Behavioural semantics of algebraic specifications in arbitrary logical systems

Michał Misiak

Warsaw University, Faculty of Mathematics, Informatics and Mechanics**

Abstract. Behavioural semantics for specifications plays a crucial role in the formalization of the developments process, where a specification need not to be implemented exactly but only so that the required system behaviour is achieved. There are two main approaches to the definition of behavioural semantics: the internal one (called behavioural semantics) and external one (called abstractor semantics).
In this paper we present a notion of a behavioural concrete institution which is based on a notion of a concrete institution. The basic idea to form a behavioural institution (i.e. to ensure the satisfaction condition holds) is adopted from [2]. The behavioural concrete institution is a generalization of the COL-institution. In this work we also compare the resulted behavioural semantics with the abstractor semantics.

1 Introduction

One of the problems of algebraic-style specification of software systems is that the strict interpretation of a specification is often inadequate in practice. Typically a specification need not to be implemented exactly but only so that the required system behaviour is achieved. To cope with this problem the semantics of specifications must be redefined resulting in the so called *behavioural* or *observational interpretation of specifications*. There are two main approaches to the definition of behavioural semantics of algebraic specifications. The internal approach involves introducing an indistinguishability relation between elements of models. The external approach is based on an equivalence relation between models. These two approaches are related to each other and coincide in some cases, see [4].

In this work we aim at a general definition of a behavioural semantics for algebraic specifications in an arbitrary logical system. The key notion for this purpose is the notion of institution, introduced in [6].

We propose a notion of a *behavioural concrete institution*. This framework is based on the notion of concrete model category as introduced in [5]. The idea is to equip the model categories of institutions considered with concretization functor, thus adding "carriers" to the models considered. Then, a *concrete institution*

** This work has been partially supported by KBN grant 7T11C 002 21 and European AGILE project IST-2001-32747.

is just an ordinary institution in which all categories of models are concrete categories and each signature has a set of sorts.

To define a behavioural concrete institution we first need to define the behavioural satisfaction relation and so the behavioural semantics of flat specifications, and then extend it to form an institution. The behavioural semantics of flat specifications was introduced in [5], but this approach doesn't allow us to form an institution, since the satisfaction condition doesn't hold. Therefore we follow the idea presented in [2]. The behavioural concrete institution is a generalization of the COL-institution.

2 Basic notions

An S-sorted set is a family $X = (X_s)_{s\in S}$ of sets. Most standard notions concerning sets can be generalized to S-sorted sets. For example, let $X = (X_s)_{s\in S}$, $Y = (Y_s)_{s\in S}$ be S-sorted sets:

- X is a subset of Y, written $X \subseteq Y$ if $X_s \subseteq Y_s$ for all $s \in S$;
- Cartesian product of X and Y is defined as $X \times Y = (X_s \times Y_s)_{s\in S}$;
- an S-sorted relation between elements of X and Y is $R \subseteq X \times Y$; if $x \in X_s$ and $y \in Y_s$ for some $s \in S$, then the fact that x is in relation R with y will be denoted $x\ R_s\ y$ or simply $x\ R\ y$;
- an S-sorted function from X to Y is $f = (f : X_s \to Y_s)_{s\in S}$;
- a kernel of an S-sorted function $f : X \to Y$ is $\mathsf{ker}(f) = (\mathsf{ker}(f_s))_{s\in S}$, where $\mathsf{ker}(f_s) = \{(x, x') \mid f_s(x) = f_s(x')\}$;

The subscript s will be often omitted, for example $x \in X_s$ will be written $x \in X$, for short.

A relation $\approx \subseteq X \times X$ is an equivalence if for all $s \in S$, $\approx_s$ is an equivalence. A quotient of X by an equivalence $\approx$ is defined $X/{\approx} = \{[x]_\approx \mid x \in X\}$ (and it is an S-sorted set), where $[x]_\approx = \{x' \in X \mid x \approx x'\}$.

S-sorted sets with S-sorted functions form the category $\mathbf{Set}^S$ of S-sorted sets.

Categories are denoted with bold faces, like $\mathbf{Set}^S$. Objects of a category $\mathbf{K}$ are denoted $|\mathbf{K}|$. The fact that A is an object of a category $\mathbf{K}$ is written $A \in |K|$. If $f : A \to B$ is a morphism of $\mathbf{K}$ then it will be denoted $f \in \mathbf{K}$ or $f : A \to B \in \mathbf{K}$ (the latter brings an additional information about the source and target of the morphism f). The composition of morphisms $f : A \to B$ and $g : B \to C$ is denoted with ';' (semicolon) and written in the diagrammatic order, $f; g$.

Functors are also usually denoted with bold faces, $\mathbf{F} : \mathbf{K1} \to \mathbf{K2}$.

A notion of *institution* was introduced in [6], but in this paper we work with a slightly different definition. The definition we work with can be found, e.g in [7]. An institution $\mathbf{INS} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma\in|\mathbf{Sign}|})$ consists of:

- a category **Sign** of signatures;
- a *sentence functor* $\mathbf{Sen} : \mathbf{Sign} \to \mathbf{Set}$;

– a *model functor* $\mathbf{Mod} : \mathbf{Sign}^{op} \rightarrow \mathbf{Cat}$;
– for each signature $\Sigma \in |\mathbf{Sign}|$, a *satisfaction relation* $\models_{\Sigma} \subseteq |\mathbf{Mod}(\Sigma)| \times \mathbf{Sen}(\Sigma)$ such that for any signature morphism $\sigma : \Sigma \rightarrow \Sigma' \in \mathbf{Sign}$, Σ-sentence $\phi \in \mathbf{Sen}(\Sigma)$ and Σ'-model $M' \in |\mathbf{Mod}(\Sigma')|$:

$$M' \models_{\Sigma'} \mathbf{Sen}(\sigma)(\phi) \qquad \textit{iff} \qquad \mathbf{Mod}(\sigma)(M') \models_{\Sigma} \phi.$$

The above condition is called *satisfaction condition.*

Throughout this paper the notation for $\mathbf{Sen}(\sigma)(\phi)$ and $\mathbf{Mod}(\sigma)(M')$ will be simplified, i.e. $\mathbf{Sen}(\sigma)(\phi)$ will be simply written as $\sigma(\phi)$ and $\mathbf{Mod}(\sigma)(M')$ will be denoted $M'|_{\sigma}$. The functor $\_|_{\sigma} : \mathbf{Mod}(\Sigma') \rightarrow \mathbf{Mod}(\Sigma)$ ($\mathbf{Mod}(\sigma)$) is called *reduct functor*.

An institution **INS** has the *amalgamation property* is for each pushout in the category of signatures **Sign**,

$$\begin{array}{ccc} \Sigma^1 & \xrightarrow{\sigma'_1} & \Sigma' \\ \sigma_1 \big\uparrow & & \big\uparrow \sigma'_2 \\ \Sigma & \xrightarrow[\sigma_2]{} & \Sigma^2, \end{array}$$

Σ^1-model $M_1 \in |\mathbf{Mod}(\Sigma^1)|$, Σ^2-model $M_2 \in |\mathbf{Mod}(\Sigma^2)|$ such that $M_1|_{\sigma_1} = M_2|_{\sigma_2}$ there exists a unique model $M' \in |\mathbf{Mod}(\Sigma')|$ such that $M|_{\sigma'_1} = M_1$ and $M'|_{\sigma'_2} = M_2$.

The semantics of a specification SP in any institution **INS** is a signature of this specification, Sig[SP] and a class of models of this specification, Mod[SP]. In each institution **INS** the following standard specification building operations are available:

– for $\Sigma \in |\mathbf{Sign}|$, $\Phi \subseteq \mathbf{Sen}(\Sigma)$, a basic specification (presentation), (Σ, Φ):
  • $\mathrm{Sig}[\mathrm{SP}] = \Sigma$,
  • $\mathrm{Mod}[\mathrm{SP}] = \{M \in |\mathbf{Mod}(\Sigma)| \mid M \models_{\Sigma} \Phi\}$;
– for any specification SP^1, SP^2 with the same signature Σ, their union $\mathrm{SP}^1 \cup \mathrm{SP}^2$:
  • $\mathrm{Sig}[\mathrm{SP}^1 \cup \mathrm{SP}^2] = \Sigma$,
  • $\mathrm{Mod}[\mathrm{SP}^1 \cup \mathrm{SP}^2] = \mathrm{Mod}[\mathrm{SP}^1] \cap \mathrm{Mod}[\mathrm{SP}^2]$;
– for a signature morphism $\sigma : \Sigma \rightarrow \Sigma'$ and a specification SP with the signature Σ, **translate** SP **by** σ:
  • Sig[**translate** SP **by** σ] $= \Sigma'$,
  • Mod[**translate** SP **by** σ] $= \{M' \in |\mathbf{Mod}(\Sigma')| \mid M'|_{\sigma} \in \mathrm{Mod}[\mathrm{SP}]\}$;
– for a signature morphism $\sigma : \Sigma \rightarrow \Sigma'$ and a specification SP′ with the signature Σ', **derive from** SP′ **by** σ:
  • Sig[**derive from** SP′ **by** σ] $= \Sigma$,
  • Mod[**derive from** SP′ **by** σ] $= \{M'|_{\sigma} \mid M' \in \mathrm{Mod}[SP']\}$.

Let $\sigma : \Sigma \rightarrow \Sigma'$ be a signature morphism. The reduct functor $\_|_{\sigma}$ is *isomorphic compatible* if for each Σ'-model $M' \in |\mathbf{Mod}(\Sigma')|$, Σ-model $N \in |\mathbf{Mod}(\Sigma)|$ that is isomorphic to $M'|_{\sigma}$ there exists a model $N' \in |\mathbf{Mod}(\Sigma')|$ isomorphic to M' such that $N'|_{\sigma} = N$. A specification SP *has isomorphic compatible reduct functors* if for each signature morphism used to build this specification, the corresponding reduct functor is isomorphic compatible.

3 Concrete categories

The contents of this section is a selection of notions presented in [1] and [5]. The basic intuition to follow is that objects of a concrete category come equipped with carrier sets and morphisms can be though of as a functions between carrier sets that preserve the object structure. This additional structure of a category allows us to define many concepts from the universal algebra, like subobjects or quotients.

Definition 1. *An* S-concrete category *is a category* $\mathbf{K}$ *together with a concretization functor* $|\_| : \mathbf{K} \to \mathbf{Set}^S$ *that is faithful.*

The indicator S will be often omitted when dealing with S-concrete categories.

Concrete categories as defined above are similar to *constructs* in [1] but in this work we deal with many-sorted sets.

Concrete categories will be denoted simply by $|\_| : \mathbf{K} \to \mathbf{Set}^S$ instead of $\mathbf{K}$ together with $|\_| : \mathbf{K} \to \mathbf{Set}^S$ since in the concretization functors the whole information about the concrete category is included (i.e. the category $\mathbf{K}$ and the concretization functor itself).

Throughout this section, let $|\_| : \mathbf{K} \to \mathbf{Set}^S$ be a concrete category.

Proposition 1. *For any morphism* $f : A \to B \in \mathbf{K}$*:*

- *if* $|f|$ *is surjective then* f *is an epimorphism;*
- *if* $|f|$ *is injective then* f *is a monomorphism.*

Definition 2. *A concrete category* $|\_| : \mathbf{K} \to \mathbf{Set}^S$ *is* transportable *if for each object* $A \in |\mathbf{K}|$ *and a bijective function* $i : |A| \to X$ *there exists an object* $B \in |\mathbf{K}|$ *and an isomorphism* $i' : A \to B$ *such that* $|i'| = i$ *(and* $|B| = X$*).*

In [5] the notion of transportability is called *admitting of renaming of elements of objects*.

Definition 3. *An isomorphism* $i : A \to B \in \mathbf{K}$ *is* identity-carried *if* $|i|$ *is an identity. Two objects* $A, B \in |\mathbf{K}|$ *are* exactly isomorphic *if there exists an identity-carried isomorphism* $i : A \to B$*.*

Subobjects A notion of a subobject can be found e.g. in [1], where it is called *initial subobject*, but in this work, to simplify matters, we use a slightly different definition.

Definition 4. *Let* $A \in |\mathbf{K}|$ *be an object of* $\mathbf{K}$*. A* subobject *of* A *is an object* $B \in |\mathbf{K}|$ *together with a morphism* $\iota_{B \hookrightarrow A} : B \to A$ *such that* $|\iota_{B \hookrightarrow A}| : |B| \to |A|$ *is an inclusion and for each morphism* $f : C \to A$ *with* $|f|(|C|) \subseteq |B|$[1] *there exists a morphism* $f' : C \to B$ *such that* $f'; \iota_{B \hookrightarrow A} = f$*.*

[1] $|f|(|C|)$ is the image of the set $|C|$ under the function $|f|$.

If B is a subobject of A then there exists exactly one morphism $\iota_{B\hookrightarrow A} : B \to A$ such that $|\iota_{B\hookrightarrow A}|$ is an inclusion (follows from the faithfulness of $|\_|$). Moreover if $f : C \to A$ is a morphism with $|f|(|C|) \subseteq |B|$ then the morphism $f' : C \to B$ such that $f'; \iota_{B\hookrightarrow A} = f$ is unique.

The difference between the notion of a subobject presented here and the notion of an initial subobject from [1] is that the embedding ($\iota_{B\hookrightarrow A}$, see [1]) is required here to be an inclusion, not only an injection (like in [1]). That simplifies the definition of a generated subobject.

There is also a slight difference between the definition of a subobject presented here and in [5]. The subobjects defined here are called *full* in [5].

Proposition 2. *Let $A, B, C \in |\mathbf{K}|$. If B is a subobject of A and C is a subobject of B then C is a subobject of A.*

Definition 5. *Let $A \in |\mathbf{K}|$ be an object and $X \subseteq |A|$. A* subobject *of A* generated by *X is a subobject B of A such that $X \subseteq |B|$ and for any subobject C of A if $X \subseteq |C|$ then $|B| \subseteq |C|$.*

Proposition 3. *A subobject of A generated by $X \subseteq |A|$, if it exists, is unique up to an identity-carried isomorphism. Moreover, any object exactly isomorphic to a subobject of A generated by X is a subobject of A generated by X.*

The generated subobject of A by $X \subseteq |A|$ will be denoted $\langle X\rangle_A$ and the inclusion morphism, $\iota_{\langle X\rangle_A\hookrightarrow A} : \langle X\rangle_A \to A$, will be written $\iota_{X\hookrightarrow A}$, for short.

Definition 6. *A concrete category $|\_| : \mathbf{K} \to \mathbf{Set}^S$* has generated subobjects *if for each $A \in |\mathbf{K}|$ and $X \subseteq |A|$ there exists the subobject of A generated by X.*

Quotients

Definition 7. *Let $A \in |\mathbf{K}|$ be an object of $\mathbf{K}$. A* quotient *of A is an object $B \in |\mathbf{K}|$ together with an epimorphism $\pi_{A/B} : A \to B$ such that for any morphism $f : A \to C$ with $\mathsf{ker}(|\pi_{A/B}|) \subseteq \mathsf{ker}(|f|)$ there exists a morphism $f' : B \to C$ such that $\pi_{A/B}; f' = f$.*

A quotient B of A is final *if $|\pi_{A/B}|$ is surjective.*

If B is a quotient of A then the morphism $\pi_{A/B}$ is called the *quotient projection*.

The notion of a final quotient as defined above comes from [1]. In [5] final quotients are called *surjective quotients*.

Definition 8. *Let $A \in |\mathbf{K}|$ be an object of $\mathbf{K}$. An equivalence relation $\approx\, \subseteq |A| \times |A|$ is a* congruence *on A if there exists a morphism $f : A \to B$ such that $\mathsf{ker}(|f|) = \approx$.*

A quotient of A by $\approx$ is a quotient B of A with $\pi_{A/B} : A \to B$ such that $\mathsf{ker}(|\pi_{A/B}|) = \approx$.

Proposition 4. *A quotient of A by a congruence $\approx \subseteq |A| \times |A|$, if it exists, is unique up to an isomorphism. Moreover, any object isomorphic to a quotient of A by $\approx$ is a quotient of A by $\approx$.*

The quotient of A by a congruence $\approx \subseteq |A| \times |A|$ will be denoted $A/\approx$ and the morphism $\pi_{A/(A/\approx)}$ will be simply written as $\pi_{A/\approx} : A \to A/\approx$.

Let $A \in |\mathbf{K}|$ be an object of K and $\approx \subseteq |A| \times |A|$ be a congruence. If B is a subobject of A then the quotient (if it exists) of B by $\approx \cap |B| \times |B|$ will be simply denoted $B/\approx$ instead of $B/(\approx \cap |B| \times |B|)$. Notice that if $\approx$ is a congruence on A then $\approx$ (more precisely $\approx \cap |B| \times |B|$) is also a congruence on B.

Definition 9. *A concrete category* $|\_| : \mathbf{K} \to \mathbf{Set}^S$ has (final) quotients *if for each object $A \in |\mathbf{K}|$ and a congruence $\approx \subseteq |A| \times |A|$ on A there exists a (final) quotient of A by $\approx$.*

Definition 10. *In a concrete category* $|\_| : \mathbf{K} \to \mathbf{Set}^S$ subobjects are compatible with quotients *if for each object $A \in |\mathbf{K}|$, its subobject $\iota_{B \hookrightarrow A} : B \to A$ and a congruence $\approx \subseteq |A| \times |A|$ if quotients $A/\approx$ and $B/\approx$ exist then $B/\approx$ is a subobject of $A/\approx$ (formally there exist an object $C \in |\mathbf{K}|$ isomorphic to $B/\approx$ which is a subobject of A) and the following diagram commute:*

$$\begin{array}{ccc} B & \xrightarrow{\iota_{B \hookrightarrow A}} & A \\ {\scriptstyle \pi_{B/\approx}} \downarrow & & \downarrow {\scriptstyle \pi_{A/\approx}} \\ B/\approx & \xrightarrow[\iota_{B/\approx \hookrightarrow A/\approx}]{} & A/\approx, \end{array}$$

i.e. $\iota_{B \hookrightarrow A}; \pi_{A/\approx} = \pi_{B/\approx}; \iota_{B/\approx \hookrightarrow A/\approx}$.

4 Concrete institutions

In this section we follow the ideas of the previous section and define a *concrete institution* which is an extension of the notion of the institution introduced in [6]. A concrete institution is an institution in which for each signature a set of sorts of this signature is available and each category of models is a concrete category.

The notation for the category of S-sorted sets, $\mathbf{Set}^S$, can be extended to denote the functor: $\mathbf{Set}^{(\text{-})} : \mathbf{Set}^{op} \to \mathbf{Cat}$. For a set S, $\mathbf{Set}^S$ is the category of S-sorted sets. For a function $\sigma : S \to S'$, $\mathbf{Set}^\sigma$ is the reduct functor, $\mathbf{Set}^\sigma : \mathbf{Set}^{S'} \to \mathbf{Set}^S$ defined: $\mathbf{Set}^\sigma((X_s)_{s \in S'}) = (Y_s)_{s \in S}$ with $Y_s = X_{\sigma(s)}$ and similarly for S'-sorted functions.

If $\approx' \subseteq A' \times B'$ is an S'-sorted relation then $\mathbf{Set}^\sigma(\approx')$ is well defined since, in fact, $\approx'$ is an S'-sorted set (and the result is an S-sorted relation between elements of $\mathbf{Set}^\sigma(A)$ and $\mathbf{Set}^\sigma(B)$).

Definition 11. *A* concrete institution $\mathbf{INS_c}$ *based on an institution* $\mathbf{INS} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|})$ *consists of* $\mathbf{INS}$ *together with a functor sorts* : $\mathbf{Sign} \to \mathbf{Set}$ *and a natural transformation* $|\_| : \mathbf{Mod} \to sorts^{op}; \mathbf{Set}^{(\text{-})}$ *between functors from* $\mathbf{Sign}^{op}$ *to* $\mathbf{Cat}$.

Thus, a concrete institution is a tuple:

$$\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, sorts, |\_|).$$

The functor $sorts : \mathbf{Sign} \to \mathbf{Set}$ yields, for each signature Σ, a set of sorts of this signature. The natural transformation $|\_| : \mathbf{Mod} \to sorts^{op}; \mathbf{Set}^{(\_)}$ is a family of concretization functors, $(|\_|_\Sigma : \mathbf{Mod}(\Sigma) \to \mathbf{Set}^{sorts(\Sigma)})_{\Sigma \in |\mathbf{Sign}|}$. The naturality of this transformation ensures that the following diagram commutes:

$$\begin{array}{cccc}
\Sigma & \mathbf{Mod}(\Sigma) & \xrightarrow{|_|_\Sigma} & \mathbf{Set}^{sorts(\Sigma)} \\
\sigma \big\downarrow & _|_\sigma \big\uparrow & & \big\uparrow \mathbf{Set}^{sorts(\sigma)} \\
\Sigma' & \mathbf{Mod}(\Sigma') & \xrightarrow[|_|_{\Sigma'}]{} & \mathbf{Set}^{sorts(\Sigma')},
\end{array}$$

where $\sigma : \Sigma \to \Sigma' \in \mathbf{Sign}$.

The commutativity of the above diagram allows us to simplify the notation. The functor $\mathbf{Set}^{sorts(\sigma)} : \mathbf{Set}^{sorts(\Sigma')} \to \mathbf{Set}^{sorts(\Sigma)}$ will be denoted $\_|_\sigma : \mathbf{Set}^{sorts(\Sigma')} \to \mathbf{Set}^{sorts(\Sigma)}$.

Let $\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, sorts, |\_|)$ be a concrete institution, fixed throughout this section.

Definition 12. *A satisfaction relation* $\models_\Sigma \subseteq |\mathbf{Mod}(\Sigma)| \times \mathbf{Sen}(\Sigma)$, *where* $\Sigma \in |\mathbf{Sign}|$, *is* isomorphism compatible *if for all isomorphic models* $A, B \in |\mathbf{Mod}(\Sigma)|$ *and* Σ*-sentence* $\phi \in \mathbf{Sen}(\Sigma)$ *the following holds:* $A \models_\Sigma \phi$ *iff* $B \models_\Sigma \phi$.

A concrete institution $\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, sorts, |\_|)$ has isomorphic compatible satisfaction relations *if for each signature* $\Sigma \in |\mathbf{Sign}|$ *the satisfaction relation* $\models_\Sigma$ *is isomorphic compatible.*

Definition 13. *A reduct functor* $\_|_\sigma$, *where* $\sigma : \Sigma \to \Sigma' \in \mathbf{Sign}$, preserves subobjects *if for each* Σ'*-model* $A' \in \mathbf{Mod}(\Sigma')$ *and its subobject* B' *the reduct* $B'|_\sigma$ *is a subobject of* $A'|_\sigma$.

In a concrete institution $\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, sorts, |\_|)$ reduct functors preserve subobjects *if for each signature morphism* $\sigma : \Sigma \to \Sigma' \in \mathbf{Sign}$, *the reduct functor* $\_|_\sigma$ *preserves subobjects.*

Definition 14. *A reduct functor* $\_|_\sigma$, *where* $\sigma : \Sigma \to \Sigma' \in \mathbf{Sign}$, preserves quotients *if for each* Σ'*-model* $A' \in \mathbf{Mod}(\Sigma')$ *and its quotient* $\pi_{A'/B'} : A' \to B'$ *the reduct* $\pi_{A'/B'}|_\sigma : A'|_\sigma \to B'|_\sigma$ *is a quotient of* $A'|_\sigma$.

In a concrete institution $\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, sorts, |\_|)$ reduct functors preserve quotients *if for each signature morphism* $\sigma : \Sigma \to \Sigma' \in \mathbf{Sign}$, *the reduct functor* $\_|_\sigma$ *preserves quotients.*

A concrete institution $\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, sorts, |\_|)$

- is transportable;
- has generated subobjects;

– has (final) quotients;
– subobjects are compatible with quotients

if for each signature $\Sigma \in |\mathbf{Sign}|$, the concrete category $|\_|_\Sigma : \mathbf{Mod}(\Sigma) \to \mathbf{Set}^{sorts(\Sigma)}$ has the corresponding property.

5 Behavioural concrete institutions

The extra structure of a concrete institution allows us to redefine the satisfaction relation to obtain its behavioural version.

Let $\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, sorts, |\_|)$ be a concrete institution, fixed throughout this section. We assume that $\mathbf{INS_c}$

– is transportable,
– has generated subobjects,
– has final quotients,
– subobjects are compatible with quotients,
– has isomorphic compatible satisfaction relations,
– reduct functors preserve subobjects and quotients.

5.1 Behavioural satisfaction relation

Reachability is an important concept of system specifications. A *reachability structure* on a model is a subset of the carrier set of this models. It contains the elements which are of interest from the user's point of view. In this work we follow the ideas introduced in [2] and do not require a reachability structure to be a subobject of a model considered contrary e.g. to [4] (where a reachability structure is implicitly incorporated into a notion of a partial congruence).

Definition 15. *A* reachability structure *over a signature $\Sigma \in |\mathbf{Sign}|$ is a family $\mathcal{R} = (\mathcal{R}_M)_{M \in |\mathbf{Mod}(\Sigma)|}$ of $sorts(\Sigma)$-sorted sets such that $\mathcal{R}_M \subseteq |M|_\Sigma$ for each $M \in |\mathbf{Mod}(\Sigma)|$.*

Another important aspect of system specifications is the concept of observability. In this work we generalize the notion of observational equality from [2] which we call here an *observability structure*. An observability structure on a model is an equivalence relation on the carrier set of this model. Unlike the approach presented in [4] we do not impose any further restrictions on an observability structure. The idea comes from [2].

Definition 16. *An* observability structure *over a signature $\Sigma \in |\mathbf{Sign}|$ is a family $\approx = (\approx_M)_{M \in |\mathbf{Mod}(\Sigma)|}$ of equivalence relations such that $\approx_M \subseteq |M|_\Sigma \times |M|_\Sigma$ (i.e. $\approx_M$ is an equivalence relation on $|M|_\Sigma$).*

Usually a reachability structure is determined by a distinguished set of constructor operations and an observability structure by a distinguished set of observer operations, see [2], but it is not the purpose of this work to present how those structures can be defined. The problem here is more complicated since in an arbitrary (concrete) institution the notion of an operation is not available. In this work we only present the way of defining the behavioural semantics of specifications given arbitrary reachability and observability structures.

A pair $(\mathcal{R}, \approx)$, where $\mathcal{R}$ is a reachability structure over a signature Σ and $\approx$ is an observability structure over the signature Σ, is called a *behavioural structure* over the signature Σ.

Definition 17. *A* behavioural signature $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *consists of:*

- *a signature* $\Sigma \in |\mathbf{Sign}|$*;*
- *a reachability structure* $\mathcal{R}$ *over the signature* Σ*;*
- *an observability structure* $\approx$ *over the signature* Σ*.*

Following the ideas from [2], since no restrictions were imposed on reachability and observability structures, we introduce two kinds of constraints on the class of models: the reachability and the observability constraint. The former is a well-known constraint which expresses the property that the only admissible models are those on which the reachability structure is a subobject of the model considered (intuitively it is closed under operations). But if we deal both with the reachability and observability concepts such a requirement is too strong, since from the user's point of view this is not different from allowing the elements of the submodel generated by its reachability structure to be indistinguishable from some elements in this reachability structure.

The latter (the observability constraint) simply states that the observability structure on a model must be a congruence on the subobject of this model generated by the reachability structure.

Definition 18. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature. A* Σ*-model* $M \in |\mathbf{Mod}(\Sigma)|$ *satisfies*

- *the* reachability constraint *if for each* $a \in |\langle \mathcal{R}_M \rangle_M|_\Sigma$ *there exists* $b \in \mathcal{R}_M$ *such that* $a \approx_M b$*;*
- *the* observability constraint *if* $\approx_M$ *is a congruence on* $\langle \mathcal{R}_M \rangle_M$ *(more precisely if* $\approx \cap\, |\langle \mathcal{R}_M \rangle_M|_\Sigma \times |\langle \mathcal{R}_M \rangle_M|_\Sigma$ *is a congruence on* $\langle \mathcal{R}_M \rangle_M$*).*

Definition 19. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature. A* Σ*-model* $M \in |\mathbf{Mod}(\Sigma)|$ *is called* behavioural *if it satisfies the reachability and the observability constraints.*

The class of all behavioural models over a behavioural signature Σ_{Beh} *will be denoted* $\mathrm{Mod}_{\mathrm{Beh}}(\Sigma_{\mathrm{Beh}})$*.*

The standard way of defining a *behavioural satisfaction relation* independently on the logical system is by the notion of a *behaviour* of a model, see [4] or [5]. For the logical systems in which the satisfaction relation is based on an equality between terms this approach (this definition of a behavioural satisfaction relation) is equivalent to the approach which involves changing the semantics of equality, see [4] or [2].

Definition 20. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature. The* behaviour $\mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M)$ *of a behavioural model* $M \in \mathrm{Mod}_{\mathrm{Beh}}(\Sigma_{\mathrm{Beh}})$ *is defined:*

$$\mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M) = \langle \mathcal{R}_M \rangle_M / {\approx_M}.$$

Definition 21. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature. A behavioural model* $M \in \mathrm{Mod}_{\mathrm{Beh}}(\Sigma_{\mathrm{Beh}})$ behaviourally satisfies *a sentence* $\phi \in \mathbf{Sen}(\Sigma)$,

$$M \models_{\Sigma_{\mathrm{Beh}}} \phi$$

if its behaviour satisfies the sentence ϕ *in the original sense,*

$$\mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M) \models_{\Sigma} \phi.$$

5.2 Behavioural concrete institutions

The above section covers only the case of flat specifications, sometimes called *satisfaction frame*, which is only a single fibre of an institution. In this section we present how the notions of the previous section can be used to form a *behavioural concrete institution.*

The first step is to impose additional requirements of the signature morphisms of the original institution $\mathbf{INS_c}$, to eliminate those that violate the satisfaction condition for behavioural satisfaction relation and behavioural models.

Definition 22. *A* behavioural signature morphism $\sigma : \Sigma_{\mathrm{Beh}} \to \Sigma'_{\mathrm{Beh}}$*, where* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *and* $\Sigma'_{\mathrm{Beh}} = (\Sigma', \mathcal{R}', \approx')$*, is a signature morphism* $\sigma : \Sigma \to \Sigma'$ *such that it preserves the reachability structure and the observability structure, i.e. if for each* Σ'*-model* $M' \in |\mathbf{Mod}(\Sigma')|$ *the following holds:* $\mathcal{R}'_{M'}|_\sigma = \mathcal{R}_{M'|_\sigma}$ *and* $\approx'_{M'}|_\sigma = \approx_{M'|_\sigma}$*.*

Now, given the notion of a behavioural morphism we can define the *category of all behavioural signatures.* This category contains all behavioural signatures and morphisms of this category are all behavioural signature morphisms.

Definition 23. *The* category of all behavioural signatures, $\mathbf{ASign}$*, consists of:*

- *objects are all behavioural signatures* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *such that* $\Sigma \in |\mathbf{Sign}|$ *and* $(\mathcal{R}, \approx)$ *is a behavioural structure over* Σ*;*
- *morphisms are all behavioural signature morphisms.*

The functor from the category of all behavioural signatures $\mathbf{ASign}$ to the category of signatures $\mathbf{Sign}$ which simply "forgets" about behavioural structures is called the *forgetful functor*, $\mathbf{AF} : \mathbf{ASign} \to \mathbf{Sign}$. It is defined: $\mathbf{AF}(\Sigma_{\mathrm{Beh}}) = \Sigma$ for $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx) \in |\mathbf{ASign}|$ and $\mathbf{AF}(\sigma) = \sigma$ for $\sigma \in \mathbf{ASign}$.

Theorem 1. *Let* $\sigma : \Sigma_{\mathrm{Beh}} \to \Sigma'_{\mathrm{Beh}} \in \mathbf{ASign}$ *be a behavioural signature morphism,* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *and* $\Sigma'_{\mathrm{Beh}} = (\Sigma', \mathcal{R}', \approx')$*. Then for any behavioural model* $M' \in \mathbf{Mod}(\Sigma')$ *the reduct of this model,* $M'|_\sigma$[2]*, is behavioural.*

[2] Formally, it should be written $M'|_{\mathbf{AF}(\sigma)}$, but to simplify matters, since it doesn't throw into confusion it will be denoted like above (i.e. $M'|_\sigma$).

The above theorem allows us to define the *behavioural model functor*, which for each behavioural signature $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ yields the category of all behavioural models over the signature Σ and for each behavioural signature morphism $\sigma : \Sigma_{\mathrm{Beh}} \to \Sigma'_{\mathrm{Beh}}$ it yields the restriction of the reduct functor $\_|_\sigma$ to the category of all behavioural models over the signature Σ' ($\Sigma'_{\mathrm{Beh}} = (\Sigma', \mathcal{R}', \approx')$). Th. 1 states that this definition is correct, i.e. the reduct of a behavioural model over the signature Σ' is a behavioural model over the signature Σ (w.r.t $(\mathcal{R}, \approx)$, where $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$).

Definition 24. *The* behavioural model functor $\mathbf{AMod} : \mathbf{ASign}^{op} \to \mathbf{Cat}$ *is defined:*

- *for* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx) \in |\mathbf{ASign}|$, $\mathbf{AMod}(\Sigma_{\mathrm{Beh}}) = \mathrm{Mod}_{\mathrm{Beh}}(\Sigma_{\mathrm{Beh}})$ *is the full subcategory of* $\mathbf{Mod}(\Sigma)$*;*
- *for* $\sigma : \Sigma_{\mathrm{Beh}} \to \Sigma'_{\mathrm{Beh}} \in \mathbf{ASign}$, *where* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *and* $\Sigma'_{\mathrm{Beh}} = (\Sigma', \mathcal{R}', \approx')$, $\mathbf{AMod}(\sigma) = \_|_\sigma$ *is the restriction of the reduct functor* $\_|_\sigma : \mathbf{Mod}(\Sigma') \to \mathbf{Mod}(\Sigma)$ *to the category* $\mathbf{AMod}(\Sigma'_{\mathrm{Beh}})$*.*

Theorem 2. *For each behavioural signature morphism* $\sigma : \Sigma_{\mathrm{Beh}} \to \Sigma'_{\mathrm{Beh}}$*, where* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *and* $\Sigma'_{\mathrm{Beh}} = (\Sigma', \mathcal{R}', \approx')$*, for each behavioural* Σ'*-model* $M' \in |\mathbf{AMod}(\Sigma'_{\mathrm{Beh}})|$ *and* Σ*-sentence* $\phi \in \mathbf{Sen}(\Sigma)$ *the following holds:*

$$M'|_\sigma \models_{\Sigma_{\mathrm{Beh}}} \phi \qquad \textit{iff} \qquad M' \models_{\Sigma'_{\mathrm{Beh}}} \sigma(\phi).$$

Definition 25. *The tuple*

$$\mathbf{AINS_c} = (\mathbf{ASign}, \mathbf{ASen}, \mathbf{AMod}, (\models_{\Sigma_{\mathrm{Beh}}})_{\Sigma_{\mathrm{Beh}} \in |\mathbf{ASign}|}, asorts, |\_|^a)$$

is the behavioural concrete institution *also called the* concrete institution of behavioural logic *based on the concrete institution* $\mathbf{INS_c}$*, where*

- $\mathbf{ASign}$ *is the category of all behavioural signatures,*
- $\mathbf{ASen} : \mathbf{ASign} \to \mathbf{Set}$ *is the behavioural sentence functor, defined:* $\mathbf{ASen} = \mathbf{AF}; \mathbf{Sen}$*,*
- $\mathbf{AMod} : \mathbf{ASign}^{op} \to \mathbf{Cat}$ *is the behavioural model functor,*
- *for each* $\Sigma_{\mathrm{Beh}} \in |\mathbf{ASign}|$*,* $\models_{\Sigma_{\mathrm{Beh}}}$ *is the behavioural satisfaction relation,* $\models_{\Sigma_{\mathrm{Beh}}} \subseteq |\mathbf{AMod}(\Sigma)| \times \mathbf{ASen}(\Sigma)$*,*
- $asorts : \mathbf{ASign} \to \mathbf{Set}$ *is the behavioural sorts functor, defined:* $asorts = \mathbf{AF}; sorts$*,*
- $|\_|^a : \mathbf{AMod} \to asorts^{op}; \mathbf{Set}^{(\_)}$ *is a natural transformation between functors from* $\mathbf{ASign}^{op}$ *to* $\mathbf{Cat}$*, defined: for a behavioural signature* $\Sigma_{\mathrm{Beh}} \in |\mathbf{ASign}|$*,* $|\_|^a_{\Sigma_{\mathrm{Beh}}} : \mathbf{AMod}(\Sigma) \to \mathbf{Set}^{asorts(\Sigma_{\mathrm{Beh}})}$ *is the restriction of the functor* $|\_|_\Sigma : \mathbf{Mod}(\Sigma) \to \mathbf{Set}^{sorts(\mathbf{AF}(\Sigma_{\mathrm{Beh}}))}$ *to the category of all behavioural models over* Σ_{Beh}*.*

The superscript 'a' in the natural transformation $|\_|^a : \mathbf{AMod} \to asorts^{op}; \mathbf{Set}^{(\_)}$ will be omitted.

The institution $\mathbf{AINS_c}$ is a rather "large" institution. The category of signatures of this institution contains all behavioural signatures $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ such that Σ is a signature from the original category of signatures **Sign** and $(\mathcal{R}, \approx)$ is an arbitrary behavioural structure. Such a freedom is usually inadequate in practice (when defining a behavioural semantics of a specification language based on the original semantics). Interesting cases are when behavioural structures are determined for example by given sets of constructor and observer operations (see [2]). The institution $\mathbf{AINS_c}$ was introduced for technical reasons, to express some properties concerning behavioural structures. Therefore we introduce a new behavioural concrete institution $\mathbf{BINS_c}$ in which the category of signatures **BSign** contains only some behavioural signatures and behavioural signature morphisms. In other words **BSign** is a subcategory of **ASign**. Formally $\mathbf{BINS_c}$ is a tuple:

$$\mathbf{BINS_c} = (\mathbf{BSign}, \mathbf{BSen}, \mathbf{BMod}, (\models_{\Sigma_{\mathrm{Beh}}})_{\Sigma_{\mathrm{Beh}} \in |\mathbf{BSign}|}, bsorts, |\_|^b).$$

The other components (apart from **BSign**) of $\mathbf{BINS_c}$ are defined in exactly the same way as in the institution $\mathbf{AINS_c}$. The forgetful functor $\mathbf{BF} : \mathbf{BSign} \rightarrow \mathbf{Sign}$ can also be easily defined.

An institution $\mathbf{BINS_c}$ can be thought of as a "subinstitution" of $\mathbf{AINS_c}$ with a smaller category of signatures. Note that $\mathbf{AINS_c}$ is a special case of $\mathbf{BINS_c}$.

5.3 Properties of behavioural concrete institutions

Let $\mathbf{INS_c}$ be a concrete institution that satisfies all the properties mentioned in the beginning of this section and $\mathbf{BINS_c}$ be an arbitrary behavioural concrete institution based on $\mathbf{INS_c}$. Of course there also exists the behavioural concrete institution $\mathbf{AINS_c}$ based on $\mathbf{INS_c}$ in which the category of signatures contains all behavioural signatures and all behavioural signature morphisms.

In this subsection we assume that the functor $sorts : \mathbf{Sign} \rightarrow \mathbf{Set}$ is cocontinuous.

Proposition 5. *If the category* **Sign** *of signatures if cocomplete then so is the category of all behavioural signatures* **ASign** *and the forgetful functor* $\mathbf{AF} : \mathbf{ASign} \rightarrow \mathbf{Sign}$ *is cocontinuous.*

One important property of an institution is the amalgamation property. Unfortunately a behavioural concrete institution doesn't have the amalgamation property even if the concrete institution on which it is based on has the amalgamation property. The counterexample can be found in [3], where the constructor based observational logic institution, which is a special case of a behavioural concrete institution, is presented. However there are some conditions under which the amalgamation union of two models exists. These conditions are generalization of the conditions for amalgamation from [3].

Proposition 6. *Let*

$$\begin{array}{ccc} \Sigma^1_{\mathrm{Beh}} = (\Sigma^1, \mathcal{R}^1, \approx^1) & \xrightarrow{\sigma'_1} & \Sigma'_{\mathrm{Beh}} = (\Sigma', \mathcal{R}', \approx') \\ \sigma_1 \uparrow & & \uparrow \sigma'_2 \\ \Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx) & \xrightarrow[\sigma_2]{} & \Sigma^2_{\mathrm{Beh}} = (\Sigma^2, \mathcal{R}^2, \approx^2) \end{array}$$

be a pushout in the category **BSign** *of behavioural signatures such that, the image of this diagram under the forgetful functor* **BF**,

$$\begin{array}{ccc} \Sigma^1 & \xrightarrow{\sigma'_1} & \Sigma' \\ \sigma_1 \uparrow & & \uparrow \sigma'_2 \\ \Sigma & \xrightarrow[\sigma_2]{} & \Sigma^2, \end{array}$$

is a pushout in the category **Sign**. *Assume that* $\mathbf{INS_c}$ *has the amalgamation property on this pushout, i.e. for any* $N_1 \in |\mathbf{Mod}(\Sigma^1)|$, $N_2 \in |\mathbf{Mod}(\Sigma^2)|$ *such that* $N_1|_{\sigma_1} = N_2|_{\sigma_2}$ *there exists the unique amalgamation* $N' \in |\mathbf{Mod}(\Sigma')|$ *of* N_1 *and* N_2 *(i.e.* $N'|_{\sigma'_1} = N_1$ *and* $N'|_{\sigma'_2} = N_2$*). Now, let* $M_1 \in |\mathbf{BSign}(\Sigma^1)|$, $M_2 \in |\mathbf{BSign}(\Sigma^2)|$ *be behavioural models such that* $M_1|_{\sigma_1} = M_2|_{\sigma_2}$. *If* $\langle \mathcal{R}^1_{M_1} \rangle_{M_1}|_{\sigma_1} = \langle \mathcal{R}_{M_1|_{\sigma_1}} \rangle_{M_1|_{\sigma_1}}$ *and* $\langle \mathcal{R}^2_{M_2} \rangle_{M_2}|_{\sigma_2} = \langle \mathcal{R}_{M_2|_{\sigma_2}} \rangle_{M_2|_{\sigma_2}}$ *then there exists the unique amalgamation* $M' \in |\mathbf{BSign}(\Sigma')|$ *of* M_1 *and* M_2.

5.4 Behavioural specifications

Given a behavioural concrete institution $\mathbf{BINS_c}$ all standard specification building operations are available:

- for $\Sigma_{\mathrm{Beh}} \in |\mathbf{BSign}|$, $\Phi \subseteq \mathbf{BSen}(\Sigma_{\mathrm{Beh}})$, a basic specification (presentation) $(\Sigma_{\mathrm{Beh}}, \Phi)$:
  - $\mathrm{Sig}[(\Sigma_{\mathrm{Beh}}, \Phi)] = \Sigma_{\mathrm{Beh}}$,
  - $\mathrm{Mod}[(\Sigma_{\mathrm{Beh}}, \Phi)] = \{M \in |\mathbf{BMod}(\Sigma_{\mathrm{Beh}})| \mid M \models_{\Sigma_{\mathrm{Beh}}} \Phi\}$;
- for any specifications $\mathrm{SP}^1_{\mathrm{Beh}}$, $\mathrm{SP}^2_{\mathrm{Beh}}$ with the same signature Σ_{Beh}, their union $\mathrm{SP}^1_{\mathrm{Beh}} \cup \mathrm{SP}^2_{\mathrm{Beh}}$:
  - $\mathrm{Sig}[\mathrm{SP}^1_{\mathrm{Beh}} \cup \mathrm{SP}^2_{\mathrm{Beh}}]$,
  - $\mathrm{Mod}[\mathrm{SP}^1_{\mathrm{Beh}} \cup \mathrm{SP}^2_{\mathrm{Beh}}] = \mathrm{Mod}[\mathrm{SP}^1_{\mathrm{Beh}}] \cap \mathrm{Mod}[\mathrm{SP}^2_{\mathrm{Beh}}]$;
- for a behavioural signature morphism $\sigma : \Sigma_{\mathrm{Beh}} \to \Sigma'_{\mathrm{Beh}}$ and a specification $\mathrm{SP}_{\mathrm{Beh}}$ with the signature Σ_{Beh}, **translate** $\mathrm{SP}_{\mathrm{Beh}}$ **by** σ:
  - $\mathrm{Sig}[$**translate** $\mathrm{SP}_{\mathrm{Beh}}$ **by** $\sigma] = \Sigma'_{\mathrm{Beh}}$,
  - $\mathrm{Mod}[$**translate** $\mathrm{SP}_{\mathrm{Beh}}$ **by** $\sigma] = \{M' \in |\mathbf{BMod}(\Sigma'_{\mathrm{Beh}})| \mid M'|_\sigma \in \mathrm{Mod}[\mathrm{SP}_{\mathrm{Beh}}]\}$;
- for a signature morphism $\sigma : \Sigma_{\mathrm{Beh}} \to \Sigma'_{\mathrm{Beh}}$ and a specification $\mathrm{SP}'_{\mathrm{Beh}}$ with the signature Σ'_{Beh}, **derive from** $\mathrm{SP}'_{\mathrm{Beh}}$ **by** σ:
  - $\mathrm{Sig}[$**derive from** $\mathrm{SP}'_{\mathrm{Beh}}$ **by** $\sigma] = \Sigma_{\mathrm{Beh}}$,
  - $\mathrm{Mod}[$**derive from** $\mathrm{SP}'_{\mathrm{Beh}}$ **by** $\sigma] = \{M'|_\sigma \mid M' \in \mathrm{Mod}[\mathrm{SP}'_{\mathrm{Beh}}]\}$.

For each behavioural specification $\mathrm{SP}_{\mathrm{Beh}}$ (i.e. a specification in the institution $\mathbf{BINS_c}$) there exists a standard specification SP (i.e. in the institution $\mathbf{INS_c}$) built in the same way as $\mathrm{SP}_{\mathrm{Beh}}$, which corresponds to this behavioural specification with the following property: $\mathrm{Sig}[\mathrm{SP}] = \mathbf{BF}(\mathrm{Sig}[\mathrm{SP}_{\mathrm{Beh}}])$. This correspondence can be easily defined by the induction on the structure of specifications. For example, if $\mathrm{SP}_{\mathrm{Beh}} = (\Sigma_{\mathrm{Beh}}, \Phi)$ is a basic specification, where $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$, then the corresponding standard specification is defined: $\mathrm{SP} = (\Sigma, \Phi)$. The definition of the correspondence for the others specification building operations are obvious, e.g. if $\mathrm{SP}_{\mathrm{Beh}} = \textbf{derive from } \mathrm{SP}'_{\mathrm{Beh}} \textbf{ by } \sigma$ then the corresponding standard specification is defined: $\mathrm{SP} = \textbf{derive from } \mathrm{SP}' \textbf{ by } \sigma$, where SP' is a standard specification that corresponds to $\mathrm{SP}'_{\mathrm{Beh}}$.

5.5 Examples

The described above notion of a behavioural concrete institution covers many institution of interest: institution of standard algebras, partial algebras with strong homomorphisms and CASL -institution with a slightly changed notion of a homomorphism between models.

6 Relating behavioural and abstractor semantics

In this section we present relations between the behavioural and abstractor semantics.

Let $\mathbf{INS_c} = (\mathbf{Sign}, \mathbf{Sen}, \mathbf{Mod}, (\models_\Sigma)_{\Sigma \in |\mathbf{Sign}|}, \mathit{sorts}, |\_|)$ be a concrete institution that satisfies all the assumptions presented in the beginning of the previous section (which allow us to define a behavioural concrete institution based on $\mathbf{INS_c}$), fixed throughout this section.

6.1 Abstractor specifications

Let us now briefly focus on the abstractor semantics. More information about it can be found for example in [5] or [4].

In fact the additional structure available in concrete institutions is not needed to define the abstractor semantics (i.e. the notion of the standard institution is sufficient for that purpose).

Let $\Sigma \in |\mathbf{Sign}|$ be a signature of a concrete institution $\mathbf{INS_c}$. An *abstractor equivalence* over the signature Σ is an equivalence relation between Σ-models, $\equiv \subseteq |\mathbf{Mod}(\Sigma)| \times |\mathbf{Mod}(\Sigma)|$. An abstractor equivalence $\equiv$ is called *isomorphism protecting* if all isomorphic models $M, N \in |\mathbf{Mod}(\Sigma)|$ are equivalent, $M \equiv N$.

For any class of Σ-models $\mathcal{M} \subseteq |\mathbf{Mod}(\Sigma)|$, the *abstractor closure* of $\mathcal{M}$ is the closure of this class under the abstractor equivalence, $\mathrm{Abs}_\equiv(\mathcal{M}) = \{M \in |\mathbf{Mod}(\Sigma)| \mid M \equiv N \textit{ for some } N \in \mathcal{M}\}$.

The notion of an abstractor closure allows us to define the abstractor semantics of specifications. Let SP be a specification with the signature Σ. The class of models which behaviourally (up to the abstractor equivalence $\equiv$) satisfy the specification SP is the abstractor closure of the class of models which satisfy the specification SP literally, i.e. $\mathrm{Mod}[\textbf{abstract}\ \mathrm{SP}\ \textbf{wrt}\ \equiv] = \mathrm{Abs}_{\equiv}(\mathrm{Mod}[\mathrm{SP}])$.

6.2 Behavioural specifications and behavioural closure operator

Let $\mathbf{BINS_c} = (\mathbf{BSign}, \mathbf{BSen}, \mathbf{BMod}, (\models_{\Sigma_{\mathrm{Beh}}})_{\Sigma_{\mathrm{Beh}} \in |\mathbf{BSign}|}, \mathit{bsorts}, |\_|)$ be a behavioural concrete institution based on $\mathbf{INS_c}$.

A similar condition to isomorphism protecting can be expressed for behavioural structures. It is called isomorphism compatibility and it differs slightly from the one introduced in [4] since in this framework not for each model the behaviour is defined.

Definition 26. *A behavioural signature* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx) \in |\mathbf{BSign}|$ *is* isomorphism compatible *if for each isomorphic Σ-models $M, N \in |\mathbf{Mod}(\Sigma)|$ if M is behavioural then N is behavioural and in that case their behaviours are isomorphic,* $\mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M) \cong \mathcal{B}_{\Sigma_{\mathrm{Beh}}}(N)$.

The notion of a fully abstract model can be found in [4] or [5]. A model is fully abstract if the behavioural structure on this model is trivial, i.e. the reachability structure on such a model is the whole carrier set of this model and the observability structure is an identity relation.

Definition 27. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx) \in |\mathbf{BSign}|$ *be a behavioural signature. A model* $M \in |\mathbf{Mod}(\Sigma)|$ *is* fully abstract *if* $\mathcal{R}_M = |M|_{\Sigma_{\mathrm{Beh}}}$ *and* $\approx_M = id_{|M|_{\Sigma_{\mathrm{Beh}}}}$. *If* $\mathcal{M} \subseteq |\mathbf{Mod}(\Sigma)|$ *is a class of Σ-models then,* $\mathrm{FA}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M})$ *denotes the class of all fully abstract models in* $\mathcal{M}$, $\mathrm{FA}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M}) = \{M \in \mathcal{M} \mid M \textit{ is fully abstract}\}$.

Note that if a model $M \in |\mathbf{Mod}(\Sigma)|$ is fully abstract then it is behavioural. Therefore $\mathrm{FA}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M})$ is a class of behavioural models (even if in $\mathcal{M}$ there are non-behavioural models).

The regularity of a behavioural structure (signature) is also an important property, see [4]. It express the idempotency of the behaviour operator.

Definition 28. *A behavioural signature* $\Sigma_{\mathrm{Beh}} \in |\mathbf{BSign}|$ *is called:*

- weakly regular *if for each behavioural model* $M \in |\mathbf{BMod}(\Sigma_{\mathrm{Beh}})|$ *its behaviour is behavioural and it is isomorphic to the behaviour of the behaviour of this model, i.e.* $\mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M) \cong \mathcal{B}_{\Sigma_{\mathrm{Beh}}}(\mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M))$;
- regular *if for each behavioural model* $M \in |\mathbf{BMod}(\Sigma_{\mathrm{Beh}})|$ *its behaviour is a fully abstract model.*

Regularity implies weak regularity. If a behavioural signature Σ_{Beh} is weakly regular and isomorphic compatible then the behavioural satisfaction relation $\models_{\Sigma_{\mathrm{Beh}}}$ is isomorphism compatible.

The above conditions which should be satisfied by any reasonable behavioural concrete institution (by all signatures in the behavioural concrete institution) will allow us to express relations between the internal approach and the external approach to the definition of behavioural semantics.

Definition 29. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx) \in |\mathbf{BSign}|$ *be a behavioural signature,* $\mathcal{M} \subseteq |\mathbf{Mod}(\Sigma)|$ *be a class of* Σ*-models. The* behavioural closure *of the class* M *is a class* $\mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M}) = \{M \in |\mathbf{BMod}(\Sigma_{\mathrm{Beh}})| \mid \mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M) \in \mathcal{M}\}$.

Note that, similarly as for the operator which yields the class of fully abstract models, even if there are some non-behavioural models in $\mathcal{M}$, the behavioural closure of this class contains only behavioural models.

Corollary 1. *Let* $\mathrm{SP}_{\mathrm{Beh}} = (\Sigma_{\mathrm{Beh}}, \Phi)$ *be a behavioural specification and* SP *be its corresponding standard specification. Then* $\mathrm{Mod}[\mathrm{SP}_{\mathrm{Beh}}] = \mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathrm{Mod}[\mathrm{SP}])$.

The following lemma is useful to prove relations between the behavioural and abstractor semantics.

Lemma 1. *Let* $\mathrm{SP}_{\mathrm{Beh}}$ *be a behavioural specification and* SP *its corresponding standard specification. If* SP *has isomorphic compatible reduct functors then* $\mathrm{Mod}[\mathrm{SP}_{\mathrm{Beh}}] \subseteq \mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathrm{Mod}[\mathrm{SP}])$*, where* $\Sigma_{\mathrm{Beh}} = \mathrm{Sig}[\mathrm{SP}_{\mathrm{Beh}}]$.

The proof of the above lemma is by the induction on the structure of specifications.

The opposite inclusion doesn't hold in general (i.e. $\mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathrm{Mod}[\mathrm{SP}]) \not\subseteq \mathrm{Mod}[\mathrm{SP}_{\mathrm{Beh}}]$), even in the case of standard algebras and equational logic.

6.3 Relations

The crucial notion for expressing relations between the two approaches to the definition of the behavioural semantics is the notion of *factorizability*, introduced in [4].

Definition 30. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature and* $\equiv$ *be an abstractor equivalence over* Σ*. The abstractor equivalence* $\equiv$ *is called* factorizable *by* Σ_{Beh} *(or by the behavioural structure* $(\mathcal{R}, \approx)$*) if the following two conditions hold:*

- *for all behavioural models* $M, N \in |\mathbf{BMod}(\Sigma_{\mathrm{Beh}})|$*,* $M \equiv N$ *iff* $\mathcal{B}_{\Sigma_{\mathrm{Beh}}}(M) \cong \mathcal{B}_{\Sigma_{\mathrm{Beh}}}(N)$*;*
- *for each behavioural model* $M \in |\mathbf{BMod}(\Sigma_{\mathrm{Beh}})|$ *and* $N \in |\mathbf{Mod}(\Sigma)|$ *if* $M \equiv N$ *then* N *is behavioural,* $N \in |\mathbf{BMod}(\Sigma_{\mathrm{Beh}})|$.

The second condition in the above definition states that the class of behavioural models over Σ (w.r.t $(\mathcal{R}, \approx)$) is closed under the abstractor equivalence $\equiv$. The first condition is standard, i.e. it comes from the original definition of factorizability in [4].

The abstractor equivalence $\equiv_{Obs,In}$ from [4] is factorizable by Σ_{COL} w.r.t the above definition, if *Obs* is the set of observable sorts of Σ_{COL} (i.e. S_{Obs}) and *In* is the set of loose sorts of Σ_{COL} (i.e. S_{Loose}), see [2] for the definition of observable and loose sorts of a COL-signature.

Throughout the rest of this section we assume that all behavioural signatures $\Sigma_{\mathrm{Beh}} \in |\mathbf{BSign}|$ are isomorphic compatible and all abstractor equivalences considered are isomorphic protecting.

Lemma 2. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature that is weakly regular and* $\equiv$ *is an abstractor equivalence over* Σ*, factorizable by* Σ_{Beh}*. Then for any class of models* $\mathcal{M} \subseteq |\mathbf{Mod}(\Sigma)|$ *the following holds:* $\mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M}) \subseteq \mathrm{Abs}_{\equiv}(\mathcal{M})$*. If moreover* $\mathcal{M}$ *is closed under isomorphism and* $\mathcal{M} \subseteq \mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M})$ *(behavioural consistency) then* $\mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M}) = \mathrm{Abs}_{\equiv}(\mathcal{M})$*.*

Note that in the above lemma, the class $\mathcal{M}$ is not required to be a class of behavioural models. But if $\mathcal{M}$ is behaviourally consistent ($\mathcal{M} \subseteq \mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M})$) then it implies that $\mathcal{M}$ contains only behavioural models.

Proposition 7. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature that is weakly regular and* $\equiv$ *be an abstractor equivalence over* Σ*, factorizable by* Σ_{Beh}*. Let also* $\mathrm{SP}_{\mathrm{Beh}}$ *be a behavioural specification with* $\mathrm{Sig}[\mathrm{SP}_{\mathrm{Beh}}] = \Sigma_{\mathrm{Beh}}$ *and* SP *be a standard specification which corresponds to the behavioural specification* $\mathrm{SP}_{\mathrm{Beh}}$*. We assume that* SP *has isomorphic compatible reduct functors. Then* $\mathrm{Mod}[\mathrm{SP}_{\mathrm{Beh}}] \subseteq \mathrm{Mod}[\mathbf{abstract}\ \mathrm{SP}\ \mathbf{wrt}\ \equiv]$.

The opposite inclusion doesn't hold in general. Consider a basic specification $\mathrm{SP}_{\mathrm{Beh}}$ with an empty set of axioms. If not all models over the signature $\mathrm{Sig}[\mathrm{SP}_{\mathrm{Beh}}]$ are behavioural then $\mathrm{Mod}[\mathbf{abstract}\ \mathrm{SP}\ \mathbf{wrt}\ \equiv] \not\subseteq \mathrm{Mod}[\mathrm{SP}_{\mathrm{Beh}}]$ (SP is the standard specification corresponding to $\mathrm{SP}_{\mathrm{Beh}}$), since on the left-hand side of the inclusion there is the whole class of models over Sig[SP] and on the right-hand side only behavioural models.

The last fact in this subsection concerns relations between behavioural specifications and the abstractor closure of the class of fully abstract models.

Lemma 3. *Let* $\Sigma_{\mathrm{Beh}} = (\Sigma, \mathcal{R}, \approx)$ *be a behavioural signature that is regular and* $\equiv$ *is an abstractor equivalence over* Σ*, factorizable by* Σ_{Beh}*. Then for any class of models closed under isomorphism* $\mathcal{M} \subseteq |\mathbf{Mod}(\Sigma)|$ *the following holds:* $\mathrm{Beh}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M}) = \mathrm{Abs}_{\equiv}(\mathrm{FA}_{\Sigma_{\mathrm{Beh}}}(\mathcal{M}))$*.*

7 Final remarks

In this paper we attempted to define a behavioural semantics for specifications built in an arbitrary logical system formalized as an institution. Although the presented framework covers many institutions of interest it doesn't cover, for example the institution of continuous algebras (an institution of behavioural logic for continuous algebras can be defined if we deal only with the observability concepts). In fact, all assumptions needed to form a behavioural concrete institution from an ordinary concrete institution are quite numerous.

A technical tool used in this work are standard techniques of concrete categories. However, an interesting issue, for further work is to define a behavioural institution based on an institution (without the additional structure of concrete institutions) by a given behaviour operator.

Another important issue for further work is to find a proof system for behavioural specifications basing on a given proof system for ordinary specifications.

References

1. J. Adámek, H. Herrlich, and G. E. Strecker. *Abstract and Concrete Categories*. Pure and Applied Mathematics. Wiley-Interscience Publication, 1990.
2. M. Bidoit and R. Hennicker. On the integration of observability and reachability concepts. In *Proc. 5th Int. Conf. Foundations of Software Science and Computation Structures (FOSSACS 2002)*, volume 2303 of *Lecture Notes in Computer Science*, pages 21–36. Springer, 2002.
3. M. Bidoit and R. Hennicker. The constructor-based observational logic. Research Report, LSV-03-9, 2003.
4. M. Bidoit, R. Hennicker, and M. Wirsing. Behavioural and abstractor specifications. In *Science of Computer Programming*, pages 149–186, 1995.
5. M. Bidoit and A. Tarlecki. Behavioural satisfaction and equivalence in concrete model categories. In *Proc. 21st Int. Coll. Trees in Algebra and Programming*, volume 1059 of *Lecture Notes in Computer Science*, pages 241–256. Springer, 1996.
6. R. Burstall and J. Goguen. Institutions: abstract model theory for computer science. *Journal of the Assoc. for Computing Machinery*, 39:95–146, 1992.
7. A. Tarlecki. Institutions: An abstract framework for formal specifications. In *Algebraic foundations of system specification*, chapter 3, pages 105–130. Springer, 1999.