimits of such signatures admit amalgamation; this allows us to combine compatible algebraic constructions. Specifying algebraic constructions turns out to be very simple — any mechanism to specify the underlying algebras will suffice. Colimits of algebraic construction signatures allow us to define a sum operation both at the level of algebraic constructions and at the level of their specifications. Crucially, the compatibility of sums at these two levels can be shown.

The framework presented here is based on the ideas of the first author, see [Mar14] where they are worked out somewhat differently, but in greater generality and detail.

Special acknowledgement: We dedicate this paper to Bernd Krieg-Brückner, our professional colleague, long-term collaborator and good personal friend — all the best, Bernd! (DTS, AT)

2 Algebraic construction signatures

We will work here with the usual definition of algebraic (many-sorted) signatures — see for instance [ST12] for a more detailed presentation — except that we restrict attention to finite signatures and assume that all the symbols in a signature are unambiguous. In particular, sort names and operation names are distinct, and operation names are not overloaded. *Ad hoc* overloading of operation names could be added at the expense of the need for extra decoration; on the other hand, adding some form of *parametric polymorphism* would lead us to a different and interesting framework, where overloaded operations with the same names would have to behave in the same way w.r.t. the extra structure to be introduced below and its semantic consequences. Rather than trying to treat this in detail here, we view this option as a special case of a further, more general development, where the ideas presented are recast in the framework of an arbitrary institution [GB92, Mar14].

Hence, an algebraic signature is a quadruple $\Sigma = \langle S, \Omega, \mathit{arity}, \mathit{sort}\rangle$, where S and Ω are finite disjoint sets of sort and operation names, respectively, and $\mathit{arity}\colon \Omega \to S^*$, $\mathit{sort}\colon \Omega \to S$ give the profile of each operation name. Given an algebraic signature Σ as above, we write $f\colon s_1 \times \cdots \times s_n \to s$ for $f \in \Omega$, $\mathit{arity}(f) = s_1 \cdots s_n$ and $\mathit{sort}(f) = s$. Algebraic signature morphisms are defined as usual: $\sigma\colon \Sigma \to \Sigma'$ maps sort names in Σ to sort names in Σ' and operation names in Σ to operation names in Σ' preserving their arities and result sorts. With the usual component-wise composition, this yields the well-known, finitely cocomplete category $\mathbf{AlgSig}$ of algebraic signatures and their morphisms. The obvious functor $\mathbf{Symb}\colon \mathbf{AlgSig} \to \mathbf{Set}$[1], which maps any $\Sigma = \langle S, \Omega, \mathit{arity}, \mathit{sort}\rangle$ to $\mathbf{Symb}(\Sigma) = S \cup \Omega$ and any signature morphism to its underlying function, is cocontinuous.

To obtain algebraic construction signatures, we enhance algebraic signatures as above with information of two kinds. First, we indicate which symbols in a signature are considered *defined* and distinguish them from the remaining *assumed* symbols. Second, we introduce a *dependency* relation on the symbols, given by a strict order (that is, a relation that is transitive and irreflexive). The informal intuition is that an algebraic construction (to be introduced

[1] $\mathbf{Set}$ is the usual category of sets; we will also refer to $\mathbf{SET}$, the (quasi-)category of "large sets" (classes, discrete categories).

below) expects definitions for the assumed symbols to be obtained from the outside, while defining itself its defined symbols. Moreover, the definition for a symbol may use only the symbols below it in the dependency ordering.

Hence, an *algebraic construction signature* (or *construction signature* for short) is a triple $\mathcal{S} = \langle \Sigma, D, \prec \rangle$, where Σ is an algebraic signature, $D \subseteq \mathbf{Symb}(\Sigma)$ is a set of symbols *defined* in $\mathcal{S}$, and $\prec\, \subseteq \mathbf{Symb}(\Sigma) \times \mathbf{Symb}(\Sigma)$ is a strict order of *dependency* in $\mathcal{S}$ such that for each $f \colon s_1 \times \cdots \times s_n \to s$ in Σ, $s_1 \prec f, \ldots, s_n \prec f, s \prec f$. The required dependencies $s_1 \prec f$, $\ldots, s_n \prec f, s \prec f$, for $f \colon s_1 \times \cdots \times s_n \to s$, are called *basic* in Σ.

We say that a construction signature $\mathcal{S} = \langle \Sigma, D, \prec \rangle$ is *empty* if $D = \emptyset$; it is *complete* if $D = \mathbf{Symb}(\Sigma)$.

Consider a construction signature $\mathcal{S} = \langle \Sigma, D, \prec \rangle$. Let $X \subseteq \mathbf{Symb}(\Sigma)$ be a set of symbols that are mutually independent w.r.t. $\prec$. The *dependency structure below* X is defined as the construction signature $\mathcal{S}^X{\Downarrow} = \langle \Sigma', D', \prec' \rangle$, where Σ' is the unique subsignature of Σ with $\mathbf{Symb}(\Sigma') = \{y \in \mathbf{Symb}(\Sigma) \mid y \prec x \text{ for some } x \in X\}$, $D' = D \cap \mathbf{Symb}(\Sigma')$ and $\prec'$ is the restriction of $\prec$ to $\mathbf{Symb}(\Sigma')$. Then $X \cap \mathbf{Symb}(\Sigma') = \emptyset$.

For any set $X \subseteq \mathbf{Symb}(\Sigma)$ of symbols, the *dependency structure of* X is defined as the construction signature $\mathcal{S}^X{\downarrow} = \langle \Sigma'', D'', \prec'' \rangle$, where Σ'' is the unique subsignature of Σ with $\mathbf{Symb}(\Sigma'') = \{y \in \mathbf{Symb}(\Sigma) \mid y \in X \text{ or } y \prec x \text{ for some } x \in X\}$, $D'' = D \cap \mathbf{Symb}(\Sigma'')$ and $\prec''$ is the restriction of $\prec$ to $\mathbf{Symb}(\Sigma'')$.

Given $x \in \mathbf{Symb}(\Sigma)$, we write $\mathcal{S}^x{\Downarrow}$ and $\mathcal{S}^x{\downarrow}$ for $\mathcal{S}^{\{x\}}{\Downarrow}$ and $\mathcal{S}^{\{x\}}{\downarrow}$, respectively.

Given two construction signatures $\mathcal{S}_1 = \langle \Sigma_1, D_1, \prec_1 \rangle$ and $\mathcal{S}_2 = \langle \Sigma_2, D_2, \prec_2 \rangle$, an (algebraic) *construction signature morphism* $\sigma \colon \mathcal{S}_1 \to \mathcal{S}_2$ is an algebraic signature morphism $\sigma \colon \Sigma_1 \to \Sigma_2$ such that:

- defined symbols are preserved: $\sigma(D_1) \subseteq D_2$,
- dependencies are preserved: $\sigma(\prec_1) \subseteq\, \prec_2$,
- dependency down-closures are reflected: for all $a_1 \in \mathbf{Symb}(\Sigma_1)$ and $b_2 \in \mathbf{Symb}(\Sigma_2)$ such that $b_2 \prec_2 \sigma(a_1)$ there exists $b_1 \in \mathbf{Symb}(\Sigma_1)$ such that $b_1 \prec_1 a_1$ and $\sigma(b_1) = b_2$.

With the usual composition, this yields the category $\mathbf{ConSig}$ of construction signatures and their morphisms, with the obvious projection functor $\mathbf{Sig} \colon \mathbf{ConSig} \to \mathbf{AlgSig}$.

Fact 2.1. The category $\mathbf{ConSig}$ is finitely cocomplete, and the functor $\mathbf{Sig} \colon \mathbf{ConSig} \to \mathbf{AlgSig}$ is cocontinuous.

Proof. Coproducts of construction signatures in $\mathbf{ConSig}$ are essentially given as disjoint unions of the underlying algebraic signatures, dependency relations and sets of defined symbols. Hence, they are preserved by $\mathbf{Sig} \colon \mathbf{ConSig} \to \mathbf{AlgSig}$.

Coequalisers are easy to build: consider construction signatures $\mathcal{S}_1 = \langle \Sigma_1, D_1, \prec_1 \rangle$ and $\mathcal{S}_2 = \langle \Sigma_2, D_2, \prec_2 \rangle$, and construction signature morphisms $\sigma_1, \sigma_2 \colon \mathcal{S}_1 \to \mathcal{S}_2$. The coequaliser of σ_1, σ_2 is $\sigma \colon \mathcal{S}_2 \to \langle \Sigma_2/{\equiv}, \sigma(D_2), \sigma(\prec_2) \rangle$, where $\equiv$ is the least equivalence relation on $\mathbf{Symb}(\Sigma_2)$ such that $\sigma_1(x) \equiv \sigma_2(x)$ for all $x \in \mathbf{Symb}(\Sigma_1)$, $\Sigma_2/{\equiv}$ is the obvious quotient of Σ_2 by $\equiv$, and σ maps each symbol $y \in \mathbf{Symb}(\Sigma_2)$ to its equivalence class $[y]_{\equiv}$.

First notice that if for some $x \in \mathbf{Symb}(\Sigma_1)$, $y_1 = \sigma_1(x)$ and $y_2 = \sigma_2(x)$, and $y_1' \prec_2 y_1$, then since σ_1 reflects dependency down-closures, there is $x' \in \mathbf{Symb}(\Sigma_1)$ such that $x' \prec_1 x$ and $\sigma_1(x') = y_1'$, and hence $\sigma_2(x') \prec_2 y_2$ since σ_2 preserves dependencies. This gives the basis for an easy inductive proof that for any $y_1, y_2, y'_1 \in \mathbf{Symb}(\Sigma_2)$ such that $y_1 \equiv y_2$ and $y'_1 \prec_2 y_1$ there exists $y'_2 \in \mathbf{Symb}(\Sigma_2)$ such that $y'_2 \equiv y'_1$ and $y'_2 \prec y_2$.

We argue now that $\sigma(\prec_2)$ is indeed a strict order. Suppose that there exists $y_1 \in \mathbf{Symb}(\Sigma)$ such that for some $y_2 \in \mathbf{Symb}(\Sigma)$, $y_1 \prec_2 y_2$ and $y_1 \equiv y_2$, and consider such a minimal (w.r.t. $\prec_2$) y_1. By the above remark though, we have then $y'_1 \in \mathbf{Symb}(\Sigma_2)$ such that $y'_1 \equiv y_1$ and $y'_1 \prec_2 y_1$, which contradicts minimality of y_1. Hence, $\sigma(\prec_2)$ is irreflexive. Consider then $y_1, y_2, y_3, y_4 \in \mathbf{Symb}(\Sigma_2)$ such that $y_1 \prec_2 y_2$, $y_2 \equiv y_3$, $y_3 \prec_2 y_4$. By the above remark again, there is $y'_1 \in \mathbf{Symb}(\Sigma_2)$ such that $y'_1 \equiv y_1$ and $y'_1 \prec_2 y_3$, hence also $y'_1 \prec_2 y_4$, which shows that $\sigma(\prec_2)$ is transitive.

Hence, $\langle \Sigma_2/{\equiv}, \sigma(D_2), \sigma(\prec_2)\rangle$ is a construction signature. Moreover, by definition, $\sigma\colon \mathcal{S}_2 \to \langle \Sigma_2/{\equiv}, \sigma(D_2), \sigma(\prec_2)\rangle$ preserves defined symbols and dependencies, and reflects dependency down-closures, and so is a construction signature morphism. Its coequaliser property now follows easily from the fact that $\sigma\colon \Sigma_2 \to \Sigma_2/{\equiv}$ is a coequaliser of $\sigma_1, \sigma_2\colon \Sigma_1 \to \Sigma_2$ in $\mathbf{AlgSig}$, and that for any construction signature morphism $\sigma'\colon \Sigma_2 \to \Sigma'$ with $\sigma_1;\sigma' = \sigma_2;\sigma'$, the unique signature morphism $\sigma_0\colon \Sigma_2/{\equiv} \to \Sigma'$ such that $\sigma;\sigma_0 = \sigma'$ is in fact a construction signature morphism. □

The following lemma follows directly from the definition of a construction signature morphism:

Lemma 2.2. Let $\sigma\colon \mathcal{S}_1 \to \mathcal{S}_2$ be a construction signature morphism as above. Then for each symbol $x \in \mathbf{Symb}(\mathbf{Sig}(\mathcal{S}_1))$, $\sigma(\mathbf{Sig}(\mathcal{S}_1^x{\Downarrow})) = \mathbf{Sig}(\mathcal{S}_2^{\sigma(x)}{\Downarrow})$ and $\sigma(\mathbf{Sig}(\mathcal{S}_1^x{\downarrow})) = \mathbf{Sig}(\mathcal{S}_2^{\sigma(x)}{\downarrow})$. □

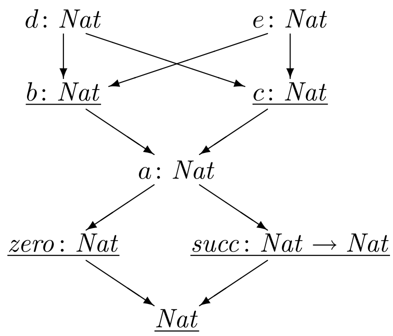
A *subsignature* of a construction signature $\mathcal{S} = \langle \Sigma, D, \prec\rangle$ is a construction signature $\mathcal{S}_1 = \langle \Sigma_1, D_1, \prec_1\rangle$ such that Σ_1 is a subsignature of Σ, the inclusion $\iota\colon \Sigma_1 \hookrightarrow \Sigma$ is a construction signature morphism $\iota\colon \mathcal{S}_1 \to \mathcal{S}$ and $D_1 = D \cap \mathbf{Symb}(\Sigma_1)$ (it follows that $\prec_1$ is $\prec$ restricted to $\mathbf{Symb}(\Sigma_1)$). Clearly, given a set $X \subseteq \mathbf{Symb}(\Sigma)$ of symbols in $\mathcal{S}$, $\mathcal{S}^X{\downarrow}$ is a subsignature of $\mathcal{S}$; if the symbols in X are independent then $S^X{\Downarrow}$ is also a subsignature of $\mathcal{S}^X{\downarrow}$ (as well as of $\mathcal{S}$).

Example 2.3 (Ordinary signatures and algebras). A usual algebraic signature may be captured as a complete construction signature by marking all symbols as defined (so that a complete construction signature is obtained) and taking basic dependencies only. □

Example 2.4 (Parameterised modules). In many standard frameworks (see for instance CASL [BM04, Mos04] or ACT ONE [EM85] — the latter is somewhat more general, but this does not change the point made below) a parameterised module (or *unit* in CASL) has a "type" that identifies a parameter signature Σ_P and extends it to a result signature Σ_R along a signature inclusion $\iota\colon \Sigma_P \hookrightarrow \Sigma_R$. Here this is captured by a construction signature $\mathcal{S}_\iota = \langle \Sigma_R, D, \prec\rangle$, where $D = \mathbf{Symb}(\Sigma_R) \setminus \mathbf{Symb}(\Sigma_P)$ and $\prec$ is generated by basic dependencies in Σ_R plus dependencies $p \prec r$ for all $p \in \mathbf{Symb}(\Sigma_P)$, $r \in D$.

Parameterisation of this kind restricts attention to constructions where the assumed symbols form a subsignature. Clearly, this is not necessarily the case with construction signatures. We do allow construction signatures where an operation symbol is assumed (i.e., under the above analogy, is part of the parameter) even if some of the sorts in its profile are defined (i.e., under the above analogy, are defined by the construction, and are not part of the parameter). □

Example 2.5 (Complex dependencies). A more complex construction signature is $\mathcal{S}_0 = \langle \Sigma_0, D_0, \prec_0 \rangle$, where the algebraic signature Σ_0 has a unique sort Nat, unary operation $succ\colon Nat \to Nat$, and constants $zero, a, b, c, d, e\colon Nat$, $D_0 = \{a, d, e\}$ and $\prec_0$ is generated by basic dependencies as well as $succ \prec_0 a$, $zero \prec_0 a$, $a \prec_0 b$, $a \prec_0 c$, $b \prec_0 d$, $c \prec_0 d$, $b \prec_0 e$, $c \prec_0 e$. This may be depicted as follows, underlining the assumed symbols:



The signature above illustrates more complex dependencies that may be captured in the framework proposed. We will use it to illustrate the technical issues such dependencies may involve. We refrain here from presenting any more practically meaningful case study, to keep the example relatively compact and hopefully easy to follow.

The dependencies here cannot result as the dependencies given by the standard parameterisation mechanism as sketched in Example 2.4. Under some standard approaches (e.g., SPECTRAL [KBS91]) we could resort to higher-order parameterisation, where we might capture the intended dependencies by the following "type" (listing just the symbols of the algebraic signatures involved):

$$\begin{aligned}&\{Nat, zero, succ\} \times \\ &\quad((\{Nat, zero, succ\} \to \{Nat, zero, succ, a\}) \to \{Nat, zero, succ, a, b, c\}) \\ &\qquad \to \{Nat, zero, succ, a, b, c, d, e\}\end{aligned}$$

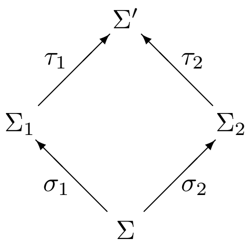
We will not attempt any formal claims concerning this analogy. Let us just note rather informally that constructions over the above construction signature will correspond only to what may be thought of as "cumulative" modules (where the result accumulates the parameters and their applications to other parameters one by one), which is a rather restrictive form of higher-order modules of the above higher-order type. □

3 Algebraic constructions

The overall idea is that an algebraic construction over a construction signature $\mathcal{S} = \langle \Sigma, D, \prec \rangle$ gives a way to provide a meaning for any defined symbol in D in terms of the meanings of the symbols in the dependency structure below this symbol. Our starting point is the usual definition of an algebra, which gives interpretations to symbols in an algebraic signature, see for instance [ST12].

Given an algebraic signature $\Sigma = \langle S, \Omega, arity, sort \rangle$, $\mathbf{Alg}(\Sigma)$ stands for the class of all Σ-algebras, defined as usual, except that we restrict attention to algebras with non-empty carriers to avoid minor technical problems in the sequel, which are by now well-understood, see [Tar11]. In fact, with the usual notion of Σ-homomorphism, $\mathbf{Alg}(\Sigma)$ is a category, but we may disregard homomorphisms for our purposes here. As usual, each signature morphism

$\sigma\colon \Sigma \to \Sigma'$ determines a *reduct* functor $\_|_\sigma\colon \mathbf{Alg}(\Sigma') \to \mathbf{Alg}(\Sigma)$,[2] which is injective for surjective σ. This yields a functor $\mathbf{Alg}\colon \mathbf{AlgSig}^{op} \to \mathbf{SET}$, which is continuous, and in particular maps signature pushouts to pullbacks in $\mathbf{SET}$, so that we have the so-called *amalgamation property*: given a pushout in $\mathbf{AlgSig}$



and algebras $A_1 \in \mathbf{Alg}(\Sigma_1)$ and $A_2 \in \mathbf{Alg}(\Sigma_2)$ such that $A_1|_{\sigma_1} = A_2|_{\sigma_2}$, there exists a unique algebra $A' \in \mathbf{Alg}(\Sigma')$ such that $A'|_{\tau_1} = A_1$ and $A'|_{\tau_2} = A_2$. When the pushout diagram is evident from the context, we write $A_1 \oplus A_2$ for A' and call it the *amalgamation* of A_1 and A_2. See [ST12] for a more detailed presentation.

Now, given an algebraic construction signature $\mathcal{S} = \langle \Sigma, D, \prec \rangle$, an *algebraic* $\mathcal{S}$*-construction* (or $\mathcal{S}$*-construction* for short) is a class $\mathcal{C} \subseteq \mathbf{Alg}(\Sigma)$ of Σ-algebras such that for any defined symbol $x \in D$ and any two algebras $A, A' \in \mathcal{C}$, if $A|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)} = A'|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)}$ then $A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} = A'|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$. In other words: in a construction, the interpretation of each defined symbol is unambiguously determined by the interpretation of the dependency structure below this symbol. However, the interpretation of assumed symbols remains unconstrained, and may vary within any construction.

A *trivial* $\mathcal{S}$-construction is the empty class of algebras (which is an $\mathcal{S}$-construction).

Lemma 3.1. Let $\sigma\colon \mathcal{S}_1 \to \mathcal{S}_2$ be a construction signature morphism from $\mathcal{S}_1 = \langle \Sigma_1, D_1, \prec_1 \rangle$ to $\mathcal{S}_2 = \langle \Sigma_2, D_2, \prec_2 \rangle$. For any $\mathcal{S}_2$-construction $\mathcal{C}_2 \subseteq \mathbf{Alg}(\Sigma_2)$, the σ*-reduct* $\mathcal{C}_1 = \mathcal{C}_2|_\sigma \subseteq \mathbf{Alg}(\Sigma_1)$ is an $\mathcal{S}_1$-construction.

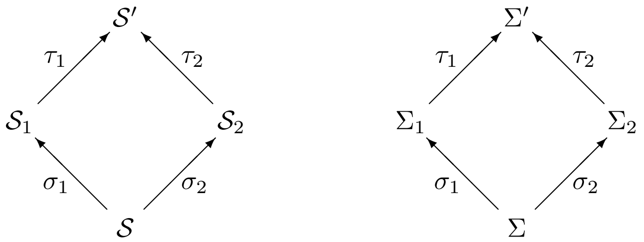
Proof. Let $x \in D_1$ be a defined symbol in $\mathcal{S}_1$. Given two Σ_2-algebras $A_2, A'_2 \in \mathcal{C}_2$ such that $(A_2|_\sigma)|_{\mathbf{Sig}(\mathcal{S}_1^x\Downarrow)} = (A'_2|_\sigma)|_{\mathbf{Sig}(\mathcal{S}_1^x\Downarrow)}$, by Lemma 2.2 we have $A_2|_{\mathbf{Sig}(\mathcal{S}_2^{\sigma(x)}\Downarrow)} = A'_2|_{\mathbf{Sig}(\mathcal{S}_2^{\sigma(x)}\Downarrow)}$. Hence, since $\mathcal{C}_2$ is an $\mathcal{S}_2$-construction, $A_2|_{\mathbf{Sig}(\mathcal{S}_2^{\sigma(x)}\downarrow)} = A'_2|_{\mathbf{Sig}(\mathcal{S}_2^{\sigma(x)}\downarrow)}$, and consequently, using Lemma 2.2 again, $(A_2|_\sigma)|_{\mathbf{Sig}(\mathcal{S}_1^x\downarrow)} = (A'_2|_\sigma)|_{\mathbf{Sig}(\mathcal{S}_1^x\downarrow)}$. ☐

We write $\mathbf{Con}(\mathcal{S})$ for the class of all $\mathcal{S}$-constructions. By Lemma 3.1, a construction signature morphism $\sigma\colon \mathcal{S}_1 \to \mathcal{S}_2$ determines a reduct function $\_|_\sigma\colon \mathbf{Con}(\mathcal{S}_2) \to \mathbf{Con}(\mathcal{S}_1)$. This yields a functor $\mathbf{Con}\colon \mathbf{ConSig}^{op} \to \mathbf{SET}$.

We will work now to establish an amalgamation property for constructions. For the rest of this section consider a construction signature pushout in $\mathbf{ConSig}$ and its projection on algebraic signatures, which is a pushout in $\mathbf{AlgSig}$:

---

[2] For a class $\mathcal{A}' \subseteq \mathbf{Alg}(\Sigma')$ of Σ'-algebras, $\mathcal{A}'|_\sigma = \{A'|_\sigma \mid A' \in \mathcal{A}'\}$ is the image of $\mathcal{A}'$ w.r.t. $\_|_\sigma$. Reducts $\_|_\iota$ w.r.t. a signature inclusion $\iota\colon \Sigma_s \hookrightarrow \Sigma_t$ are denoted by $\_|_{\Sigma_s}$.



Lemma 3.2. For any $\mathcal{S}_1$-construction $\mathcal{C}_1 \subseteq \mathbf{Alg}(\Sigma_1)$ and $\mathcal{S}_2$-construction $\mathcal{C}_2 \subseteq \mathbf{Alg}(\Sigma_2)$, their *amalgamation* $\mathcal{C}_1 \oplus \mathcal{C}_2 = \{A_1 \oplus A_2 \mid A_1 \in \mathcal{C}_1, A_2 \in \mathcal{C}_2, A_1|_{\sigma_1} = A_2|_{\sigma_2}\} \subseteq \mathbf{Alg}(\Sigma')$ is an $\mathcal{S}'$-construction.

Proof. Let $x' \in \mathbf{Symb}(\Sigma')$ be defined in $\mathcal{S}'$; suppose that $x' = \tau_1(x_1)$ where $x_1 \in \mathbf{Symb}(\Sigma_1)$ is defined in $\mathcal{S}_1$ (the other option, $x' = \tau_2(x_2)$ where $x_2 \in \mathbf{Symb}(\Sigma_2)$ is defined in $\mathcal{S}_2$, is symmetric). Consider $A', B' \in \mathcal{C}_1 \oplus \mathcal{C}_2$, where $A' = A_1 \oplus A_2$ and $B' = B_1 \oplus B_2$, for $A_1, B_1 \in \mathcal{C}_1$, $A_2, B_2 \in \mathcal{C}_2$, such that $A_1|_{\sigma_1} = A_2|_{\sigma_2}$ and $B_1|_{\sigma_1} = B_2|_{\sigma_2}$. Suppose $A'|_{\mathbf{Sig}(\mathcal{S}^{x'}\Downarrow)} = B'|_{\mathbf{Sig}(\mathcal{S}^{x'}\Downarrow)}$. Then, by Lemma 2.2, $A_1|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\Downarrow)} = B_1|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\Downarrow)}$, and so, since $\mathcal{C}_1$ is an $\mathcal{S}_1$-construction, $A_1|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\downarrow)} = B_1|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\downarrow)}$. Hence, by Lemma 2.2 again, $A'|_{\mathbf{Sig}(\mathcal{S}^{x'}\downarrow)} = B'|_{\mathbf{Sig}(\mathcal{S}^{x'}\downarrow)}$. □

Corollary 3.3. Constructions admit a weak amalgamation property: given an $\mathcal{S}_1$-construction $\mathcal{C}_1 \subseteq \mathbf{Alg}(\Sigma_1)$ and an $\mathcal{S}_2$-construction $\mathcal{C}_2 \subseteq \mathbf{Alg}(\Sigma_2)$ such that $\mathcal{C}_1|_{\sigma_1} = \mathcal{C}_2|_{\sigma_2}$, their amalgamation $\mathcal{C}' = \mathcal{C}_1 \oplus \mathcal{C}_2$ is an $\mathcal{S}'$-construction such that $\mathcal{C}'|_{\tau_1} = \mathcal{C}_1$ and $\mathcal{C}'|_{\tau_2} = \mathcal{C}_2$.

Proof. $\mathcal{C}'$ is an $\mathcal{S}'$-construction by Lemma 3.2. Clearly, $\mathcal{C}'|_{\tau_1} \subseteq \mathcal{C}_1$. Moreover, since $\mathcal{C}_1|_{\sigma_1} = \mathcal{C}_2|_{\sigma_2}$, for each $A_1 \in \mathcal{C}_1$ there is $A_2 \in \mathcal{C}_2$ such that $A_1|_{\sigma_1} = A_2|_{\sigma_2}$, and so $\mathcal{C}'|_{\tau_1} = \mathcal{C}_1$. Similarly, $\mathcal{C}'|_{\tau_2} = \mathcal{C}_2$. □

In general, $\mathcal{C}' = \mathcal{C}_1 \oplus \mathcal{C}_2$ need not be a unique $\mathcal{S}'$-construction such that $\mathcal{C}'|_{\tau_1} = \mathcal{C}_1$ and $\mathcal{C}'|_{\tau_2} = \mathcal{C}_2$. There may exist "weaker" $\mathcal{S}'$-constructions $\mathcal{C}'' \subset \mathcal{C}'$ with $\mathcal{C}''|_{\tau_1} = \mathcal{C}_1$ and $\mathcal{C}''|_{\tau_2} = \mathcal{C}_2$ — this may be the case when $\mathcal{C}''$ allows only some combinations of the interpretation of assumed symbols in $\mathcal{S}_1$ and $\mathcal{S}_2$, but some such combinations are missed. This lack of uniqueness is not a major problem in our view, since the amalgamation operation on constructions offers a natural *canonical* way to combine constructions over pushouts of construction signatures.

Given a construction signature $\mathcal{S} = \langle \Sigma, D, \prec \rangle$, we say that an $\mathcal{S}$-construction $\mathcal{C} \subseteq \mathbf{Alg}(\Sigma)$ is *well-grouped* if any interpretations of symbols it permits may be arbitrarily combined with each other, that is, for any set $X \subseteq \mathbf{Symb}(\Sigma)$ of symbols, for all Σ-algebras $A \in \mathbf{Alg}(\Sigma)$, if $A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in \mathcal{C}|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$ for all $x \in X$ then also $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$. Any well-grouped construction is non-trivial (to see this, consider $X = \emptyset$). Also, since in any construction the interpretation of the dependency structure below a defined symbol unambiguously determines the interpretation of this symbol, the condition may be limited to require arbitrary combinations of independent assumed symbols only.

Example 3.4 (Ordinary signatures and algebras). Recall Example 2.3. Given a complete construction signature $\mathcal{S}$, each non-trivial $\mathcal{S}$-construction consists of a single $\mathbf{Sig}(\mathcal{S})$-algebra. Clearly, amalgamation of such constructions corresponds to the amalgamation of the algebras, as expected, and similarly for reducts w.r.t. construction signature morphisms. □

Example 3.5 (Parameterised modules). Recall Example 2.4. Given a "type" $\iota \colon \Sigma_P \hookrightarrow \Sigma_R$, a parameterised module over ι is a (perhaps partial) function $F\colon \mathbf{Alg}(\Sigma_P) \to \mathbf{Alg}(\Sigma_R)$ that is *persistent*, i.e., for each $A \in \mathbf{Alg}(\Sigma_P)$ such that $F(A) \in \mathbf{Alg}(\Sigma_R)$ is defined, we have $F(A)\big|_{\Sigma_P} = A$. We may identify each such function F with its range $\mathcal{C}_F = \{F(A) \mid A \in \mathbf{Alg}(\Sigma_P)\}$. One may check now that $\mathcal{C}_F$ is a construction over the construction signature $\mathcal{S}_\iota$ that captures $\iota\colon \Sigma_P \hookrightarrow \Sigma_R$ as in Example 2.4. Moreover, any $\mathcal{S}_\iota$-construction determines a persistent (possibly partial) function from $\mathbf{Alg}(\Sigma_P)$ to $\mathbf{Alg}(\Sigma_R)$: given an $\mathcal{S}_\iota$-construction $\mathcal{C}$, we define $F_\mathcal{C}\colon \mathbf{Alg}(\Sigma_P) \to \mathbf{Alg}(\Sigma_R)$ by $F_\mathcal{C} = \{A_P \mapsto A_R \mid A_R \in \mathcal{C}, A_P = A_R\big|_{\Sigma_P}\}$. This is a well-defined function, since all symbols in Σ_R and not in Σ_P are defined in $\mathcal{S}_\iota$, therefore each Σ_P-algebra in $\mathcal{C}\big|_{\Sigma_P}$ has a unique expansion to a Σ_R-algebra in $\mathcal{C}$. □

Example 3.6 (Complex dependencies). Recall Example 2.5 and the construction signature $\mathcal{S}_0$ defined there. An $\mathcal{S}_0$-construction may be built as follows. We start with an arbitrary collection of interpretations for Nat, $zero\colon Nat$ and $succ\colon Nat$. Then, each such interpretation is uniquely extended by a value for $a\colon Nat$. Once this is given, interpretations for $b\colon Nat$ and $c\colon Nat$ may be given freely again — but the choice of their values must not influence the value of $a\colon Nat$. Finally, for each such interpretation of Nat, $zero\colon Nat$, $succ\colon Nat$, $(a\colon Nat)$, $b\colon Nat$ and $c\colon Nat$, unique values for $d\colon Nat$ and $e\colon Nat$ may be given.

Consider the following Σ_0-algebras that interpret Nat, $zero\colon Nat$ and $succ\colon Nat$ in the standard way, and:

$$A_1 = \{a = 1, b = 2, c = 3, d = 3, e = 4\}$$
$$A_2 = \{a = 2, b = 3, c = 4, d = 4, e = 5\}$$
$$A_3 = \{a = 1, b = 3, c = 4, d = 4, e = 5\}$$
$$A_4 = \{a = 1, b = 3, c = 3, d = 4, e = 4\}$$
$$A_5 = \{a = 1, b = 2, c = 4, d = 3, e = 5\}$$

Now, $\{A_1, A_2, A_3, A_4\}$ is not an $\mathcal{S}_0$-construction, since A_1 and A_2 define a differently even though they interpret the dependency structure below a in the same way. On the other hand, $\mathcal{C} = \{A_1, A_3, A_4\}$ is an $\mathcal{S}_0$-construction: different values of d and e result from different interpretations of the dependency structure below d and e. However, $\mathcal{C}$ is not well-grouped: b and c are independent, but their values are not combined arbitrarily here, the combination $b = 2$ and $c = 4$ is missing. $\mathcal{C}_0 = \{A_1, A_3, A_4, A_5\}$ is a well-grouped $\mathcal{S}_0$-construction. □

4 Construction specifications

We will not try to present here any specific framework for algebraic specifications — see [ST12] for an overview, a presentation of such a framework and historical remarks. We assume as given a class *Spec* of *specifications* with a semantics that for each specification $SP \in Spec$ yields its *signature* $Sig[SP] \in |\mathbf{AlgSig}|$ and a class of *models* $Mod[SP] \subseteq \mathbf{Alg}(Sig[SP])$. A specification SP is *consistent* if $Mod[SP] \neq \emptyset$. Specifications SP with $Sig[SP] = \Sigma$ will be referred to as Σ-*specifications*. Typically, specifications can be given simply as theories of some standard logic (like equational logic, or first-order logic with equality), perhaps with various forms of higher-order constraints added, as well as built from such *basic specifications* by means of predefined specification-building operations.

We assume that the class of specifications is closed under the following specification-building operations [ST88]; see [ST12] for extensive explanation and examples.

union: Given two Σ-specifications SP_1, SP_2, we also have a specification $SP_1 \cup SP_2$ with $Sig[SP_1 \cup SP_2] = \Sigma$, $Mod[SP_1 \cup SP_2] = Mod[SP_1] \cap Mod[SP_2]$.

translation: Given a Σ-specification SP and a signature morphism $\sigma\colon Sig[SP] \to \Sigma'$ we also have a specification $\sigma(SP)$ with $Sig[\sigma(SP)] = \Sigma'$, $Mod[\sigma(SP)] = \{A' \in \mathbf{Alg}(\Sigma') \mid A'|_\sigma \in Mod[SP]\}$.

hiding: Given a Σ-specification SP and a signature morphism $\sigma\colon \Sigma' \to Sig[SP]$ we also have a specification $SP|_\sigma$ with $Sig[SP|_\sigma] = \Sigma'$, $Mod[SP|_\sigma] = \{A|_\sigma \mid A \in Mod[SP]\}$.

We use specifications of essentially this form to specify the constructions introduced in Sect. 3: a *construction specification* $\mathcal{SP} = \langle \mathcal{S}, SP \rangle$ consists of a construction signature $\mathcal{S}$ and a $\mathbf{Sig}(\mathcal{S})$-specification SP. We write $CSig[\mathcal{SP}]$ for $\mathcal{S}$, and somewhat ambiguously, $Sig[\mathcal{SP}]$ for $\mathbf{Sig}(\mathcal{S})$ and $Mod[\mathcal{SP}]$ for $Mod[SP]$. The definition of the construction models of $\mathcal{SP}$ is somewhat more complex.

Given a construction specification $\mathcal{SP} = \langle \mathcal{S}, SP \rangle$, where $\mathcal{S} = \langle \Sigma, D, \prec \rangle$, we write $CMod[\mathcal{SP}] \subseteq \mathbf{Con}(CSig[\mathcal{SP}])$ for the class of its *construction models*, where an $\mathcal{S}$-construction $\mathcal{C} \subseteq \mathbf{Alg}(\Sigma)$ is a model of $\mathcal{SP}$ if the following conditions hold:

($\mathcal{C}$ is correct for $\mathcal{SP}$): for all $x \in D$ and $A \in \mathcal{C}$, if $A|_{\mathbf{Sig}(\mathcal{S}^{x}\Downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^{x}\Downarrow)}]$ then $A|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)}]$.

($\mathcal{C}$ is complete for $\mathcal{SP}$): for all $x \in \mathbf{Symb}(\Sigma) \setminus D$ and $A \in Mod[SP]$, if $A|_{\mathbf{Sig}(\mathcal{S}^{x}\Downarrow)} \in \mathcal{C}|_{\mathbf{Sig}(\mathcal{S}^{x}\Downarrow)}$ then $A|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)} \in \mathcal{C}|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)}$.

(grouping): $\mathcal{C}$ is well-grouped.

($\mathcal{C}$-dependency-wise): for all $X \subseteq \mathbf{Symb}(\Sigma)$ and $A \in \mathcal{C}$, if $A|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)}]$ for each $x \in X$ then $A|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)}]$.

Somewhat informally, the specification SP is used here to determine both the scope of the construction (the requirements on the assumed symbols under which the construction must work) as well as its results (the requirements on the defined symbols which the construction must ensure). $\mathcal{C}$ is correct for $\mathcal{SP}$ if for each interpretation of the dependency structure below a defined symbol x that is allowed by SP, it interprets the symbol x in a way that satisfies the requirements imposed by SP. $\mathcal{C}$ is complete for $\mathcal{SP}$ if any interpretation of assumed symbols consistent with SP is allowed. The requirement that $\mathcal{C}$ is well-grouped was discussed in Sect. 3: no combination of allowed interpretations of assumed symbols should be excluded. Finally, the "$\mathcal{C}$-dependency-wise" condition states that as far as algebras within the construction are concerned, the requirements imposed by SP must reflect the dependency structure: SP must not directly relate symbol that are mutually independent (although some relationship between such symbols may follow via common symbols in their dependency structures). However, this does not always concern sets of defined symbols, as in the models of the specification their interpretation is uniquely determined by the construction. This remark notwithstanding, in a way, the $\mathcal{C}$-dependency-wise requirement may be seen as constraining specifications rather than constructions.

A construction specification $\mathcal{SP} = \langle \mathcal{S}, SP \rangle$, where $\mathcal{S} = \langle \Sigma, D, \prec \rangle$, is *dependency-wise* if for all $X \subseteq \mathbf{Symb}(\Sigma)$ and $A \in \mathbf{Alg}(\Sigma)$, if $A|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^{x}\downarrow)}]$ for each $x \in X$ then $A|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)}]$.

Lemma 4.1. For any dependency-wise construction specification $\mathcal{SP} = \langle \mathcal{S}, SP \rangle$, any well-grouped $\mathcal{S}$-construction $\mathcal{C} \subseteq \mathbf{Alg}(\mathbf{Sig}(\mathcal{S}))$ that is correct and complete for $\mathcal{SP}$ is its construction model: $\mathcal{C} \in CMod[\mathcal{SP}]$.

Proof. If $\mathcal{SP}$ is dependency-wise then the $\mathcal{C}$-dependency-wise condition holds as well. $\square$

Theorem 4.2. Every dependency-wise construction specification has a construction model.

Proof. Let $\mathcal{SP} = \langle \mathcal{S}, SP \rangle$, where $\mathcal{S} = \langle \Sigma, D, \prec \rangle$, be dependency-wise.

First note that SP is consistent: take any algebra $A \in \mathbf{Alg}(\Sigma)$, then since $\mathcal{SP}$ is dependency-wise, $A|_{\Sigma_\emptyset} \in Mod[SP|_{\Sigma_\emptyset}]$, where $\Sigma_\emptyset$ is the empty algebraic signature, which implies that $Mod[SP] \neq \emptyset$.

Then, let n be the number of symbols in Σ and $x_1, \ldots, x_n$ be an enumeration of $\mathbf{Symb}(\Sigma)$ consistent with $\prec$, that is $\mathbf{Symb}(\Sigma) = \{x_1, \ldots, x_n\}$ and if $x_i \prec x_j$ then $i < j$. Then taking $X_0 = \emptyset$ and $X_i = X_{i-1} \cup \{x_i\}$, $\mathbf{Symb}(\mathcal{S}^{x_i}\Downarrow) \subseteq X_{i-1}$ for $i = 1, \ldots, n$, and $\mathcal{S}^{X_i}\downarrow$ is a (proper) subsignature of $\mathcal{S}^{X_k}\downarrow$ for $0 \leq i < k \leq n$.

Let $\mathcal{C}_0 = Mod[SP]$ and then, for $i = 1, \ldots, n$, let $\mathcal{C}_i$ be defined as follows:

- if x_i is assumed, $\mathcal{C}_i = \mathcal{C}_{i-1}$;
- if x_i is defined, let $\mathcal{C}_i$ be a maximal subset of $\mathcal{C}_{i-1}$ such that for all $A, A' \in \mathcal{C}_i$, if $A|_{\mathbf{Sig}(\mathcal{S}^{x_i}\Downarrow)} = A'|_{\mathbf{Sig}(\mathcal{S}^{x_i}\Downarrow)}$ then $A|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} = A'|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}$. Such a $\mathcal{C}_i$ exists by the Zorn-Kuratowski Lemma (since given any chain of subsets of $\mathcal{C}_{i-1}$ that satisfy the requirement, its union satisfies the requirement as well). Then we have a unique persistent (along the inclusion $\mathcal{S}^{x_i}\Downarrow \hookrightarrow \mathcal{S}^{x_i}\downarrow$) function $F_i \colon \mathcal{C}_{i-1}|_{\mathbf{Sig}(\mathcal{S}^{x_i}\Downarrow)} \to \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}$, which is total by maximality of $\mathcal{C}_i$. Moreover $\mathcal{C}_i = \{A \in \mathcal{C}_{i-1} \mid A|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} = F_i(A|_{\mathbf{Sig}(\mathcal{S}^{x_i}\Downarrow)})\}$. (In fact, choosing $\mathcal{C}_i$ is the same as choosing a total persistent function $F_i \colon \mathcal{C}_{i-1}|_{\mathbf{Sig}(\mathcal{S}^{x_i}\Downarrow)} \to \mathcal{C}_{i-1}|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}$ and then defining $\mathcal{C}_i$ in this way.)

It follows by an easy induction that $\mathcal{C}_i = \{A \in Mod[SP] \mid A|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} \in \mathcal{C}_i\}$ and that $\mathcal{C}_k|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} = \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)}$ for $0 \leq i < k \leq n$.

We show that $\mathcal{C}_n$ is a construction model of $\mathcal{SP}$.

Clearly, $\mathcal{C}_n$ is a construction, and it is correct for $\mathcal{SP}$. By Lemma 4.1, we have to show that it is complete for $\mathcal{SP}$ and is well-grouped.

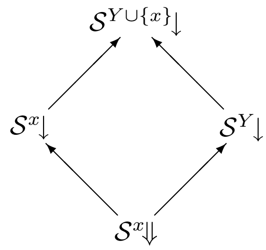
First, we show by induction on $i = 0, \ldots, n$ that $\mathcal{C}_i$ is complete for $\mathcal{SP}$. This is obvious for $i = 0$. Assume $\mathcal{C}_{i-1}$ is complete for $\mathcal{SP}$, $0 < i \leq n$. If x_i is assumed, then $\mathcal{C}_i = \mathcal{C}_{i-1}$ is complete for $\mathcal{SP}$ as well. Thus, the interesting case is when x_i is defined. Let then $x = x_k \in \mathbf{Symb}(\Sigma) \setminus D$ be an assumed symbol, and $A \in Mod[SP]$ be such that $A|_{\mathbf{Sig}(S^x\Downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(S^x\Downarrow)}$. We have to show that $A|_{\mathbf{Sig}(S^x\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(S^x\downarrow)}$.

Since $\mathcal{C}_i \subseteq \mathcal{C}_{i-1}$, we have $A|_{\mathbf{Sig}(S^x\Downarrow)} \in \mathcal{C}_{i-1}|_{\mathbf{Sig}(S^x\Downarrow)}$, and so by the inductive assumption, $A|_{\mathbf{Sig}(S^x\downarrow)} \in \mathcal{C}_{i-1}|_{\mathbf{Sig}(S^x\downarrow)}$,

If $k < i$ then $S^x\downarrow$ is a subsignature of $S^{X_{i-1}}\downarrow$. Then, since $\mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^{X_{i-1}}\downarrow)} = \mathcal{C}_{i-1}|_{\mathbf{Sig}(\mathcal{S}^{X_{i-1}}\downarrow)}$, we get $\mathcal{C}_i|_{\mathbf{Sig}(S^x\downarrow)} = (\mathcal{C}_i|_{\mathcal{S}^{X_{i-1}}\downarrow})|_{\mathbf{Sig}(S^x\downarrow)} = (\mathcal{C}_{i-1}|_{\mathcal{S}^{X_{i-1}}\downarrow})|_{\mathbf{Sig}(S^x\downarrow)} = \mathcal{C}_{i-1}|_{\mathbf{Sig}(S^x\downarrow)}$. Hence $A|_{\mathbf{Sig}(S^x\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(S^x\downarrow)}$.

If $i < k$ and $x_i \prec x$ then $S^{x_i}\downarrow$ is a subsignature of $S^x\Downarrow$. Then since $A|_{\mathbf{Sig}(S^x\Downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(S^x\Downarrow)}$, we have $A|_{\mathbf{Sig}(S^{x_i}\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(S^{x_i}\downarrow)}$, which together with $A \in Mod[SP]$ yields $A \in \mathcal{C}_i$, and $A|_{\mathbf{Sig}(S^x\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(S^x\downarrow)}$.

Finally, consider $i < k$ and $x_i \not\prec x$ (so that x_i and x are independent). Let $Y = \mathbf{Symb}(\mathbf{Sig}(S^x\Downarrow)) \cup \{x_i\}$. Then the following is a pushout in $\mathbf{ConSig}$, with all four morphisms being inclusions:



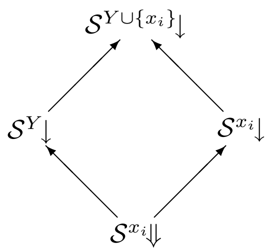
Since $A|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)}$, there is $B \in \mathcal{C}_i$ such that $B|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)} = A|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)}$. Let $B_0 \in \mathbf{Alg}(\mathbf{Sig}(\mathcal{S}^{Y\cup\{x\}}\!\downarrow))$ be the amalgamation of $A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$ and $B|_{\mathbf{Sig}(\mathcal{S}^Y\downarrow)}$ (over the algebraic signature pushout underlying the above construction signature pushout). Let B' be an expansion of B_0 to a $\mathbf{Sig}(\mathcal{S})$)-algebra, that is, $B' \in \mathbf{Alg}(\mathbf{Sig}(\mathcal{S}))$ and $B'|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x\}}\downarrow)} = B_0$.[3] Now, $B'|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} = A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}]$ and $B'|_{\mathbf{Sig}(\mathcal{S}^Y\downarrow)} = B|_{\mathbf{Sig}(\mathcal{S}^Y\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^Y\downarrow)} \subseteq Mod[SP|_{\mathbf{Sig}(\mathcal{S}^Y\downarrow)}]$. Hence, since $\mathcal{SP}$ is dependency-wise, $B_0 = B'|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x\}}\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x\}}\downarrow)}]$. Therefore, there is $A'' \in Mod[SP]$ such that $A''|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x\}}\downarrow)} = B_0$. Since $x_i \in Y$, we also have $A''|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} = B_0|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} = B|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}$. Consequently, $A'' \in \mathcal{C}_i$, and $A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} = B_0|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} = A''|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$, which proves that $\mathcal{C}_i$ is complete for $\mathcal{SP}$, and so in particular $\mathcal{C}_n$ is complete for $\mathcal{SP}$.

To show that $\mathcal{C}_n$ is well-grouped, consider an algebra $A \in \mathbf{Alg}(\Sigma)$ and a set $X \subseteq \mathbf{Symb}(\Sigma)$ such that for each $x \in X$, $A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in \mathcal{C}_n|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$. Since $\mathcal{C}_n \subseteq Mod[SP]$ and $\mathcal{SP}$ is dependency-wise, it follows that $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in Mod[SP|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}]$. Let $A' \in Mod[SP]$ be such that $A'|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} = A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$. We prove that $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$ by induction, $i = 0, \ldots, n$. Clearly, by definition of $\mathcal{C}_0$, $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} = A'|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}_0|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$.

Suppose now $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}_{i-1}|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$, for some $i = 1, \ldots, n$. If x_i is assumed then $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$ since $\mathcal{C}_i = \mathcal{C}_{i-1}$. Otherwise x_i is defined.

If $x_i \prec x$ for some $x \in X$ then $A'|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} = (A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)})|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}$, since $A|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in \mathcal{C}_n|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$ and $\mathcal{C}_n \subseteq \mathcal{C}_i$. Consequently, $A' \in \mathcal{C}_i$, and $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}_i|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$.

Otherwise $x_i \notin \mathbf{Symb}(\mathbf{Sig}(\mathcal{S}^X\!\downarrow))$. Put $Y = \mathbf{Symb}(\mathbf{Sig}(\mathcal{S}^X\!\downarrow) \cup \mathbf{Symb}(\mathbf{Sig}(\mathcal{S}^{x_i}\!\Downarrow))$. The following is a pushout in $\mathbf{ConSig}$, with all four morphisms being inclusions:



Since $A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}_{i-1}|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$, there is $B \in \mathcal{C}_{i-1}$ such that $B|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} = A|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$. By the definition of $\mathcal{C}_i$, there is $B' \in \mathcal{C}_i$ such that $B'|_{\mathcal{S}^{x_i}\Downarrow} = B|_{\mathcal{S}^{x_i}\Downarrow}$. Let $B_0 \in \mathbf{Alg}(\mathbf{Sig}(\mathcal{S}^{Y\cup\{x_i\}}\!\downarrow))$ be the amalgamation of $B|_{\mathbf{Sig}(\mathcal{S}^Y\downarrow)}$ and $B'|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}$ (over the algebraic signature pushout underlying the above construction signature pushout). Let B'' be an expansion of B_0 to a $\mathbf{Sig}(\mathcal{S})$-algebra, that is, $B'' \in \mathbf{Alg}(\mathbf{Sig}(\mathcal{S}))$ and $B''|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x_i\}}\downarrow)} = B_0$. Now, $B''|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} =$

[3] If Σ is an algebraic subsignature of Σ' than any Σ-algebra (with non-empty carriers, as we assume here) has an expansion to a Σ'-algebra.

$B'\big|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} \in \mathcal{C}_i\big|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} \subseteq Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}]$ and $B''\big|_{\mathbf{Sig}(\mathcal{S}^{Y}\downarrow)} = B\big|_{\mathbf{Sig}(\mathcal{S}^{Y}\downarrow)} \in \mathcal{C}_{i-1}\big|_{\mathbf{Sig}(\mathcal{S}^{Y}\downarrow)} \subseteq Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{Y}\downarrow)}]$. Hence, since $\mathcal{SP}$ is dependency-wise, $B_0 = B''\big|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x_i\}}\downarrow)} \in Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x_i\}}\downarrow)}]$. Therefore, there is $A'' \in Mod[SP]$ such that $A''\big|_{\mathbf{Sig}(\mathcal{S}^{Y\cup\{x_i\}}\downarrow)} = B_0$. Then $A''\big|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} = B_0\big|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} = B'\big|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)} \in \mathcal{C}_i\big|_{\mathbf{Sig}(\mathcal{S}^{x_i}\downarrow)}$. Consequently, $A'' \in \mathcal{C}_i$, and $A\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} = B\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} = B_0\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} = A''\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} \in \mathcal{C}_i\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)}$.

This proves that $\mathcal{C}_n$ is well-grouped, and completes the proof of the theorem. □

Perhaps surprisingly, even if $\mathcal{C}$ is a construction model of a construction specification $\mathcal{SP} = \langle \mathcal{S}, SP\rangle$, it does not follow that all algebras in $\mathcal{C}$ are models of SP — we call $\mathcal{C}$ a *clean* construction model of $\mathcal{SP}$ if $\mathcal{C} \subseteq Mod[SP]$. In general though, the idea is that we do not require the construction to "work as specified" when the assumed symbols are interpreted so that they do not satisfy the specification. We may want to "clean up" any construction to cover only the models permitted by the algebraic specification SP, which we write as $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C}) = \mathcal{C} \cap Mod[SP]$. Cleaning a construction model of a construction specification yields a construction model of this construction specification as well, although this is not a self-evident property:

Lemma 4.3. Given a construction specification $\mathcal{SP} = \langle \mathcal{S}, SP\rangle$ and a construction model $\mathcal{C} \in CMod[\mathcal{SP}]$, $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$ is a construction model of $\mathcal{SP}$ as well, $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C}) \in CMod[\mathcal{SP}]$.

Proof. Clearly, since $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C}) \subseteq \mathcal{C}$, if $\mathcal{C}$ is a construction then so is $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$. Moreover, if $\mathcal{C}$ if correct for $\mathcal{SP}$ then so is $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$, and the $\mathcal{C}$-dependency-wise property implies the $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$-dependency-wise property. So, we have to prove that $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$ is complete for $\mathcal{SP}$ and that $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$ is well-grouped.

Let $\mathcal{S} = \langle \Sigma, D, \prec\rangle$. As in the proof of Thm. 4.2, let n be the number of symbols in $\mathbf{Sig}(\mathcal{S})$, $x_1, \dots, x_n$ be an enumeration of $\mathbf{Symb}(\Sigma)$ consistent with $\prec$, $X_0 = \emptyset$ and $X_i = X_{i-1} \cup \{x_i\}$.

First, by induction on $i = n, \dots, 0$ (reverse order!) we prove that $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} \supseteq \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} \cap Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)}]$. Since the opposite inclusion is obvious, we will in fact prove the equality of the two algebra classes in this way.

Since $X_n = \mathbf{Symb}(\Sigma)$, there is nothing to prove for $i = n$. Suppose now for some $i = n-1, \dots, 0$ that $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)} \supseteq \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)} \cap Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)}]$. Consider a $\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)$-algebra $A_i \in \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} \cap Mod[SP]\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)}$. We have then $A' \in \mathcal{C}$ and $A'' \in Mod[SP]$ such that $A'\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} = A_i = A''\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)}$.

If x_{i+1} is defined, put $B_{i+1} = A'\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)} \in \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)}$. Then also $B_{i+1} \in Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)}]$, since $\mathcal{C}$ is correct for $\mathcal{SP}$. If x_{i+1} is assumed, put $B_{i+1} = A''\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)} \in Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)}]$. Then also $B_{i+1} \in \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)}$, since $\mathcal{C}$ is complete for $\mathcal{SP}$. In either case, $B_{i+1} \in \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)} \cap Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)}]$, and so $B_{i+1} \in \mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^{X_{i+1}}\downarrow)}$ by the inductive hypothesis. Hence $A_i = B_{i+1}\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} \in \mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)}$.

This proves $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} = \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)} \cap Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{X_i}\downarrow)}]$, for $i = 0, \dots, n$. In fact, it follows immediately that for any set $X \subseteq \mathbf{Symb}(\Sigma)$ and Σ-algebra A, if $A\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} \in \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)}$ and $A\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} \in Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)}]$ then $A\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)} \in \mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^{X}\downarrow)}$ (just choose the enumeration $x_1, \dots, x_n$ used above so that $\mathbf{Symb}(\mathbf{Sig}(\mathcal{S}^{X}\downarrow)) = X_i$ for some i).

Now, back to the proof of the lemma: to see that $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$ is complete for $\mathcal{SP}$, consider $x \in \mathbf{Symb}(\Sigma) \setminus D$ and $A \in Mod[SP]$ such that $A\big|_{\mathbf{Sig}(\mathcal{S}^{x}\Downarrow)} \in \mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^{x}\Downarrow)}$.

Clearly, $A\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}]$, and since $\mathcal{C}$ is complete for $\mathcal{SP}$, $A\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$. Consequently, by the above fact, $A\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in \mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$.

To see that $\mathbf{clean}_{\mathcal{SP}}(\mathcal{C})$ is well-grouped, consider a set $X \subseteq \mathbf{Symb}(\Sigma)$ and a Σ-algebra A such that $A\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in \mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}$ for all $x \in X$. Since $\mathcal{C}$ is well-grouped, $A\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathcal{C}\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$. Take any Σ-algebra $A' \in \mathcal{C}$ such that $A'\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} = A\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$. Then for all $x \in X$, $A'\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}]$, and so by the $\mathcal{C}$-dependency-wise property, $A\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} = A'\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in Mod[SP\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}]$. Hence $A\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in \mathbf{clean}_{\mathcal{SP}}(\mathcal{C})\big|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$. □

Example 4.4 (Ordinary signatures and algebras). Recall Examples 2.3 and 3.4. Consider a construction specification $\mathcal{SP} = \langle \mathcal{S}, SP \rangle$, where $\mathcal{S}$ is a complete construction signature and SP is a $\mathbf{Sig}(\mathcal{S})$-specification. Then any algebra $A \in Mod[SP]$ yields a clean construction model $\{A\} \in CMod[\mathcal{SP}]$; all clean construction models of $\mathcal{SP}$ are of this form.

Note that in general $\mathcal{SP}$ need not be dependency-wise. However, the $\{A\}$-dependency-wise property trivially holds. □

Example 4.5 (Parameterised modules). Recall Examples 2.4 and 3.5.

A specification of parameterised modules over a "type" $\iota\colon \Sigma_P \hookrightarrow \Sigma_R$ is typically given by a *parameter specification* SP_P with $Sig[SP_P] = \Sigma_P$ and *result specification* SP_R with $Sig[SP_R] = \Sigma_R$, such that $Mod[SP_R\big|_{\Sigma_P}] \subseteq Mod[SP_P]$ (or in other words, $\iota\colon SP_P \to SP_R$ is a specification morphism, see Sect. 5). A parameterised module, which is a persistent partial function $F\colon \mathbf{Alg}(\Sigma_P) \to \mathbf{Alg}(\Sigma_R)$, satisfies a specification so given if for all $A \in Mod[SP_P]$, $F(A)$ is defined and $F(A) \in Mod[SP_R]$. Clearly, such a correct parameterised module exists only if SP_R is a *conservative extension* of SP_P, i.e., each model in $Mod[SP_P]$ may be expanded to a model in $Mod[SP_P]$, or in other words: $Mod[SP_R\big|_{\Sigma_P}] = Mod[SP_P]$ — we assume below that this is the case.

Consider a construction specification $\mathcal{SP}_\iota = \langle \mathcal{S}_\iota, SP_R \rangle$, with $SP_P = SP_R\big|_{\Sigma_P}$, where $\mathcal{S}_\iota$ is the construction signature determined by ι as defined in Example 2.4.

Let $\mathcal{C} \subseteq \mathbf{Alg}(\Sigma_R)$ be a construction model of $\mathcal{SP}_\iota$, and let $F_\mathcal{C} = \{A_P \mapsto A_R \mid A_R \in \mathcal{C}, A_P = A_R\big|_{\Sigma_P}\}$ be the parameterised module it defines, as in Example 3.5. Then, since $\mathcal{C}$ is complete for $\mathcal{SP}_\iota$, and all the symbols in Σ_P are assumed in $\mathcal{S}_\iota$, $\mathcal{C}\big|_{\Sigma_P} \supseteq Mod[SP_P]$, so $F_\mathcal{C}$ is defined on all algebras in $Mod[SP_P]$. Moreover, since all symbols in Σ_R that are not in Σ_P are defined in $\mathcal{S}_\iota$, each $A_P \in Mod[SP_P]$ has a unique expansion to a Σ_R-algebra in $\mathcal{C}$, and this expansion is a model in $Mod[SP_R]$. Hence, $F_\mathcal{C}$ is indeed correct w.r.t. parameter specification SP_P and result specification SP_R.

Conversely, consider a persistent partial function $F\colon \mathbf{Alg}(\Sigma_P) \to \mathbf{Alg}(\Sigma_R)$ that is correct w.r.t. parameter specification SP_P and result specification SP_R, and define its corresponding $\mathcal{S}_\iota$-construction $\mathcal{C}_F = \{F(A) \mid A \in \mathbf{Alg}(\Sigma_P)\}$, as in Example 3.5. Unfortunately, in general $\mathcal{C}_F$ need not be a construction model of $\mathcal{SP}_\iota$. A minor problem is that $\mathcal{C}_F$ need not be well-grouped: we know nothing about how F works on algebras outside $Mod[SP_P]$, and so its domain need not be closed under arbitrary combination of interpretations of symbols in the parameter signature allowed in some algebras in its domain. No harm is done though by cleaning $\mathcal{C}_F$ to leave models of SP_R only, thus removing applications of F to algebras not in $Mod[SP_P]$ — or alternatively, by adding any Σ_R-expansion as a result for any Σ_P-algebra needed to make the domain of F (and hence also $\mathcal{C}_F$) well-grouped.

A more serious problem is that the $\mathcal{C}_F$-dependency-wise condition may not hold for sets of symbols in the parameter signature. A remedy is to add to $\mathcal{S}_\iota$ enough dependencies between the symbols in Σ_P to make SP_P dependency-wise. This can always be achieved, for instance by imposing any linear strict order on the symbols of Σ_P. Once $\mathcal{S}_\iota$ is so redefined, call it $\mathcal{S}'_\iota$, $\mathcal{C}'_F = \{F(A) \mid A \in Mod[SP_P]\}$ is a (clean) construction model of the construction specification $\mathcal{SP}'_\iota = \langle \mathcal{S}'_\iota, SP_R \rangle$. □

Example 4.6 (Complex dependencies). Recall Example 2.5, the construction signature $\mathcal{S}_0$ defined there, and Example 3.6, with sample Σ_0-algebras $A_1, \ldots, A_5$, and the $\mathcal{S}_0$-construction $\mathcal{C}_0 = \{A_1, A_3, A_4, A_5\}$.

Suppose we have a Σ_0-specification SP_N with all models interpreting Nat, $zero\colon Nat$ and $succ\colon Nat$ in the standard way, and putting no constraints on the other constants in the signature. Consider extensions of SP_N by axioms that constrain the other constants (such extensions, originating from CLEAR [BG81], may be defined easily as the union of SP_N with basic specifications listing the corresponding axioms, see CASL [BM04] or [ST12]).

$$SP_1 = SP_N \textbf{ then } a = zero$$
$$SP_2 = SP_N \textbf{ then } b = c$$
$$SP_3 = SP_N \textbf{ then } d = succ(b) \wedge e = succ(c)$$
$$SP_4 = SP_N \textbf{ then } (b = succ(a) \vee b = succ(succ(a))) \wedge$$
$$(c = succ(succ(a)) \vee c = succ(succ(succ(a)))) \wedge d = succ(b) \wedge e = succ(c)$$
$$SP_5 = SP_N \textbf{ then } b = succ(succ(a)) \wedge c = succ(succ(a)) \wedge d = e$$
$$SP_6 = SP_N \textbf{ then } (b = succ(a) \vee b = succ(succ(a))) \wedge c = succ(succ(a)) \wedge d = e$$

Put $\mathcal{SP}_i = \langle \mathcal{S}_0, SP_i \rangle$, $i = 1, \ldots, 6$.

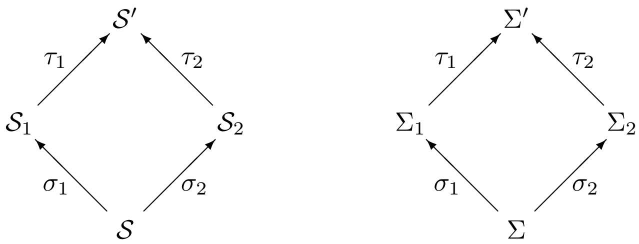
$\mathcal{SP}_1$ is a dependency-wise construction specification, but clearly, $\mathcal{C}_0$ is not its construction model (since it defines a inconsistently with SP_1). $\mathcal{SP}_2$ is not dependency-wise (since for each n there is a model of SP_2 with $b = n$ and another one with $c \neq n$, but no model of SP_2 with both $b = n$ and $c \neq n$) and in fact $\mathcal{SP}_2$ has no construction model. $\mathcal{SP}_3$ is a dependency-wise specification, but $\mathcal{C}_0$ is not its construction model (since $\mathcal{C}_0$ is not complete for $\mathcal{SP}_3$). $\mathcal{SP}_4$ is a dependency-wise specification, and $\mathcal{C}_0$ is among its (clean) construction models. $\mathcal{SP}_5$ is not a dependency-wise specification (since for each n there is a model of SP_5 with $d = n$ and another one with $e \neq n$, but no model of SP_5 has both $d = n$ and $e \neq n$); nevertheless $\mathcal{C}_0$ is its construction model (it is not clean though). $\mathcal{SP}_6$ is another construction specification which is not dependency-wise but it has some construction models; however, $\mathcal{C}_0$ is not among them. □

5 Putting construction specifications together

In the usual algebraic specification framework, one defines a category **Spec** of specifications, where a specification morphism $\sigma\colon SP \to SP'$ is a signature morphism $\sigma\colon Sig[SP] \to Sig[SP']$ such that for all models $A' \in Mod[SP']$, $A'|_\sigma \in Mod[SP]$, see [ST88, ST12]. Then the obvious projection functor from **Spec** to **AlgSig** lifts colimits: given a (finite) diagram of specifications, to build its colimit in **Spec** one first constructs a colimit of its underlying signature diagram in **AlgSig**, and then the colimit specification in **Spec** is built over the colimit signature as the union of translations of the specifications in the diagram along the corresponding signature morphisms of the colimiting cocone in **AlgSig** (this has its roots in [BG77, GB92], see [ST88, ST12]). Colimits in the category of specifications, constructed in this way, are often viewed as a basic way to combine specifications [BG77, BG81].

One may want to take a similar approach here, for putting together construction specifications. However, problems are encountered already with the basic definition of a morphism between construction specifications. To give a hint of the problems, consider two construction specifications $\mathcal{SP} = \langle \mathcal{S}, SP \rangle$ and $\mathcal{SP}' = \langle \mathcal{S}', SP' \rangle$, and a construction signature morphism $\sigma \colon \mathcal{S} \to \mathcal{S}'$ that is also a specification morphism $\sigma \colon SP \to SP'$. Let $\mathcal{C}' \in CMod[\mathcal{SP}']$ be a construction model of $\mathcal{SP}'$. Then in general $\mathcal{C}'|_{\sigma}$, which is an $\mathcal{S}$-construction by Lemma 3.1, is *not* a construction model of $\mathcal{SP}$. This can be seen for instance when the two signatures coincide, σ is the identity, and $Mod[SP] \supseteq Mod[SP']$. The key problem is that weaker requirements in SP concern not only defined, but also assumed symbols in the construction signature, which makes some requirements concerning construction models of construction specifications (e.g., the completeness condition) more difficult to meet. See Sect. 6 for some further remarks on this topic.

Instead, we try to mimic this technique directly and provide compositionality results to justify its usefulness. As in Sect. 3, consider a pushout of construction signatures in **ConSig** and its underlying pushout in **AlgSig** of the following form:



Now, given construction specifications $\mathcal{SP}_1 = \langle \mathcal{S}_1, SP_1 \rangle$ and $\mathcal{SP}_2 = \langle \mathcal{S}_2, SP_2 \rangle$, where SP_1 and SP_2 are specifications with $Sig[SP_1] = \Sigma_1$ and $Sig[SP_2] = \Sigma_2$, we may attempt to put them together to form an $\mathcal{S}'$-construction specification of the form $\langle \mathcal{S}', \tau_1(SP_1) \cup \tau_2(SP_2) \rangle$. However, not all such combinations make methodological sense, and certainly not all of them lead to consistent specifications, even if both $\mathcal{SP}_1$ and $\mathcal{SP}_2$ have construction models. First, no shared symbol should be defined simultaneously in $\mathcal{S}_1$ and $\mathcal{S}_2$. Second, requirements concerning shared symbols in SP_1 and SP_2 must be compatible.

A span $\mathcal{S}_1 \xleftarrow{\sigma_1} \mathcal{S} \xrightarrow{\sigma_2} \mathcal{S}_2$ is a *fitting* between $\mathcal{S}_1$ and $\mathcal{S}_2$ if for each symbol $x \in \mathbf{Symb}(\Sigma)$, if $\sigma_1(x)$ is defined in $\mathcal{S}_1$ then $\sigma_2(x)$ is assumed in $\mathcal{S}_2$, and vice versa, if $\sigma_2(x)$ is defined in $\mathcal{S}_2$ then $\sigma_1(x)$ is assumed in $\mathcal{S}_1$. It follows that the construction signature $\mathcal{S}$ is empty (that is, all its symbols are assumed).[4]

Given construction specifications $\mathcal{SP}_1 = \langle \mathcal{S}_1, SP_1 \rangle$ and $\mathcal{SP}_2 = \langle \mathcal{S}_2, SP_2 \rangle$ and a fitting $ft = \mathcal{S}_1 \xleftarrow{\sigma_1} \mathcal{S} \xrightarrow{\sigma_2} \mathcal{S}_2$ between their signatures, their *sum* $\mathcal{SP}_1 \oplus_{ft} \mathcal{SP}_2$ is a construction specification defined as $\mathcal{SP}_1 \oplus_{ft} \mathcal{SP}_2 = \langle \mathcal{S}', \tau_1(SP_1) \cup \tau_2(SP_2) \rangle$, where $\tau_1 \colon \mathcal{S}_1 \to \mathcal{S}'$ and $\tau_2 \colon \mathcal{S}_2 \to \mathcal{S}'$ form a pushout of σ_1 and σ_2 as above (the subscript *ft* will be omitted when the fitting and hence the pushout are clear).

Still, in general the sum $\mathcal{SP}_1 \oplus_{ft} \mathcal{SP}_2$ may be inconsistent, even if each of $\mathcal{SP}_1$ and $\mathcal{SP}_2$ is consistent on its own, since they may impose incompatible requirements on the shared symbols.

[4] This assumption may be dropped at the expense of complications in the exact formulation and proof of the compatibility result.

We say that construction specifications $\mathcal{SP}_1 = \langle \mathcal{S}_1, SP_1 \rangle$ and $\mathcal{SP}_2 = \langle \mathcal{S}_2, SP_2 \rangle$ are *compatible* w.r.t. the fitting $ft = \mathcal{S}_1 \xleftarrow{\sigma_1} \mathcal{S} \xrightarrow{\sigma_2} \mathcal{S}_2$ if:

- for all sets $X \subseteq \mathbf{Symb}(\Sigma)$ of mutually independent symbols in $\mathcal{S}$ such that $\sigma_1(x)$ is assumed in $\mathcal{S}_1$ for all $x \in X$, and for all $A_2 \in Mod[SP_2]$, if $(A_2|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^X\Downarrow)} \in Mod[SP_1|_{\sigma_1}]|_{\mathbf{Sig}(\mathcal{S}^X\Downarrow)}$ then $(A_2|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in Mod[SP_1|_{\sigma_1}]|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$; and
- (vice versa) for all sets $X \subseteq \mathbf{Symb}(\Sigma)$ of mutually independent symbols in $\mathcal{S}$ such that $\sigma_2(x)$ is assumed in $\mathcal{S}_2$ for all $x \in X$, and for all $A_1 \in Mod[SP_1]$, if $(A_1|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^X\Downarrow)} \in Mod[SP_2|_{\sigma_2}]|_{\mathbf{Sig}(\mathcal{S}^X\Downarrow)}$ then $(A_1|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)} \in Mod[SP_2|_{\sigma_2}]|_{\mathbf{Sig}(\mathcal{S}^X\downarrow)}$.

Informally the above requirements state that neither of the specifications excludes an interpretation of shared symbols that is permitted by the other specification.

Theorem 5.1. Consider construction specifications $\mathcal{SP}_1 = \langle \mathcal{S}_1, SP_1 \rangle$ and $\mathcal{SP}_2 = \langle \mathcal{S}_2, SP_2 \rangle$ that are compatible w.r.t. a fitting $ft = \mathcal{S}_1 \xleftarrow{\sigma_1} \mathcal{S} \xrightarrow{\sigma_2} \mathcal{S}_2$. Let $\mathcal{C}_1 \in CMod[\mathcal{SP}_1]$ and $\mathcal{C}_2 \in CMod[\mathcal{SP}_2]$ be their clean models. Then $\mathcal{C}_1 \oplus \mathcal{C}_2$ is a clean model of $\mathcal{SP}_1 \oplus \mathcal{SP}_2$.

Proof. Let $\tau_1 \colon \mathcal{S}_1 \to \mathcal{S}'$ and $\tau_2 \colon \mathcal{S}_2 \to \mathcal{S}'$ form a pushout of σ_1 and σ_2 as above. Recall that $\mathcal{SP}_1 \oplus \mathcal{SP}_2 = \langle \mathcal{S}', SP' \rangle$, where $SP' = \tau_1(SP_1) \cup \tau_2(SP_2)$. Put $\mathcal{C}' = \mathcal{C}_1 \oplus \mathcal{C}_2$, $\Sigma' = \mathbf{Sig}(\mathcal{S}')$, $\Sigma_1 = \mathbf{Sig}(\mathcal{S}_1)$ and $\Sigma_2 = \mathbf{Sig}(\mathcal{S}_2)$.

$\mathcal{C}'$ is an $\mathcal{S}'$-construction by Lemma 3.2. By definition of $\mathcal{C}_1 \oplus \mathcal{C}_2$, for each $A' \in \mathcal{C}'$, $A'|_{\tau_1} \in \mathcal{C}_1$ and $A'|_{\tau_2} \in \mathcal{C}_2$. Hence, since $\mathcal{C}_1$ and $\mathcal{C}_2$ are clean construction models of $\mathcal{SP}_1$ and $\mathcal{SP}_2$, respectively, $\mathcal{C}' \subseteq Mod[SP']$, and so if $\mathcal{C}'$ is a construction model of $\mathcal{SP}_1 \oplus \mathcal{SP}_2$ then it is its clean construction model.

We have to show that $\mathcal{C}'$ is indeed a construction model of $\mathcal{SP}_1 \oplus \mathcal{SP}_2$.

To show that $\mathcal{C}'$ is correct for $\mathcal{SP}_1 \oplus \mathcal{SP}_2$, consider $x \in \mathbf{Symb}(\Sigma')$ defined in $\mathcal{S}'$ and a Σ'-algebra $A' \in \mathcal{C}'$ such that $A'|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)} \in Mod[SP'|_{\mathbf{Sig}(\mathcal{S}^x\Downarrow)}]$. In fact, since $\mathcal{C}' \subseteq Mod[SP']$, the latter requirement follows from $A' \in \mathcal{C}'$; but so does the required conclusion that $A'|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)} \in Mod[SP'|_{\mathbf{Sig}(\mathcal{S}^x\downarrow)}]$.

A similar argument as above for the correctness condition, using the fact that $\mathcal{C}' \subseteq Mod[SP']$, easily yields the $\mathcal{C}'$-dependency-wise condition.

To show that $\mathcal{C}'$ is complete for $\mathcal{SP}_1 \oplus \mathcal{SP}_2$ and is well-grouped, we will use an auxiliary lemma:

Lemma 5.2. Under the notation introduced above, consider any $X \subseteq \mathbf{Symb}(\Sigma')$ and let $X_1 = \{x_1 \in \mathbf{Symb}(\Sigma_1) \mid \tau_1(x_1) \in X\}$ and $X_2 = \{x_2 \in \mathbf{Symb}(\Sigma_2) \mid \tau_2(x_2) \in X\}$. Then for all Σ'-algebras $A' \in \mathbf{Alg}(\Sigma')$, if $(A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)}$ and $(A'|_{\tau_2})|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)} \in \mathcal{C}_2|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)}$ then $A'|_{\mathbf{Sig}((\mathcal{S}')^X\downarrow)} \in \mathcal{C}'|_{\mathbf{Sig}((\mathcal{S}')^X\downarrow)}$.

Proof. Given a Σ'-algebra A' satisfying the assumptions in the statement of the lemma, we need an algebra $B' \in \mathcal{C}'$ such that $B'|_{\mathbf{Sig}((\mathcal{S}')^X\downarrow)} = A'|_{\mathbf{Sig}((\mathcal{S}')^X\downarrow)}$. Since $X = \tau_1(X_1) \cup \tau_2(X_2)$, by construction of $\mathcal{C}'$ it is enough to show that there exist $B_1 \in \mathcal{C}_1$ and $B_2 \in \mathcal{C}_2$ such that $B_1|_{\sigma_1} = B_2|_{\sigma_2}$ and $B_1|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)} = (A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)}$ and $B_2|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)} = (A'|_{\tau_2})|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)}$; then $B' = B_1 \oplus B_2 \in \mathcal{C}'$ satisfies the above requirement.

Let $Y = \{x \in \mathbf{Symb}(\Sigma) \mid \sigma_1(x) \in X_1\}$. Then also $Y = \{x \in \mathbf{Symb}(\Sigma) \mid \sigma_2(x) \in X_2\} = \{x \in \mathbf{Symb}(\Sigma) \mid \tau_1(\sigma_1(x)) = \tau_2(\sigma_2(x)) \in X\}$. Let then, similarly as in the proof of Thm. 4.2, $x_1, \ldots, x_n$ be an enumeration of $\mathbf{Symb}(\Sigma) \setminus Y$ consistent with the dependencies in $\mathcal{S}$. Put $Y^0 = Y$ and $Y^i = Y^{i-1} \cup \{x_i\}$ for $i = 1, \ldots, n$, and $X_1^i = \sigma_1(Y^i)$ and

$X_2^i = \sigma_2(Y^i)$. Clearly, $Y^n = \mathbf{Symb}(\Sigma)$. By Lemma 2.2, $\mathbf{Sig}(\mathcal{S}_1^{X_1^i}{\downarrow}) = \sigma_1(\mathbf{Sig}(\mathcal{S}^{Y^i}{\downarrow}))$, $\mathbf{Sig}(\mathcal{S}_1^{\sigma_1(x_i)}{\Downarrow}) = \sigma_1(\mathbf{Sig}(\mathcal{S}^{x_i}{\Downarrow}))$, and $\mathbf{Sig}(\mathcal{S}_1^{\sigma_1(x_i)}{\downarrow}) = \sigma_1(\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow}))$.

By induction on $i = 0, \ldots, n$, we construct algebras $B_1^i \in \mathcal{C}_1$ and $B_2^i \in \mathcal{C}_2$ such that $(B_1^i|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^{Y^i}{\downarrow})} = (B_2^i|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{Y^i}{\downarrow})}$ and $B_1^i|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})} = (A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})}$ and $B_2^i|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}{\downarrow})} = (A'|_{\tau_2})|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}{\downarrow})}$. Once this is done, putting $B_1 = B_1^n$ and $B_2 = B_2^n$ will complete the proof of the lemma.

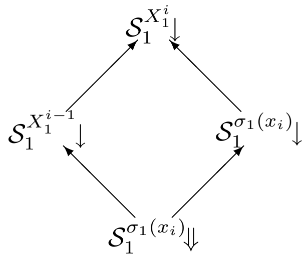
Let $B_1^0 \in \mathcal{C}_1$ be such that $B_1^0|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})} = (A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})}$ (such B_1^0 exists, since $(A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})}$). Similarly, let $B_2^0 \in \mathcal{C}_2$ be such that $B_2^0|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}{\downarrow})} = (A'|_{\tau_2})|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}{\downarrow})}$. Clearly, B_1^0 and B_2^0 satisfy the inductive hypothesis for $i = 0$.

Now, for the induction step, let for some $i = 1, \ldots, n$, $B_1^{i-1} \in \mathcal{C}_1$ and $B_2^{i-1} \in \mathcal{C}_2$ satisfy the inductive hypothesis.

By our assumptions, either $\sigma_1(x_i) \in \mathbf{Symb}(\Sigma_1)$ is assumed in $\mathcal{S}_1$ or $\sigma_2(x_i) \in \mathbf{Symb}(\Sigma_2)$ is assumed in $\mathcal{S}_2$. Suppose $\sigma_1(x_i) \in \mathbf{Symb}(\Sigma_1)$ is assumed in $\mathcal{S}_1$ — the other case is symmetric and we omit it here.

Put $B_2^i = B_2^{i-1}$. Since $\mathcal{C}_2$ is a clean model of $\mathcal{SP}_2$, $B_2^i \in \mathit{Mod}[SP_2]$. Moreover, since $\mathbf{Symb}(\mathbf{Sig}(\mathcal{S}^{x_i}{\Downarrow})) \subseteq Y^{i-1}$ and $\mathcal{C}_1$ is a clean model of $\mathcal{SP}_1$, we have $(B_2^i|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\Downarrow})} = (B_1^{i-1}|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\Downarrow})} \in \mathit{Mod}[SP_1|_{\sigma_1}]|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\Downarrow})}$. By compatibility of $\mathcal{SP}_1$ and $\mathcal{SP}_2$ w.r.t. the fitting considered, $(B_2^i|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow})} \in \mathit{Mod}[SP_1|_{\sigma_1}]|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow})}$. Let $A_1^i \in \mathit{Mod}[SP_1]$ be such that $(A_1^i|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow})} = (B_2^i|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow})}$.

The following is a pushout in $\mathbf{ConSig}$, with all four morphisms being inclusions:



Let $B_1'' \in \mathbf{Alg}(\Sigma_1)$ be an expansion of the amalgamation (over the algebraic signature pushout underlying the above construction signature pushout) of $B_1^{i-1}|_{\mathcal{S}_1^{X_1^{i-1}}{\downarrow}}$ and $A_1^i|_{\mathcal{S}^{\sigma_1(x_i)}{\downarrow}}$.

Then, for each $x \in \mathbf{Symb}(\mathbf{Sig}(\mathcal{S}_1^{X_1^i}{\downarrow}))$, either $x \in \mathbf{Symb}(\mathbf{Sig}(\mathcal{S}_1^{X_1^{i-1}}{\downarrow}))$ and so $B_1''|_{\mathbf{Sig}(\mathcal{S}_1^x{\downarrow})} = B_1^{i-1}|_{\mathbf{Sig}(\mathcal{S}_1^x{\downarrow})} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^x{\downarrow})}$, or $x \in \mathbf{Symb}(\mathbf{Sig}(\mathcal{S}_1^{\sigma_1(x_i)}{\downarrow}))$ and so $B_1''|_{\mathbf{Sig}(\mathcal{S}_1^x{\downarrow})} = A_1^i|_{\mathbf{Sig}(\mathcal{S}_1^x{\downarrow})} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^x{\downarrow})}$. Consequently, since $\mathcal{C}_1$ is well-grouped, $B_1''|_{\mathbf{Sig}(\mathcal{S}_1^{X_1^i}{\downarrow})} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{X_1^i}{\downarrow})}$. Therefore, there is $B_1^i \in \mathcal{C}_1$ such that $B_1^i|_{\mathbf{Sig}(\mathcal{S}_1^{X_1^i}{\downarrow})} = B_1''|_{\mathbf{Sig}(\mathcal{S}_1^{X_1^i}{\downarrow})}$. We have $(B_1^i|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^{Y^{i-1}}{\downarrow})} = (B_1^{i-1}|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^{Y^{i-1}}{\downarrow})} = (B_2^{i-1}|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{Y^{i-1}}{\downarrow})} = (B_2^i|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{Y^{i-1}}{\downarrow})}$ and $(B_1^i|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow})} = (A_1^i|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow})} = (B_2^i|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{x_i}{\downarrow})}$.

It follows now easily that the pair of algebras $B_1^i \in \mathcal{C}_1$ and $B_2^i \in \mathcal{C}_2$ satisfies the requirements: $(B_1^i|_{\sigma_1})|_{\mathbf{Sig}(\mathcal{S}^{Y^i}{\downarrow})} = (B_2^i|_{\sigma_2})|_{\mathbf{Sig}(\mathcal{S}^{Y^i}{\downarrow})}$ and $B_1^i|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})} = (A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}{\downarrow})}$ and $B_2^i|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}{\downarrow})} = (A'|_{\tau_2})|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}{\downarrow})}$.

This completes the proof of Lemma 5.2. □ (Lemma)

Now, back to the proof of the main theorem.

To show that $\mathcal{C}'$ is complete for $\mathcal{SP}'$, let x be an assumed symbol in $\mathcal{S}'$ and $A' \in Mod[SP']$ be such that $A'|_{\mathbf{Sig}((\mathcal{S}')^{x}\Downarrow)} \in \mathcal{C}'|_{\mathbf{Sig}((\mathcal{S}')^{x}\Downarrow)}$. Let $X_1 = \{x_1 \in \mathbf{Symb}(\Sigma_1) \mid \tau_1(x_1) = x\}$ and $X_2 = \{x_2 \in \mathbf{Symb}(\Sigma_2) \mid \tau_2(x_2) = x\}$. Then $A'|_{\tau_1} \in Mod[SP_1]$, and for all $x_1 \in X_1$, $(A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\Downarrow)} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\Downarrow)}$. Since x is assumed in $\mathcal{S}'$, all $x_1 \in X_1$ are assumed in $\mathcal{S}_1$. Hence, since $\mathcal{C}_1$ is complete for $\mathcal{SP}_1$, $(A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\downarrow)} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\downarrow)}$ for all $x_1 \in X_1$. Now, since $\mathcal{C}_1$ is well-grouped, $(A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)}$. Similarly we get that $(A'|_{\tau_2})|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)} \in \mathcal{C}_2|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)}$. Thus, by Lemma 5.2, $A'|_{\mathbf{Sig}((\mathcal{S}')^{x}\downarrow)} \in \mathcal{C}'|_{\mathbf{Sig}((\mathcal{S}')^{x}\downarrow)}$ as required.

To see that $\mathcal{C}'$ is well-grouped, consider $X \subseteq \mathbf{Symb}(\Sigma')$ and let $A' \in \mathbf{Alg}(\Sigma')$ be such that for all $x \in X$, $A'|_{\mathbf{Sig}((\mathcal{S}')^{x}\downarrow)} \in \mathcal{C}'|_{\mathbf{Sig}((\mathcal{S}')^{x}\downarrow)}$. Let again $X_1 = \{x_1 \in \mathbf{Symb}(\Sigma_1) \mid \tau_1(x_1) \in X\}$ and $X_2 = \{x_2 \in \mathbf{Symb}(\Sigma_2) \mid \tau_2(x_2) \in X\}$. Then for all $x_1 \in X_1$, $(A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\downarrow)} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{x_1}\downarrow)}$, and since $\mathcal{C}_1$ is well-grouped, $(A'|_{\tau_1})|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)} \in \mathcal{C}_1|_{\mathbf{Sig}(\mathcal{S}_1^{X_1}\downarrow)}$. Similarly we get that $(A'|_{\tau_2})|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)} \in \mathcal{C}_2|_{\mathbf{Sig}(\mathcal{S}_2^{X_2}\downarrow)}$. Thus, by Lemma 5.2, $A'|_{\mathbf{Sig}((\mathcal{S}')^{X}\downarrow)} \in \mathcal{C}'|_{\mathbf{Sig}((\mathcal{S}')^{X}\downarrow)}$ as required. □

Example 5.3 (Ordinary signatures and algebras). Recall Examples 2.3, 3.4 and 4.4. Consider construction specifications $\mathcal{SP}_1 = \langle \mathcal{S}_1, SP_1 \rangle$ and $\mathcal{SP}_2 = \langle \mathcal{S}_2, SP_2 \rangle$, where $\mathcal{S}_1$ and $\mathcal{S}_2$ are complete construction signatures, SP_1 is a $\mathbf{Sig}(\mathcal{S}_1)$-specification and SP_2 is a $\mathbf{Sig}(\mathcal{S}_2)$-specification. The only fitting between $\mathcal{S}_1$ and $\mathcal{S}_2$ is $\mathcal{S}_1 \xleftarrow{\iota_1} \mathcal{S}_\emptyset \xrightarrow{\iota_2} \mathcal{S}_2$, where $\mathcal{S}_\emptyset$ is the entirely empty signature with no symbols at all. Construction specifications $\mathcal{SP}_1$ and $\mathcal{SP}_2$ are compatible w.r.t. such fitting whenever SP_1 and SP_2 are consistent.

The pushout of ι_1 and ι_2 is the disjoint union of $\mathcal{S}_1$ and $\mathcal{S}_2$. Given two algebras $A_1 \in Mod[SP_1]$ and $A_2 \in Mod[SP_2]$, $\{A_1\}$ is a construction model of $\mathcal{SP}_1$ and $\{A_2\}$ is a construction model of $\mathcal{SP}_2$, as explained in Example 3.4. Then the sum $\{A_1\} \oplus \{A_2\}$ is a construction model of $\mathcal{SP}_1 \oplus \mathcal{SP}_2$. It consists of a single algebra which is essentially a disjoint union of A_1 and A_2. □

Example 5.4 (Parameterised modules). Recall Examples 2.4, 3.5 and 4.5. Consider a "type" $\iota\colon \Sigma_P \hookrightarrow \Sigma_R$ for parameterised modules, Σ_P-specification SP_P and Σ_R-specification SP_R with $Mod[SP_R|_{\Sigma_P}] = Mod[SP_P]$. Let $\mathcal{SP}'_\iota = \langle \mathcal{S}'_\iota, SP_R \rangle$ be a corresponding dependency-wise construction specification, as explained in Example 4.5. Put $\mathcal{S}'_P = (\mathcal{S}')^{\mathbf{Symb}(\Sigma_P)}\downarrow = \langle \Sigma_P, \emptyset, \prec_P \rangle$ for some dependency relation $\prec_P$; then $\iota\colon \mathcal{S}'_P \to \mathcal{S}'_\iota$ is a construction signature inclusion.

Consider any construction specification $\mathcal{SP}_A = \langle \mathcal{S}_A, SP_A \rangle$, where $\mathcal{S}_A$ is a total construction signature that shares with $\mathcal{S}'_P$ the algebraic signature and the dependency relation, and SP_A is a Σ_P-specification. Then the identity morphism id_{Σ_P} on Σ_P is a construction signature morphism from $\mathcal{S}'_P$ to $\mathcal{S}_A$. Moreover, this yields a fitting $\mathcal{S}_A \xleftarrow{id_{\Sigma_P}} \mathcal{S}'_P \xrightarrow{\iota} \mathcal{S}'_\iota$ between $\mathcal{S}_A$ and $\mathcal{S}'_\iota$. If $Mod[SP_A] \subseteq Mod[SP_P]$ then construction specifications $\mathcal{SP}_A$ and $\mathcal{SP}'_\iota$ are compatible w.r.t. this fitting. We may define their sum explicitly: $\mathcal{SP}_A \oplus \mathcal{SP}'_\iota = \langle \mathcal{S}'_\iota, SP_R \cup \iota(SP_A) \rangle$.

Consider now a clean construction model $\mathcal{C}$ of $\mathcal{SP}'_\iota$. The parameterised module $F_\mathcal{C}\colon \mathbf{Alg}(\Sigma_P) \to \mathbf{Alg}(\Sigma_R)$ it defines, $F_\mathcal{C} = \{A_P \mapsto A_R \mid A_R \in \mathcal{C}, A_P = A_R|_{\Sigma_P}\}$, is correct w.r.t. parameter specification SP_P and result specification SP_R, see Example 4.5. Therefore, given $A \in Mod[SP_A] \subseteq Mod[SP_P]$, $F_\mathcal{C}(A)$ is defined and, since $F_\mathcal{C}$ is persistent, $F_\mathcal{C}(A) \in Mod[SP_R \cup \iota(SP_A)]$. This is also captured at the level of construction models: $\{A\}$ is a construction model of $\mathcal{SP}_A$, and its amalgamation with $\mathcal{C}$ yields $\{A\} \oplus \mathcal{C} = \{F_{\mathcal{C}}(A)\}$, which is a clean construction model of $\mathcal{SP}_A \oplus \mathcal{SP}'_\iota$ by Thm. 5.1.

More generally, we may mimic the standard pushout approach to parameterisation, cf. [BG77, EM85, ST12]. Let $\mathcal{SP}_A = \langle \mathcal{S}_A, SP_A \rangle$ be a construction specification with total construction signature $\mathcal{S}_A$, and let $\varphi\colon \mathcal{S}'_P \to \mathcal{S}_A$ be a *fitting morphism*. Suppose $Mod[SP_A|_\varphi] \subseteq Mod[SP_P]$. Then construction specifications $\mathcal{SP}_A$ and $\mathcal{SP}'_\iota$ are compatible w.r.t. the fitting $\mathcal{S}_A \xleftarrow{\varphi} \mathcal{S}'_P \xrightarrow{\iota} \mathcal{S}'_\iota$. Moreover, given $A \in Mod[SP_A]$ (so that $\{A\}$ is a construction model of $\mathcal{SP}_A$) and a clean construction model $\mathcal{C}$ of $\mathcal{SP}'_\iota$, the amalgamation $\{A\} \oplus \mathcal{C} = \{A'\}$, where A' amalgamates A and $F_{\mathcal{C}}(A|_\varphi)$, is a clean construction model of $\mathcal{SP}_A \oplus \mathcal{SP}'_\iota$.

In a similar way we may mimic other typical operations on parameterised modules, for instance various forms of partial application of a parameterised module to a "part" of its required argument via a fitting morphism from a subsignature of the parameter signature, etc.

□

Example 5.5 (Complex dependencies). Recall Examples 2.5, 3.6 and 4.6, and the construction signature $\mathcal{S}_0$ with $\mathbf{Sig}(\mathcal{S}_0)$-algebras $A_1, \ldots, A_5$, the $\mathcal{S}_0$-construction $\mathcal{C}_0$ and construction specifications $SP_1, \ldots, SP_6$ with construction signature $\mathcal{S}_0$.

Let $\mathcal{S}'_0 = \mathcal{S}_0^d\!\downarrow$ (so that $\Sigma'_0 = \mathbf{Sig}(\mathcal{S}'_0)$ contains all symbols in $\mathcal{S}_0$ except for $e\colon Nat$, with definedness and dependencies in $\mathcal{S}'_0$ inherited from $\mathcal{S}_0$) and let $\mathcal{S}''_0$ be as $\mathcal{S}_0^e\!\downarrow$ except that $a\colon Nat$ is assumed in $\mathcal{S}''_0$ (so that $\Sigma''_0 = \mathbf{Sig}(\mathcal{S}''_0)$ contains all symbols in $\mathcal{S}_0$ except for $d\colon Nat$). We have the obvious fitting $\mathcal{S}'_0 \xleftarrow{\iota'} (\mathcal{S}''_0)^e\!\Downarrow \xrightarrow{\iota''} \mathcal{S}''_0$ between $\mathcal{S}'_0$ and $\mathcal{S}''_0$, where ι' and ι'' are signature inclusions, and the pushout of ι' and ι'' yields $\mathcal{S}_0$ with inclusions $\tau'\colon \mathcal{S}' \to \mathcal{S}_0$ and $\tau''\colon \mathcal{S}'' \to \mathcal{S}_0$.

Consider then the following two specifications:

$$SP'_4 = SP'_N \textbf{ then } (b = succ(a) \lor b = succ(succ(a))) \land$$
$$(c = succ(succ(a)) \lor c = succ(succ(succ(a)))) \land d = succ(b)$$
$$SP''_4 = SP''_N \textbf{ then } (b = succ(a) \lor b = succ(succ(a))) \land$$
$$(c = succ(succ(a)) \lor c = succ(succ(succ(a)))) \land e = succ(c)$$
$$SP'_5 = SP'_N \textbf{ then } b = succ(succ(a)) \land c = succ(succ(a)) \land d = succ(c)$$
$$SP''_5 = SP''_N \textbf{ then } b = succ(succ(a)) \land c = succ(succ(a)) \land e = succ(c)$$

where SP'_N and SP''_N are SP_N (see Example 4.6) rewritten to the signature Σ'_0 and Σ''_0, respectively (equivalently, take $SP'_N = SP_N|_{\tau'}$ and $SP''_N = SP_N|_{\tau''}$). Let $\mathcal{SP}'_4 = \langle \mathcal{S}'_0, SP'_4 \rangle$, $\mathcal{SP}''_4 = \langle \mathcal{S}''_0, SP''_4 \rangle$, $\mathcal{SP}'_5 = \langle \mathcal{S}'_0, SP'_5 \rangle$ and $\mathcal{SP}''_5 = \langle \mathcal{S}''_0, SP''_5 \rangle$.

$\mathcal{SP}'_4$ and $\mathcal{SP}''_5$ are not compatible; neither are $\mathcal{SP}'_5$ and $\mathcal{SP}''_4$.

On the other hand, $\mathcal{SP}'_4$ and $\mathcal{SP}''_4$ are compatible, and their sum $\mathcal{SP}'_4 \oplus \mathcal{SP}''_4 = \langle \mathcal{S}_0, \tau'(SP'_4) \cup \tau''(SP''_4) \rangle$ is equivalent to $\mathcal{SP}_4$ (see Example 4.6): $Mod[\tau'(SP'_4) \cup \tau''(SP''_4)] = Mod[SP_4]$.

The $\mathcal{S}'_0$-construction $\mathcal{C}'_0 = \mathcal{C}_0|_{\tau'}$ is a (clean) construction model of $\mathcal{SP}'_4$. Perhaps surprisingly, the $\mathcal{S}''_0$-construction $\mathcal{C}''_0 = \mathcal{C}_0|_{\tau''}$ is not a construction model of $\mathcal{SP}''_4$ — since $a\colon Nat$ is assumed here, and SP''_4 does not constrain its value, $\mathcal{C}''_0$ is not complete for $\mathcal{SP}''_4$. Consider the following Σ''_0-algebras that interpret Nat, $zero\colon Nat$ and $succ\colon Nat$ in the standard way, and for $i = 0, 1, \ldots$:

$$A_1^i = \{a = i, b = i+1, c = i+2, e = i+3\}$$
$$A_3^i = \{a = i, b = i+2, c = i+3, e = i+4\}$$
$$A_4^i = \{a = i, b = i+2, c = i+2, e = i+3\}$$
$$A_5^i = \{a = i, b = i+1, c = i+3, e = i+4\}$$

Then $\mathcal{C}'' = \{A_1^i, A_3^i, A_4^i, A_5^i \mid i = 0, 1, \ldots\}$ is a (clean) construction model of $\mathcal{SP}_4''$. Moreover, $\mathcal{C}_0' \oplus \mathcal{C}''$ is a construction model of $\mathcal{SP}_4' \oplus \mathcal{SP}_4''$ by Thm. 5.1 – hardly surprising, since in fact $\mathcal{C}_0' \oplus \mathcal{C}'' = \mathcal{C}_0$.

$\mathcal{SP}_5'$ and $\mathcal{SP}_5''$ are compatible as well. Their sum $\mathcal{SP}_5' \oplus \mathcal{SP}_5'' = \langle \mathcal{S}_0, \tau'(SP_5') \cup \tau''(SP_5'') \rangle$ is stronger than $\mathcal{SP}_5$ (see Example 4.6) in the sense that every construction model of $\mathcal{SP}_5' \oplus \mathcal{SP}_5''$ is a construction model of $\mathcal{SP}_5$, $CMod[\mathcal{SP}_5' \oplus \mathcal{SP}_5''] \subseteq CMod[\mathcal{SP}_5]$ (this is a stronger property than $Mod[\tau'(SP_5') \cup \tau''(SP_5'')] \subseteq Mod[SP_5]$). Construction models of $CMod[\mathcal{SP}_5' \oplus \mathcal{SP}_5'']$ may be built by amalgamating clean construction models of $\mathcal{SP}_5'$ and $\mathcal{SP}_5''$. For instance, $\{A_4|_{\Sigma_0'}\}$ is a clean construction model of $\mathcal{SP}_5'$ and $\{A_4^i \mid i = 0, 1, \ldots\}$ is a clean construction model of $\mathcal{SP}_5'$. Their amalgamation is $\{A_4\}$ which is a clean construction model of $\mathcal{SP}_5' \oplus \mathcal{SP}_5''$ (and of $\mathcal{SP}_5$ as well). □

6 Final remarks

We propose an algebraic framework to uniformly deal with modules that covers non-parameterised and parameterised cases, as well as capturing complex dependencies between parts of modules which cannot be captured in standard approaches to parameterisation without using higher-order dependencies. The advantage of this proposal is that it keeps *constructions*, the semantic objects modelling the modules considered, relatively simple — they just are classes of algebras, subject to a simple technical condition which reflects the role of the module elements defined by the module (see Sect. 3). To specify such constructions, classified by construction signatures (see Sect. 2), we propose the use of construction specifications, essentially denoting classes of algebras (as in [ST12]). The definition of what it means for a construction to satisfy such a construction specification may appear a bit complex (see Sect. 4) but in our view they quite intuitively capture both the parameterisation mechanism and the dependency structure involved in specifying a construction. Theorem 4.2 offers simple and methodologically justified sufficient conditions to ensure that a construction specification has a construction model. We introduce the sum operation (see Sect. 5) as a basic tool to combine construction specifications, and show that under simple expected conditions, the operation is compositional in the sense that for any construction specifications, their respective constructions may be combined to yield a construction model of their sum. This operation is shown to cover typical uses of algebraic specifications of standard parameterised modules.

There are a number of directions which require further investigation and development.

Perhaps most importantly, we have not provided any notion of refinement for construction specifications. While simple refinements that just require classes of construction models to be narrowed in refinement steps (cf. [ST12]) may be used here, it soon turns out that this is unsatisfactory. The problem is that reducts along construction signature morphisms, a typical construct allowing auxiliary symbols to be introduced in the course of stepwise specification refinement, do not in general preserve construction models of a construction specification (see the remarks at the beginning of Sect. 5, with a specific instance of the problem hidden in Example 5.5). The solution must be to consider construction signature morphisms of a different kind, with different properties than those used here for putting together construction specifications. This was done in [Mar14], and used there to propose a construction-based approach to architectural design and development, in the style of architectural specifications and refinements in CASL [BST02, MST04]. It would be interesting and potentially useful to develop a uniform treatment of both kinds of construction signature morphisms. Useful sufficient conditions that ensure compatibility of reducts w.r.t. construction signature morphisms with hiding and translation for constructor specifications are needed here.

We assumed from the beginning that we deal with finite algebraic signatures only. Further developments, partially carried out in [Mar14], are needed to remove this assumption and allow the use of infinite signatures as well (for instance, to algebraically model polymorphism, or modules with higher-order operations). An arbitrary mixture of infinite signatures and strict dependency orders may raise methodological doubts and leads to technical troubles, see [Mar12] for discussion and a proposed solution to use dependency structures of a bounded height. In fact, the essence of the proofs here indicates that the induction on the number of signature elements used in the key proofs may be replaced by induction on the height of the dependency chains in the construction signatures, see [Mar14].

We worked in this paper with standard algebraic signatures and algebras. An important further task should be to follow [Mar14] and move developments to an arbitrary institution [GB92] equipped with additional structure to introduce the sets of symbols used in the signatures, similarly as in the semantics of CASL [Mos04]. We refrained from doing this here to keep the intuition, presentation and technicalities a bit clearer — we hope!

Finally, more examples and case studies are needed to better illustrate the use of the framework proposed as well as to evaluate its potential strengths and weaknesses.

Acknowledgements: Thanks to the referees for their helpful comments. This work has been partially supported by the Polish National Science Centre, grant 2013/11/B/ST6/01381 (AT).

References

[Asp97] D. Aspinall. *Type Systems for Modular Programming and Specification*. Ph.D. thesis, University of Edinburgh, Department of Computer Science, 1997.

[BG77] R. M. Burstall and J. A. Goguen. Putting theories together to make specifications. In: *Fifth International Joint Conference on Artificial Intelligence*, 1045–1058, Boston, 1977.

[BG81] R. M. Burstall and J. A. Goguen. An informal introduction to specifications using Clear. In: R. S. Boyer and J. S. Moore, eds., *The Correctness Problem in Computer Science*, 185–213. Academic Press, 1981. Also in: Narain Gehani and Andrew D. McGettrick, eds., *Software Specification Techniques*. Addison-Wesley, 1986.

[BM04] M. Bidoit and P. D. Mosses, eds. CASL *User Manual*, *Lecture Notes in Computer Science*, vol. 2900. Springer, 2004. See also `http://www.informatik.uni-bremen.de/cofi/wiki/index.php/CASL`.

[BST02] M. Bidoit, D. Sannella, and A. Tarlecki. Architectural specifications in CASL. *Formal Aspects of Computing*, 13:252–273, 2002.

[EM85] H. Ehrig and B. Mahr. *Fundamentals of Algebraic Specification 1*, *EATCS Monographs on Theoretical Computer Science*, vol. 6. Springer, 1985.

[GB92] J. A. Goguen and R. M. Burstall. Institutions: Abstract model theory for specification and programming. *Journal of the Association for Computing Machinery*, 39(1):95–146, 1992.

[Gog84] J. A. Goguen. Parameterized programming. *IEEE Trans. Software Eng.*, 10(5):528–544, 1984.

[Gog90] J. A. Goguen. Higher-order functions considered unnecessary for higher-order programming. In: D. A. Turner, ed., *Research Topics in Functional Programming*, 309–351. Addison-Welsey, 1990.

[KBS91] B. Krieg-Brückner and D. Sannella. Structuring specifications in-the-large and in-the-small: Higher-order functions, dependent types and inheritance in SPECTRAL. In: *Proc. Colloq. on Combining Paradigms for Software Development. Intl. Joint Conf. on Theory and Practice of Software Development (TAPSOFT'91)*, Brighton, *Lecture Notes in Computer Science*, vol. 494, 103–120. Springer, 1991.

[Mar12] G. Marczyński. Algebraic signatures enriched by dependency structure. In: T. Mossakowski and H. Kreowski, eds., *Recent Trends in Algebraic Development Techniques - 20th International Workshop, WADT 2010, Etelsen, Germany, July 1-4, 2010, Revised Selected Papers*, *Lecture Notes in Computer Science*, vol. 7137, 226–250. Springer, 2012.

[Mar14] G. Marczyński. *Specifications of Software Architectures using Diagrams of Constructions*. Ph.D. thesis, University of Warsaw, Faculty of Mathematics, Informatics and Mechanics, submitted 2014. See `http://mimuw.edu.pl/~gmarc/gmarc_phd_en.pdf`.

[Mos04] P. D. Mosses, ed. CASL *Reference Manual*, *Lecture Notes in Computer Science*, vol. 2960. Springer, 2004.

[MST04] T. Mossakowski, D. Sannella, and A. Tarlecki. A simple refinement language for CASL. In: J. Fiadeiro, ed., *Recent Trends in Algebraic Development Techniques. Selected Papers from the 17th International Workshop on Algebraic Development Techniques*, Barcelona, *Lecture Notes in Computer Science*, vol. 3423, 162–185. Springer, 2004.

[SST92] D. Sannella, S. Sokołowski, and A. Tarlecki. Toward formal development of programs from algebraic specifications: Parameterisation revisited. *Acta Informatica*, 29(8):689–736, 1992.

[ST88] D. Sannella and A. Tarlecki. Specifications in an arbitrary institution. *Information and Computation*, 76(2–3):165–210, 1988.

[ST91] D. Sannella and A. Tarlecki. A kernel specification formalism with higher-order parameterization. In: H. Ehrig, K. P. Jantke, F. Orejas, and H. Reichel, eds., *Recent Trends in Data Type Specification, Proceedings 7th Workshop on Abstract Data Types, Wusterhausen, Dosse, Germany, April 17-20, 1990*, *Lecture Notes in Computer Science*, vol. 534, 274–296. Springer, 1991.

[ST12] D. Sannella and A. Tarlecki. *Foundations of Algebraic Specification and Formal Software Development*. Monographs in Theoretical Computer Science. An EATCS Series. Springer, 2012.

[Tar11] A. Tarlecki. Some nuances of many-sorted universal algebra: A review. *Bulletin of the European Association for Theoretical Computer Science*, 104:89–111, 2011.