WARSAW UNIVERSITY

FACULTY OF MATHEMATICS,
INFORMATICS AND MECHANICS

Algebraic observational equivalence and open-map bisimilarity

Ph.D. Thesis

Sławomir Lasota

Thesis supervisor:
prof. dr hab. Andrzej Tarlecki

Warsaw, October 16, 2003

Acknowledgements

First of all, I want to express my gratitude to Andrzej Tarlecki, supervisor of this thesis, for introduction to the area of behavioural equivalence and algebraic specification. I am grateful to Andrzej for always being available and wiling to discuss problems and for his interest in my work. He was a stable source of inspiration during last few years.

I am grateful to Marek Zawadowski for his continuous willingness for explanations and guidance through category theory. I am also grateful to Michel Bidoit, Gian Luca Cattani, Bart Jacobs, Mogens Nielsen and many others for discussions and comments on my work.

I wish to thank Michel Bidoit and Mogens Nielsen, for invitation to Cachan and Aarhus, respectively, and their hospitality there. These visits gave me an opportunity to present my work to a wider audience.

Partial financial support was provided by the KBN grant 8 T11C 046 14 and earlier by the KBN grant 8 T11C 018 11. Part of this work was also supported by LoSSeD workpackage of the CRIT-2 project financed by European ESPRIT and INCO programmes.

Finally, I would like to thank my wife for constant encouragement and for her support during the last few years.

Contents

Chapter 1

Introduction

The concept of *behavioural equivalence* is a fundamental notion in programming methodology, as it seems to capture appropriately the "black box" character of data abstraction. It also plays a central role in the process of stepwise refinement of specifications, where each step embodies some design decisions, under the constraint that the required behaviour must be ensured. This applies to two different notions: *observational equivalence* of algebras that are widely accepted as models of sequential program modules in the algebraic approach to system specification [GTW78, EM85], as well as *bisimulation equivalence* (bisimilarity) of processes, modeling concurrent or nondeterministic systems[1].

Roughly speaking, two algebras A and B over a signature Σ are *observationally equivalent* with respect to a distinguished set OBS of *observable sorts* if all computations (using operations from Σ) yielding a result in an observable sort produce the same output in both A and B. This means that A and B are indistinguishable by means of computations with observable results. This idea goes back to [Mor68], [Hoa72] and [GGM76]; later it was widely studied mainly in the case of standard algebras, e.g. in [Rei81, GM82, SW83, Rei85, HW85, NO88, ST87, Sch87, BB91]; but also in some other frameworks, like regular algebras [BT95] and partial algebras [BT96].

Another approach to behavioural equivalence is the notion of *bisimilarity* of concurrent processes [Mil89], introduced in [Par81]. Two processes are bisimilar if there exists a relation of *bisimulation* between them, indicating how every action of one process can be simulated in the other one and vice versa. Bisimilarity is thus a formalization of indistinguishability of processes by observing their actions.

Both observational equivalence and bisimilarity play a crucial role in formal models of sequential and concurrent systems, respectively. The undeniable conceptual analogies (despite technical differences) between both approaches are well recognized and much work has been done on linking the two notions, e.g. applying methods based on bisimulation to sequential programs [AO93, Gor95]. Results presented in this thesis are a continuation of this stream of research.

During the last decade, as a result of intensive effort in attempting to explain relationships between both approaches, generalizations of both concepts were proposed. The main advantage of such an abstract (re-)formulation of well-known notions is that it allows us to

[1] In this thesis by *observational equivalence* we mean a behavioural equivalence of algebras whereas term *behavioural equivalence* will be used more generally, for observational equivalence as well as for bisimilarity. We should mention here that this terminology is not well established in the literature and both terms, behavioural equivalence and observational equivalence, are often confused.

capture in a uniform way many different, but conceptually similar situations. We review briefly these results here – formal definitions can be found in the following sections.

A categorical generalization of bisimulation was formulated in terms of *open maps* (*open morphisms*) [JNW93]; this enables a uniform definition of bisimulation equivalences across a range of different models for parallel computations. This setting turned out appropriate for defining, among others, strong and weak bisimilarity [Mil89], trace equivalence [Hoa85], testing equivalence [Hen88] and a strong (hereditary) history preserving bisimulation of event structures [Bed91, JNW93] (see [CN95], [JNW93] for an overview). Open maps can be understood as arrows witnessing a bisimulation, hence two objects A and B in a category are bisimilar if they are related by a span of open maps:

$$\begin{tikzcd} & C \arrow[dl] \arrow[dr] & \\ A & & B \end{tikzcd}$$

which can be understood as an abstract counterpart of a bisimulation relation. More precisely, open maps are defined tentatively to be those morphisms which satisfy a *computation-lifting property*, where a computation of a process is modeled by a morphisms from any of distinguished *path objects* (observation objects). Thus bisimilarity is determined (induced) by the choice of a subcategory $\mathbb{P}$ of path objects. This concept is related to open morphisms in a topos of set-valued functors from $\mathbb{P}^{op}$ [JM94].

The computation-lifting property amounts, in many cases, to the requirement that open maps are functional bisimulations. This idea was most directly captured by another abstract generalization of bisimilarity, based on the notion of coalgebra. Roughly speaking, a coalgebra is dual to algebra in the following sense. While algebra structure on a set X can be given by a function $\Sigma(X) \to X$, for the endofunctor Σ induced by an algebraic signature, a coalgebra is defined as a map $X \to B(X)$, for an endofunctor B giving, for a set X, a set $B(X)$ of possible "behaviors". Coalgebra morphisms, i.e., behaviour preserving maps, turn out to capture appropriately the concept of bisimilarity – they correspond to functional bisimulations, and a relation of bisimulation is represented in this setting, again, by a span of coalgebra morphisms.

A dual approach was used for defining behavioural equivalence for algebras. Roughly speaking, two algebras A and B are *behaviourally equivalent* if their quotients by *indistinguishability relations* $\sim_A$ and $\sim_B$ are isomorphic [Rei81, BHW95], that is if they are related by a co-span of quotient projections:

$$\begin{tikzcd} A \arrow[dr] & & & B \arrow[dl] \\ & A/_{\sim_A} \arrow[r, phantom, "\cong"] & B/_{\sim_B} & \end{tikzcd}$$

A relevant example here is an observational equivalence; in this case the indistinguishability $\sim_A$ in a many-sorted algebra A is determined by a subset of *observable sorts* and consequently by the set of *observable contexts*. The latter consists usually of all terms of an observable sort with variables of only observable sorts apart from a special one, a *context variable*, reserved for the value being observed in an algebra. Such a context term (together with a valuation of its ordinary variables) corresponds to a (finite) computation in the algebra, starting at the observed value and resulting in some output value. Then, the indistinguishability $\sim_A$ relates those values a, a' that taken as input to each such computation yield the same result: $t[a] = t[a']$, for all observable contexts $t[\_]$ (and all valuations).

We say that behavioural equivalence is *factorized* by the family of indistinguishability relations [BHW95]. This scheme is appropriate for various categories of algebras, including standard, partial and regular algebras. Moreover, a generalization of the above approach is possible in an arbitrary *concrete category* [BT96], assumed to support some algebraic constructions like subobject and quotient.

The aim of this thesis is mainly to investigate relationships between the observational equivalence and bisimilarity. The principal motivation for such study is the discrepancy between

obvious analogies at the conceptual level

and

substantial differences at the technical, mathematical level,

between different approaches to behavioural equivalences. In our investigation we intensively use frameworks described above, and exploit their generality enabling to state certain connections between different approaches. Moreover, as far as we know, for the first time abstract data types and processes are considered commonly in the context of open maps.

1.1 Content of this thesis

We briefly overview here the contents of remaining chapters. In addition, each chapter contains its own introduction, explaining in more detail the material it contains.

In the rest of Chapter 1 we recall the rather standard ideas, concepts and facts on which our work is based.

Chapter 2 The most important corollary from results stated in Chapter 2 is that observational equivalences for some classes of algebras (standard, partial and regular algebras are considered) can be captured in the setting of open maps, and thus can be thought of as abstract bisimilarities. Hopefully, the results can be easily moved to other (algebraic) frameworks, since they are formulated abstractly for an arbitrary adjunction and multiadjunction. In all cases, path subcategories consist of (multi)free objects w.r.t. the forgetful functor giving the carrier sets of observable sorts. To be able to deal also with model categories where morphisms "weakly" preserve structure of models, like category of partial algebras with weak homomorphisms, we develop the framework of pre-adjunction and state some links with the concept of multiadjunction.

We identify some observation subcategories, corresponding to "finitary" observations, sufficient to induce the observational equivalences. This can be very useful in proving observational equivalence of algebras.

Moreover, we observe that the bisimilarities of algebras are *deterministic*, in the sense that a computation (in a codomain of an open map) is always lifted uniquely (to a computation in its domain). This is in contrast to bisimilarities between models for concurrent or nondeterministic systems like transition systems or event structures.

Chapter 3 Chapter 3 completes, in a way, the results of Chapter 2: roughly speaking, the opposite direction of the analogy is investigated. We show that algebraic techniques are applicable to bisimulation of processes as well, by formulating a condition sufficient for the open-map bisimilarity to be factorizable. We prove that if the greatest open congruences exist on each object, than the induced bisimilarity is factorizable. This general result gives an interesting conclusion that a wide class of bisimilarities (including those for transition systems and presheaves) can be equivalently defined by co-spans of open quotient projections, similarly as algebraic behavioural equivalences. Our motivations are two-fold here: first, we search for uniformity for both open-map bisimilarity and factorizable observational equivalences of algebras; second, an inspiration is strong bisimilarity of transition systems, factorized by co-spans of zig-zag morphisms.

We also formulate the necessary and sufficient condition for the factorization: quotients by all maximal open congruences are isomorphic. This means that instead of a single greatest open congruence for each object, there may exist a number of distinct maximal open congruences, yielding isomorphic quotients. As an example of such a situation, we analyze in detail a subclass of simple event structures.

Chapter 4 We relate the abstract notion of bisimulation, based on open maps, with another widely studied approach, using coalgebras and their morphisms. We show that open maps correspond to coalgebra morphisms for a suitable endofunctor in the category of many-sorted sets. This demonstrates that the notion of open-map bisimilarity is of essentially coalgebraic nature. A crucial role in our development is played by a category of presheaves, which we show to correspond to the subcategory of *consistent* coalgebras with lax cohomomorphisms; in addition, open maps between presheaves are precisely coalgebra morphisms.

In order to relate open-map bisimilarity and coalgebraic bisimulation equivalence, one must consider spans of appropriate morphisms. It turns out that the category of coalgebras (or presheaves) contains more objects (hence more spans) than standard model categories. This is why a relevant question arises: which presheaves correspond to models we want to consider?

As a next step in relating algebraic observational equivalence with open-map bisimilarity, we answer this question for many-sorted standard and partial algebras. The characterizations obtained here are in close analogy to Lawvere's functorial presentation of models of algebraic theories and to standard representation results for locally finitely presentable categories.

Chapter 5 As a nontrivial example of an algebraic framework where behavioural ideas may be applied, we consider regular algebras, differing from the standard algebras in one respect: they allow one to additionally model "infinite" datatypes, like streams. This is achieved by providing in regular algebras the least fixed points of operators definable by terms. Regular algebras require a separate treatment, since their structure differs substantially from standard or partial algebras: they have partially ordered carrier sets and the fixed points are obtained as the least upper bounds of appropriate approximation chains w.r.t. that order.

The main topic of this chapter is an attempt to develop a framework for behavioural implementation in the setting of regular algebras. A general methodology is proposed, as an adaptation of the constructor behavioural implementation. Moreover, a condition sufficient for correctness of an implementation step is found.

Furthermore, this chapter contains some technical results concerning regular algebras and observational equivalence in this setting. Two different definitions of regular algebras are compared and shown equivalent and a simpler (re)formulation of the definition is proposed. Finally, we notice that observational indistinguishability in a regular algebra is determined by finitary contexts – surprisingly, even when non-trivial approximation chains exist in carrier sets of observable sorts.

In the following, some prior knowledge of category theory is assumed: for the categorical notions used in this thesis, like functor, (co)limit, and adjunction, we refer to [Mac71].

The material of this thesis is based on some earlier papers by the author ([Las97, Las98a, Las98b, Las99]); however, the content of these publications is essentially extended, suitably adapted and consolidated here in order to make this text possibly complete, self-contained and understandable. Moreover, some new material is included (for instance Sections 2.3, 3.7, 3.8, 4.5.2, 4.5.3 and Chapter 5).

1.2 Related work

An axiomatization of open maps was studied for instance in [JM94]. Open maps were used for the first time to abstractly (re)formulate bisimulation equivalence by Joyal, Winskel and Nielsen in [JNW93]. Since this seminal paper, there have been many attempts to exploit the uniformity offered by this setting. The authors of [CN95] demonstrated applicability of open maps for defining several relevant equivalences of processes, for instance testing equivalence, weak bisimilarity, etc.; in [HN98] timed bisimulation was captured in this setting. Among other equivalences, in [NW96] open maps were applied to Petri nets and in [CSW97] several bisimulations in π-calculus were redefined by open maps in a presheaf model. Recently, a uniform way of defining weak equivalences, like Milner's weak bisimulations, was presented in [CFW99].

Much of the recent work concerning open maps has been concentrated around a uniform treatment of different models of concurrent systems, mainly in the framework of presheaves, for instance [CW96], [CPW98]. The long-term challenge is a systematic study of semantic domains for concurrency, similarly as it was done in domain theory for sequential languages, see for example [CFW98].

Independently, a stream of research has been aimed at explaining relationships between observational equivalence and bisimilarity. As some examples of work in this field we should mention [Gor95], where a characterization of contextual equivalence in PCF was given in a bisimulation-like manner by means of greatest fixed points; and [AO93], where authors proposed an approach to semantics of lazy λ-calculus, inspired by bisimulation. Moreover, in [Mal96], [Cir97] some links were observed between bisimulation and observational equivalence for hidden-sorted algebras, viewed as coalgebras – this was done under the assumption that a signature is monic in non-observable sorts.

There were several attempts to develop techniques motivated by bisimulation in the field of algebraic specifications. Probably most directly a concept of bisimulation is present in the work of Schoet [Sch87, Sch90], where the notion of a correspondence was introduced, as a counterpart of bisimulation. More implicitly, the bisimulation-like flavor can be found in the notion of contextual indistinguishability in an algebra ([Rei81]) and in the related proof techniques, like context induction [Hen89, Hen91]; see also [BH95]. An attempt to commonly consider abstract data types and processes was performed in [AGR92], in the framework of so called observational structures. Interesting results in application of coalgebraic techniques to recursively defined datatypes were obtained in [Pit93, Pit94, Fio96], especially concerning mixed induction/coinduction proof techniques. Among multiple papers on coalgebraic approach to specification of state-base systems we mention [Jac96b, Jac97, Rei95].

1.3 Algebraic observational equivalence

We start by recalling the notions of observational equivalence for standard, partial and regular algebras, which are investigated in the sequel. Standard algebras, consisting of a set of values and (total) operations on them, are widely accepted models for abstract data types [Gra79], [BL70], [GTW78]. In some applications such models are not quite satisfactory; for example, when the totality of operations is too restrictive, partial algebras are used [Gra79, BW82], in which operations are partial functions. In many applications a kind of "infinitary" data is needed, for instance to specify a programming language with possibly infinite computations or to ensure that recursive definitions are meaningful – to this aim continuous algebras were studied [GTW77, TW86, ANR95], where carriers are cpo's and operations are continuous maps. The category of continuous algebras lacks some crucial algebraic properties (for example, quotients do not compose there), hence the setting of *regular algebras* was proposed [Tiu78, Tiu79, GP83], in which not all increasing chains must have least upper bounds, but only those definable by finite terms. In [BT95] regular algebras are studied in the context of algebraic specifications.

Despite such a variety of algebras, the observational equivalence is defined uniformly for all of them by means of the greatest congruence that is the identity on the carriers of some distinguished observable sorts. Two algebras are observationally equivalent if their quotients by such congruences are isomorphic. This uniformity was captured in [BT96], where a more general framework of *concrete category* was introduced, enabling a definition of observational equivalence. Its special cases are categories of standard and regular algebras as well as partial algebras with strong homomorphisms. We postpone introducing this abstract setting to Section 2.3, where also some comments are included about possible generalizations of results of this chapter. Section 2.3 contains also some definitions used in the next chapter.

At first, mainly to fix notation and terminology, we recall basic notions concerning standard, partial and regular algebras, despite that many of them, like those of (standard, partial, regular) subalgebra, congruence, quotient, etc., are instances of general concepts expressible in arbitrary concrete category. The definitions concerning standard many-sorted algebras are widely accepted [BL70], [GTW78]. Definitions of partial algebras are based on [BW82] and [BT96], while the presentation of regular algebras is a slight modification of [BT95].

1.3.1 Standard algebras

For a set S, by Set^S we denote throughout this thesis the category of S-sorted sets with S-sorted functions. We also use the term *many-sorted* sets and functions, when the set S is obvious. For an S-sorted set X (S-sorted function f) and $s \in S$, X_s (f_s) denotes the s-th component of X (f). Similarly, for a subset $R \subseteq S$ of sorts, by X_R we mean an R-sorted set consisting of those X_s with $s \in R$.

A signature is a pair $\langle S, F\rangle$, where $S = sorts(\Sigma)$ is a set of sorts and F a set of operation names with indicated arities (sequences of sorts in S) and result sort. Let Σ be a fixed signature in the following, unless stated otherwise. Σ-algebra consists of an S-sorted set, denoted by $|A|$ (*the carrier* of A) and for each $f : s_1 \times \ldots \times s_n \to s$ in Σ, of a function $f_A : |A|_{s_1} \times \ldots \times |A|_{s_n} \to |A|_s$. A Σ-homomorphism $h : A \to B$ is a S-sorted function $|h| : |A| \to |B|$ which preserves operations, i.e., for each $f : s_1 \times \ldots \times s_n \to s$ in Σ,

$$|h|_s(f_A(a_1, \ldots, a_n)) = f_B(|h|_{s_1}(a_1), \ldots, |h|_{s_n}(a_n)).$$

Let Alg_Σ denote the category of standard Σ-algebras with Σ-homomorphisms. The mapping $|\_|$ is a forgetful functor $|\_| : Alg_\Sigma \to Set^S$. One can analogously define a forgetful functor $|\_|_R : Alg_\Sigma \to Set^R$, for every subset of sorts $R \subseteq S$.

A congruence on A is a many-sorted equivalence relation $\sim \subseteq |A| \times |A|$ compatible with operations, that is, such that for each $f : s_1 \times \ldots \times s_n \to s$ in Σ and $a_1, a_1' \in |A|_{s_1}, \ldots, a_n, a_n' \in |A|_{s_n}$, if $a_1 \sim_{s_1} a_1', \ldots, a_n \sim_{s_n} a_n'$ then $f_A(a_1, \ldots, a_n) \sim_s f_A(a_1', \ldots, a_n')$. Given a congruence relation $\sim$, the quotient algebra $A/_\sim$ is defined by $|A/_\sim| = |A|/_\sim$ and $f_{A/_\sim}([a_1]_\sim, \ldots, [a_n]_\sim) = [f_A(a_1, \ldots, a_n)]_\sim$.

A subalgebra of A is any algebra B the carrier of which is a subset of $|A|$ closed under operations and operations of which are equal to the operations of A restricted to $|B|$.

The S-sorted set $|T_\Sigma(X)|$ of Σ-terms with variables from X is defined inductively as usual, for any X from Set^S. The relation of satisfaction of an equation $t = t'$, where $t, t' \in T_\Sigma(X)$, in Σ-algebra A under a valuation $v : X \to |A|$, denoted by $A[v] \models t = t'$, is defined by $A[v] \models t = t'$ iff $t_{A[v]} = t'_{A[v]}$, where $t_{A[v]}$ denotes the value of term t in algebra A under valuation $v : X \to |A|$.

The set $|T_\Sigma(X)|$, where X is in Set^S, can be equipped in the canonical way with operations of Σ to form a Σ-algebra $T_\Sigma(X)$. Namely, for each $f : s_1 \times \ldots \times s_n \to s$ in Σ, we define $f_{T_\Sigma(X)}(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$. An algebra $T_\Sigma(X)$ is a free object over X w.r.t. the forgetful functor $|\_|$, hence every function (valuation) $v : X \to |A|$ has a unique extension $v^\# : T_\Sigma(X) \to A$ in Alg_Σ; moreover $T_\Sigma(\_)$ can be extended to a functor being the left adjoint to $|\_|$.

Notation: We will overload the notation $T_\Sigma(X)$ using it also for X from Set^R, where $R \subseteq S$; thus $T_\Sigma(\_)$ will denote the left adjoint to any forgetful functor $|\_|_R : Alg_\Sigma \to Set^R$, returning the carriers of sorts from R. Generally, we will identify category Set^R with the full subcategory of Set^S containing all objects satisfying $X_r = \emptyset$ for all $r \notin R$, ignoring the subtle difference. Moreover, we will use the same notation $|\_|_R$ also for the forgetful functors from categories of partial and regular algebras.

1.3.2 Partial algebras

Partial Σ-algebras are like standard Σ-algebras except that their operations are partial functions. A (weak) partial Σ-homomorphism $h : A \to B$ is an S-sorted function $|h| : |A| \to |B|$ preserving definedness of operations, i.e., for each $f : s_1 \times \ldots \times s_n \to s$ in Σ and $a_1 \in |A|_{s_1}, \ldots, a_n \in |A|_{s_n}$, if $f_A(a_1, \ldots, a_n)$ is defined then $f_B(|h|_{s_1}(a_1), \ldots, |h|_{s_n}(a_n))$ is defined and $|h|_s(f_A(a_1, \ldots, a_n)) = f_B(|h|_{s_1}(a_1), \ldots, |h|_{s_n}(a_n))$. Let $PAlg^{\mathrm{W}}_\Sigma$ denote the category of partial Σ-algebras with weak partial Σ-homomorphisms.

A partial Σ-homomorphism is *strong* if not only it preserves but also reflects definedness of operations, i.e., $f_A(a_1, \ldots, a_n)$ is defined if and only if $f_B(|h|_{s_1}(a_1), \ldots, |h|_{s_n}(a_n))$ is defined, for each $f : s_1 \times \ldots \times s_n \to s$ in Σ and $a_1 \in |A|_{s_1}, \ldots, a_n \in |A|_{s_n}$. The subcategory of partial algebras with strong partial Σ-homomorphisms is denoted by $PAlg_\Sigma$.

A partial subalgebra of A is any partial algebra B the carrier of which is a subset of A closed under operations and operations of which are equal to operations of A restricted to $|B|$.

A (strong) congruence on A is a many-sorted equivalence relation $\sim \subseteq |A| \times |A|$, such that for each operation $f : s_1 \times \ldots \times s_n \to s$ in Σ and $a_1, a'_1 \in |A|_{s_1}, \ldots, a_n, a'_n \in |A|_{s_n}$, whenever $a_1 \sim_{s_1} a'_1, \ldots, a_n \sim_{s_n} a'_n$ then $f_A(a_1, \ldots, a_n)$ and $f_A(a'_1, \ldots, a'_n)$ are both undefined or both defined and $f_A(a_1, \ldots, a_n) \sim_s f_A(a'_1, \ldots, a'_n)$. Given a congruence relation $\sim$, the quotient algebra $A/_\sim$ is defined by $|A/_\sim| = |A|/\sim$ and $f_{A/_\sim}([a_1]_\sim, \ldots, [a_n]_\sim)$ is defined iff $f_A(a_1, \ldots, a_n)$ is defined, and then $f_{A/_\sim}([a_1]_\sim, \ldots, [a_n]_\sim) = [f_A(a_1, \ldots, a_n)]_\sim$.

The quotient projection and the inclusion of subobjects are strong partial homomorphisms, hence definitions above correspond to general definitions in a concrete category ([BT96]) applied to $PAlg_\Sigma$ (see Section 2.3).

In the case of partial algebras, the value $t_{A[v]}$ of a term under a valuation $v : X \to |A|$ may be undefined, as operations are allowed to be partial functions. The relation of satisfaction of an equation $t = t'$, for $t, t' \in T_\Sigma(X)$, in a Σ-algebra A under a valuation $v : X \to |A|$ is defined as follows: $A[v] \models t = t'$ iff either both $t_{A[v]}$ and $t'_{A[v]}$ are undefined or both are defined and $t_{A[v]} = t'_{A[v]}$.

1.3.3 Regular algebras

An *ordered Σ-algebra* is a standard Σ-algebra equipped with a partial order $\leq^A_s$, for each sort $s \in S$ and a distinguished element $\perp^A_s \in |A|_s$, for each $s \in S$.

The set of regular Σ-terms $|T^\mu_\Sigma(X)|$ with variables from X is defined inductively as usual, with the only additional case:

$$\text{for any } t \in |T^\mu_\Sigma(X \cup \{z : s\})|_s \text{ and a variable } z \text{ of the same sort } s,\ \mu z.t \in |T^\mu_\Sigma(X)|.$$

Similarly, the inductive definition of the value $t_{A[v]}$ of a term under a valuation $v : X \to |A|$ needs one more case. For any $t \in |T^\mu_\Sigma(X \cup \{z : s\})|_s$, where z is a distinguished variable of sort s, put:

- $t^0_{A[v]}(\perp) = \perp^A_s$,
- for $i \geq 0$, $t^{i+1}_{A[v]}(\perp) = t_{A[v_i]}$, where $v_i : X \cup \{z : s\} \to |A|$ extends v by $v_i(z) = t^i_{A[v]}(\perp)$.

Now, $(\mu z.t)_{A[v]}$ is defined if $t^i_{A[v]}(\perp)$ are defined, for all $i \geq 0$, $t^i_{A[v]}(\perp) \leq t^{i+1}_{A[v]}(\perp)$, for all $i \geq 0$, and the least upper bound $\bigsqcup_{i \geq 0} t^i_{A[v]}(\perp)$ exists in $|A|_s$. Then $(\mu z.t)_{A[v]} = \bigsqcup_{i \geq 0} t^i_{A[v]}(\perp)$.

An ordered algebra A is a regular Σ-algebra if it satisfies following two conditions:

- (completeness): for all terms $t \in |T^\mu_\Sigma(X)|$ and valuations $v : X \to |A|$, the value $t_{A[v]}$ of t under v in A is defined,
- (continuity): for all terms $t \in |T_\Sigma(X \cup \{y : s\})|_{s'}$, $q \in |T^\mu_\Sigma(X \cup \{z : s\})|_s$ and valuations $v : X \to |A|$, $t_{A[v_i]} \leq t_{A[v_{i+1}]}$, for $i \geq 0$, and $t_{A[v']} = \bigsqcup_{i \geq 0} t_{A[v_i]}$, where $v' : X \cup \{y : s\} \to |A|$ extends v by $v'(y) = (\mu z.q)_{A[v]}$ and $v_i : X \cup \{y : s\} \to |A|$ extends v by $v_i(z) = q^i_{A[v]}(\perp)$, for $i \geq 0$.

Note that continuity is required only for *algebraic* terms $t \in |T_\Sigma(X \cup \{y : s\})|_{s'}$, i.e., those not containing symbol μ. A seemingly stronger condition (considered in [BT95]), concerning all regular terms $t \in |T^\mu_\Sigma(X \cup \{y : s\})|_{s'}$, turns out to be equivalent to the above one – a more detailed discussion on this is deferred to Chapter 5.

By a regular Σ-homomorphism $h : A \to B$ we mean any function $|h| : |A| \to |B|$ such that for all terms $t \in |T^\mu_\Sigma(X)|$ and valuations $v : X \to |A|$, $|h|(t_{A[v]}) = t_{B[|h| \circ v]}$. Regular algebras together with regular homomorphisms form a category, which we call $RAlg_\Sigma$. A regular subalgebra of A is any regular algebra B the carrier of which is a subset of A and for all terms $t \in |T^\mu_\Sigma(X)|$ and valuations $v : X \to |B|$, $t_{B[v]} = t_{A[v]}$. It is easy to see that all operations of B are restrictions of operations of A to $|B|$ and moreover $\perp^B_s = \perp^A_s$ for each sort s.

The kernel of a many-sorted function $f : X \to Y$ is a set of pairs $\{(x, x') \mid f(x) = f(x')\}$. By a regular congruence on A we mean a kernel of any regular Σ-homomorphism $h : A \to B$. More explicitly, regular congruences can be characterized as follows. A relation $\sim \subseteq |A| \times |A|$ is a congruence iff $\sim = \sqsubseteq \cap \sqsubseteq^{-1}$, for some *pre-congruence* $\sqsubseteq$, where a pre-congruence is any pre-order on $|A|$ satisfying:

1. for all terms $t \in |T_\Sigma(X \cup \{y : s\})|_{s'}$, $q \in |T^\mu_\Sigma(X \cup \{z : s\})|_s$ and valuations $v : X \to |A|$, $\{t_{A[v_i]}\}_{i \geq 0}$ is a chain w.r.t. $\sqsubseteq$ with the least upper bound $t_{A[v']}$, where valuation $v' : X \cup \{y : s\} \to |A|$ extends v by $v'(y) = (\mu z.q)_{A[v]}$ and $v_i : X \cup \{y : s\} \to |A|$ extends v by $v_i(z) = q^i_{A[v]}(\perp)$, for $i \geq 0$,[2]

2. the equivalence $\sqsubseteq \cap \sqsubseteq^{-1}$ induced by $\sqsubseteq$ is preserved by the operations (i.e., is a standard congruence).

In particular, instantiating t in 1 with a single variable, we get: for all recursors $q \in |T^\mu_\Sigma(X \cup \{z : s\})|_s$ and valuations $v : X \to |A|$, $\{q^i_{A[v]}(\perp)\}_{i \geq 0}$ is a chain w.r.t. $\sqsubseteq$ with the least upper bound $(\mu z.q)_{A[v]}$.

Given a regular congruence $\sim$, the quotient algebra $A/\sim$ is defined (for each sort s) by $|A/\sim|_s = |A|_s/\sim$, $\perp^{A/\sim}_s = [\perp^A_s]$, $[a]_\sim \leq^{A/\sim}_s [a']_\sim \Leftrightarrow a \sqsubseteq a'$ (where $\sqsubseteq$ is a pre-congruence inducing $\sim$) and $f_{A/\sim}([a_1]_\sim, \ldots, [a_n]_\sim) = [f_A(a_1, \ldots, a_n)]_\sim$. For details we refer to [BT95].

1.3.4 Observational equivalence

There have been a number of different formalizations of the concept of behavioural equivalence od algebras [Rei81, SW83, HW85, ST87, Sch87, NO88]. For an overwiev and further references see [Kna93]. In the following we concentrate on an observational equivalence induced by a subset of observable sorts.

Throughout this thesis let us fix a subset $OBS \subseteq S$ of *observable sorts*. By *observational morphisms* we mean the (standard, partial, regular) Σ-homomorphisms which are bijective on observable sorts.

In this section we define the observational equivalence for standard, partial and regular algebras. The formal definition will be informally motivated in the next section, where we argue that the notion of observational equivalence is crucial for correctness of implementation.

[2] In [BT95] a stronger requirement on pre-congruence was assumed, for all regular terms $t \in |T^\mu_\Sigma(X \cup \{y : s\})|_{s'}$, similarly as in continuity condition above. Along the same lines, the stronger requirement is equivalent to ours – we will further expand on this in Chapter 5.

Standard, partial and regular algebras are treated uniformly here, and called commonly Σ-algebras in the following. For a Σ-algebra A and $X \subseteq |A|$, by $\langle X\rangle_A$ we denote the least (standard, partial, regular) subalgebra of A whose carrier includes X. $\langle X\rangle_A$ will be called *the subalgebra of A generated by X*. By a *partial congruence* on a Σ-algebra A we mean any congruence on a subalgebra of A.

For a Σ-algebra A we define the partial congruence $\sim_A^{OBS}$ as follows. The domain of $\sim_A^{OBS}$ is the carrier $|A_{OBS}|$ of $A_{OBS} = \langle |A|_{OBS}\rangle_A$. Then we define $\sim_A^{OBS}$ as the greatest congruence on A_{OBS} being identity on observable sorts. The partial equivalence $\sim_A^{OBS}$ is a relation of *indistinguishability* of elements of A.

Definition 1.3.1 *An observational equivalence of (standard, partial, regular) algebras is an equivalence relation $\equiv_{OBS}$* factorized *by the family*[3] $\{\sim_A^{OBS}\}_{A\in|Alg_\Sigma|}$ *of partial congruences, i.e., defined by*[4]*:*

$$A \equiv_{OBS} B \quad \textit{iff} \quad A/_{\sim_A^{OBS}} \cong B/_{\sim_B^{OBS}}.$$

The quotient $A/_{\sim_A^{OBS}}$ of A by the partial congruence $\sim_A^{OBS}$ is defined as the quotient of A_{OBS} by $\sim_A^{OBS}$.

There exist explicit characterizations of $\sim_A^{OBS}$ as the relations of indistinguishability by means of observable Σ-contexts. We review here briefly these characterizations (for details, see [BHW95] for standard algebras, [BT96] for partial algebras and [BT95] for regular algebras).

In the following, let X denote some OBS-sorted set of variables. By a Σ-*context* of sort s' on sort s we mean any term $\gamma \in |T_\Sigma(X \cup \{z_s : s\})|_{s'}$ ($\gamma \in |T_\Sigma^\mu(X \cup \{z_s : s\})|_{s'}$ in the case of regular algebras), where z_s is a special, distinguished variable of sort s such that $z_s \notin X$. Note that z_s, for $s \notin OBS$, are the only variables of non-observable sorts appearing in contexts. A special role is played by *observable contexts*, i.e., those γ which are of observable sort (that is, $s' \in OBS$). For any Σ-algebra A, Σ-context γ on sort s, valuation $v : X \to |A|_{OBS}$ and value $a \in |A|_s$, we will write $\gamma_{A[v]}(a)$ for $\gamma_{A[v_a]}$ where v_a extends v by $v(z_s) = a$.

The indistinguishability relations can be defined as follows (the references are given to the papers where these facts appear in this particular technical form; similar characterizations for standard algebras were given elsewhere, starting from [Rei81]):

Theorem 1.3.2 (standard algebras, [BHW95], regular algebras, [BT95]) *For any two elements $a, a' \in |A|_s$ of a standard (regular) algebra A, $a \sim_A^{OBS} a'$ if and only if for all valuations $v : X \to |A|_{OBS}$ and all observable contexts γ,*

$$\gamma_{A[v]}(a) = \gamma_{A[v]}(a').$$

Theorem 1.3.3 (partial algebras, [BT96]) *For any two elements $a, a' \in |A|_s$ of a partial algebra A, $a \sim_A^{OBS} a'$ if and only if for all valuations $v : X \to |A|_{OBS}$,*

- *for all contexts γ, $\gamma_{A[v]}(a)$ is defined iff $\gamma_{A[v]}(a')$ is defined,*
- *for all observable contexts γ such that $\gamma_{A[v]}(a)$ (and hence also $\gamma_{A[v]}(a')$) is defined,*

$$\gamma_{A[v]}(a) = \gamma_{A[v]}(a').$$

[3]In the following $|\mathcal{C}|$ denotes the class of all objects of a category $\mathcal{C}$; symbol $|\_|$ is thus deliberately overloaded.

[4]Throughout the paper, the symbol $\cong$ denotes an isomorphism.

Observational equivalences $\equiv_{OBS}$ in the categories of standard, partial and regular algebras, respectively, have the following characterizations, used in the sequel:

Theorem 1.3.4 (standard algebras, [BHW95], regular algebras, [BT95]) *Standard (regular) algebras A and B are observationally equivalent iff for some OBS-sorted set X of variables and surjective valuations $v_A : X \to |A|_{OBS}$ and $v_B : X \to |B|_{OBS}$, we have: for all terms (regular terms) $t, t' \in |T_\Sigma(X)|_o$ ($t, t' \in |T_\Sigma^\mu(X)|_o$) of an observable sort $o \in OBS$ with variables from X,*

$$A[v_A] \models t = t' \quad \textit{iff} \quad B[v_B] \models t = t'.$$

Theorem 1.3.5 (partial algebras, [BT96]) *Partial algebras A and B are observationally equivalent iff for some OBS-sorted set X and surjective valuations $v_A : X \to |A|_{OBS}$ and $v_B : X \to |B|_{OBS}$, we have:*

- *for all terms $t \in |T_\Sigma(X)|$, $t_{A[v_A]}$ is defined iff $t_{B[v_B]}$ is defined,*
- *for all terms $t, t' \in |T_\Sigma(X)|_o$ of an observable sort $o \in OBS$, such that both $t_{A[v_A]}$ and $t'_{A[v_A]}$ are defined (hence also $t_{B[v_B]}$ and $t'_{B[v_B]}$ are defined),*

$$A[v_A] \models t = t' \quad \textit{iff} \quad B[v_B] \models t = t'.$$

Note the similarity of the characterizations, with only slight differences between different algebraic settings. An attempt to spell this out in a uniform manner is presented in Section 2.3.

A proof method for showing observational equivalence was proposed in [Sch87]. This approach is based on the notion of an *observational correspondence*:[5]

Definition 1.3.6 *For two partial Σ-algebras A and B, a many-sorted relation $\sim \subseteq |A| \times |B|$ is an* observational Σ-correspondence *(w.r.t. a set OBS of observable sorts) if*

- *$\sim$ is a bijection on each observable sort,*
- *for each operation $f : s_1 \times \ldots \times s_n \to s$ in Σ, elements $a_1, \ldots, a_n$ in $|A|$ and elements $b_1, \ldots, b_n$ in $|B|$ of appropriate sorts, whenever $a_1 \sim b_1, \ldots, a_n \sim b_n$, then*
  - *$f^A(a_1, \ldots, a_n)$ is defined if and only if $f^B(b_1, \ldots, b_n)$ is defined,*
  - *if $f^A(a_1, \ldots, a_n)$ (and hence also $f^B(b_1, \ldots, b_n)$) is defined, then*

$$f^A(a_1, \ldots, a_n) \sim f^B(b_1, \ldots, b_n).$$

Correspondences are a useful tool for proving observational equivalence of algebras, according to the following fact:

Theorem 1.3.7 ([Sch87]) *Two partial algebras A and B are observationally equivalent iff they are related by an observational correspondence.*

A similar characterization holds also for standard algebras – in this case, however, the definition of a correspondence can be essentially simplified. An attempt to adapt this concept to the setting of regular algebras is presented in Section 5.3. In the following, by a correspondence we mean an observational correspondence of partial (or standard) algebras. When the confusion is possible, e.g. while considering correspondences of regular algebras, we will use prefixes, like standard (regular) correspondence.

[5]Strong V-correspondence, in the terminology of [Sch87].

1.3.5 Behavioural implementation

A concept of observational equivalence plays a crucial role in the process of step-wise refinement. It allows one to consider possibly large class of acceptable realizations of a specification, under the only assumption that the behavior of the implementation conforms to the specification requirements. These ideas are formalized below by the notion of behavioural constructor implementation [ST97].

First, let us look at a standard example of a specification of stacks of integers, to illustrate a behavioural approach to implementation:

specification `STACKS` **extends** `INT, BOOL` **by**
sorts
$\quad stack;$
operations
$\quad empty : stack;$
$\quad push(\_,\_) : int \times stack \to stack;$
$\quad pop(\_) : stack \to stack;$
$\quad top(\_) : stack \to int;$
$\quad empty?(\_) : stack \to bool;$
axioms
$\quad empty?(empty) = true,$
$\quad empty?(push(n,s)) = false,$
$\quad top(push(n,s)) = n,$
$\quad pop(push(n,s)) = s.$

Consider the following candidate A for implementation of this specification, which realizes a stack as an infinite array of integers (modeled here as a function from natural numbers to integers) together with a pointer to a current position (top of the stack):

$|A|_{stack} := \mathbb{N} \times \mathbb{Z}^{\mathbb{N}},$

$empty_A := \langle 0, \lambda i.\ 0\rangle,$
$empty?_A(\langle k, f\rangle) := (k = 0),$
$push_A(n, \langle k, f\rangle) := \langle k+1, \lambda i.\ \textbf{if}\ \ i = k\ \textbf{then}\ \ n\ \textbf{else}\ \ f(i)\rangle,$
$pop_A(\langle k, f\rangle) :=\ \textbf{if}\ \ k = 0\ \textbf{then}\ \ \langle 0, f\rangle\ \textbf{else}\ \ \langle k-1, f\rangle,$
$top_A(\langle k, f\rangle) :=\ \textbf{if}\ \ k = 0\ \textbf{then}\ \ 0\ \textbf{else}\ \ f(k-1).$

Obviously, this algebra does not satisfy the axioms in `STACKS`; the equation

$$pop(push(n,s)) = s \tag{1.1}$$

does not hold, for some values of s and n. On the other hand, intuitively, this seems to be a "reasonable" realization of the datatype of stacks. The intuition behind this is as follows: although the two sides of (1.1) need not be identical, they are indistinguishable w.r.t. the observable sorts $\{bool, int\}$, for any n and s. This leads to the behavioural semantics of specification, according to which models of a specification $SP = \langle \Sigma, Ax\rangle$ are all algebras which *behaviourally (observationally) satisfy* its axioms:

$$BehMod(SP) = \{A \in Alg_\Sigma : A \models_{OBS} Ax\}.$$

Relation of the behavioural satisfaction $\models_{OBS}$ is defined as usual, with the only difference that equality is interpreted in an algebra A as the indistinguishability $\sim_A^{OBS}$ and variables range over the subalgebra generated by $|A|_{OBS}$. Note that $\sim_A^{OBS}$ is a *partial* congruence in general. Formally, $\models_{OBS}$ may be defined by:

$$A \models_{OBS} \phi \iff A/_{\sim_A^{OBS}} \models \phi.$$

Behavioural semantics is closely related to the observational equivalence of algebras. When only equational specifications are considered (i.e., all the axioms in Ax are equations), for standard, partial and regular algebras the class of behavioural models coincides with the closure of standard models under observational equivalence (see [BHW95] for standard algebras and [BT96] for others):

$$BehMod(SP) = \{A \in Alg_{Sig(SP)} : \exists B \in Mod(SP).\ A \equiv_{OBS} B\} \tag{1.2}$$

This result was generalized to observational equivalence in a concrete category in [BT96] (a necessary and sufficient condition for (1.2) is the closedness of the class $Mod(SP)$ under observational quotients).

The most straightforward formalization of the concept of implementation refers to the inclusion of model classes. Namely, we say that specification SP' *implements* (*refines*) SP, denoted by $SP \rightsquigarrow SP'$, if each model of SP' is a model of SP:

$$Mod(SP') \subseteq Mod(SP)$$

(note that both SP and SP' have necessarily the same signatures here, $Sig(SP') = Sig(SP)$). This concept has been refined in two ways [ST88, ST97]. First, a notion of *constructor implementation* was proposed: SP' implements SP via a constructor κ, denoted by $SP \overset{\kappa}{\rightsquigarrow} SP'$, where $\kappa : Alg_{Sig(SP')} \to Alg_{Sig(SP)}$, if

$$\bar{\kappa}(Mod(SP')) \subseteq Mod(SP)$$

($\bar{\kappa}(\_)$ stands here for direct image of κ). Intuitively, function κ, called a *constructor*, represents a parametrized program, realizing one refinement step. Second, a behavioural realization was taken into account: we say that SP' implements behaviourally SP via κ if

$$\bar{\kappa}(BehMod(SP')) \subseteq BehMod(SP). \tag{1.3}$$

As an example, consider a specification of queues and its behavioural implementation by pairs of stacks.

specification `QUEUES` **extends** `INT`, `BOOL` by
regular sorts
  $queue$;
operations
  $empty\_queue : queue$;
  $empty\_queue?(\_) : queue \to bool$;
  $put(\_,\_) : queue \times int \to queue$;
  $get(\_) : queue \to int$;
  $rest(\_) : queue \to queue$;
axioms
  $empty\_queue?(empty\_queue) = true$
  $empty\_queue?(put(q,e)) = false$
  $empty\_queue?(q) \Rightarrow get(put(q,e)) = e \ \wedge \ rest(put(q,e)) = q$
  $\neg empty\_queue?(q) \Rightarrow get(put(q,e)) = get(q) \ \wedge \ rest(put(q,e)) = put(rest(q),e)$

A constructor behavioural implementation of `QUEUES` by `STACKS` may be given using the following constructor (some ad hoc, but hopefully self-explanatory notation is used here to define a function that maps Sig(`STACKS`)-algebras to Sig(`QUEUES`)-algebras):

$queue := stack \times stack$
$empty\_queue := \langle empty, empty \rangle$

$$
\begin{array}{l}
\mathit{empty\_queue?}(\langle s_1, s_2\rangle) := \mathit{empty?}(s_1) \ \wedge\ \mathit{empty?}(s_2) \\
\mathit{put}(\langle s_1, s_2\rangle, e) := \langle \mathit{push}(e, s_1), s_2\rangle \\
\mathit{get}(\langle s_1, s_2\rangle) := \\
\qquad\quad \textbf{if}\ \ \mathit{empty?}(s_2)\ \textbf{then}\ \ \textbf{let}\ \ s_2' := \mathit{reverse}(s_1, s_2)\ \ \textbf{in}\ \ \mathit{top}(s_2') \\
\qquad\quad \textbf{else}\ \ \mathit{top}(s_2) \\
\mathit{rest}(\langle s_1, s_2\rangle) := \\
\qquad\quad \textbf{if}\ \ \mathit{empty?}(s_2)\ \textbf{then}\ \ \textbf{let}\ \ s_2' := \mathit{reverse}(s_1, s_2)\ \ \textbf{in}\ \ \langle \mathit{empty}, \mathit{pop}(s_2')\rangle \\
\qquad\quad \textbf{else}\ \ \langle s_1, \mathit{pop}(s_2)\rangle \\
\\
\mathit{reverse}(\mathit{empty}, s) = s \\
\mathit{reverse}(\mathit{push}(n, s_1), s_2) = \mathit{reverse}(s_1, \mathit{push}(n, s_2))
\end{array}
$$

Note that this is a correct implementation only when behavioural satisfaction is considered, for observable sorts $\{int, bool\}$ both in `QUEUES` and in `STACKS`.

Chapter 5 contains further examples of behavioural implementations, mostly in the framework of regular algebras.

1.4 Bisimilarity from open maps

The concept of an open map is central to our further considerations. Open maps offer a categorical generalization of the notion of bisimulation, hence the abstract approach is illustrated by the strong bisimulation equivalence [Mil80, Mil89], adopted to transition systems with initial state. In this case, open maps are precisely *zig-zag* morphisms; by recalling them we hope to support some intuition about the nature of open maps and make the following definitions more readable.

A labelled transition system T over a labelling set L is a triple $\langle S, i, t\rangle$ consisting of a set S of states, a distinguished initial state $i \in S$ and a family $t = \{ \xrightarrow{a} \}_{a \in L}$ of transition relations, $\xrightarrow{a}\ \subseteq S \times S$, indexed by a set of labels [Plo81]. Two transition systems $T_1 = \langle S_1, i_1, t_1\rangle$ and $T_2 = \langle S_2, i_2, t_2\rangle$ are *strongly bisimilar* if they are related by a *bisimulation* relation, i.e., a binary relation $\sim\ \subseteq S_1 \times S_2$ between states of T_1 and states of T_2 such that

- it relates initial states: $i_1 {\sim} i_2$,
- each transition in T_1 is simulated by an appropriate transition in T_2: whenever $s_1 {\sim} s_2$ and $s_1 \xrightarrow{a} s_1'$ in T_1, there exists some state s_2' in T_2 such that $s_2 \xrightarrow{a} s_2'$ and $s_1' {\sim} s_2'$,
- and vice versa, each transition in T_2 is simulated by an appropriate transition in T_1: whenever $s_1 {\sim} s_2$ and $s_2 \xrightarrow{a} s_2'$ in T_2, there exists some state s_1' in T_1 such that $s_1 \xrightarrow{a} s_1'$ and $s_1' {\sim} s_2'$.

Bisimilarity relation is also called (strong) bisimulation equivalence.

Transition systems over a fixed labelling set L form a category TS_L: a morphisms σ from T_1 to T_2 is a transition-preserving function from the set of states of T_1 to the set of states of T_2 that preserves the initial state as well:

$$
\begin{gathered}
\sigma(i_1) = i_2, \\
\text{if } s \xrightarrow{a} s' \text{ in } T_1 \text{ then } \sigma(s) \xrightarrow{a} \sigma(s') \text{ in } T_2.
\end{gathered}
$$

Intuitively, σ simulates all the transitions of T_1 in T_2. On the other hand, it does not have to reflect transitions, hence it is hopeless to attempt to formulate bisimulation equivalence directly in terms of morphisms in category TS_L. A way out is to distinguish an appropriate subclass of morphisms, namely those reflecting reachable transitions. This idea leads to the concept of a *zig-zag* morphism [C85, BB88]; a morphism σ is zig-zag if it additionally satisfies the following condition, called the zig-zag property in the sequel:

$$\text{if } \sigma(s) \xrightarrow{a} s'' \text{ in } T_2 \text{ then there exists some } s' \text{ s.t. } s \xrightarrow{a} s' \text{ in } T_1 \text{ and } \sigma(s') = s'',$$

for each state s of T_1 *reachable* from the initial state, i.e., accessible by a finite sequence of transitions starting in the initial state i_1:

$$i_1 = s_0 \xrightarrow{a_1} s_2 \xrightarrow{a_2} \ldots \xrightarrow{a_n} s_n = s.$$

Zig-zag morphisms capture bisimilarity in the following sense: T_1 and T_2 are strongly bisimilar iff they are related by a pair of zig-zag morphisms

$$\begin{tikzcd} & T \arrow[dl] \arrow[dr] & \\ T_1 & & T_2 \end{tikzcd}$$

with a common domain T (a *span* of zig-zag morphisms).

In the following, we present a generalization of this concept to an arbitrary category, due to [JNW93]; most importantly, this generalization subsumes strong bisimilarity in TS_L as a special case.

Let $\mathbb{M}$ be a category of models of computation, in which we choose a (not necessarily full) *path subcategory* $\mathbb{P}$. In the sequel, objects from a path subcategory $\mathbb{P}$ will be also called *observation object*, since they play, intuitively, a role of observers of behaviour of other objects. Any morphism $\alpha : P \to A$ in $\mathbb{M}$ from a path object $P \in |\mathbb{P}|$ is understood as an observable *computation* in A following path P. A morphism $h : A \to B$ between models can be intuitively thought of as a simulation of A in B, since h transforms every computation $\alpha : P \to A$ in A to a computation[6] $\alpha; h : P \to B$ in B. Moreover any morphism $m : P \to Q$ in $\mathbb{P}$ corresponds, intuitively, to a *path extension*; whenever $\alpha = m; \alpha'$, the computation $\alpha' : Q \to A$ can be understood as an extension of α (via m).

$$\begin{tikzcd} P \arrow[r, "\alpha"] \arrow[d, "m"'] & A \\ Q \arrow[ur, "\alpha'"'] & \end{tikzcd}$$

Such a commuting triangle we call sometimes a *computation extension*.

Definition 1.4.1 (Open maps and bisimilarity [JNW93]) *A morphism $h : A \to B$ in $\mathbb{M}$ is* $\mathbb{P}$-open *if for any $m : P \to Q$ in $\mathbb{P}$ and two computations $\alpha : P \to A$, $\beta : Q \to B$ in $\mathbb{M}$, whenever the square*

$$\begin{tikzcd} P \arrow[r, "\alpha"] \arrow[d, "m"'] & A \arrow[d, "h"] \\ Q \arrow[r, "\beta"'] & B \end{tikzcd}$$

[6] We write composition in a category in a diagrammatic order.

commutes, i.e., $\alpha; h = m; \beta$*, there exists a* diagonal *morphism* $\gamma : Q \to A$ *in* $\mathbb{M}$ *making two induced triangles commutative,*

$$\begin{tikzcd} P \arrow[r, "\alpha"] \arrow[d, "m"'] & A \arrow[d, "h"] \\ Q \arrow[r, "\beta"'] \arrow[ur, dotted, "\gamma"] & B \end{tikzcd}$$

i.e., $\alpha = m; \gamma$ *and* $\beta = \gamma; h$*. Two objects* A *and* B *are* $\mathbb{P}$-bisimilar*, denoted by* $A \sim_{\mathbb{P}} B$*, if they are related by a span of* $\mathbb{P}$*-open maps in* $\mathbb{M}$*:*

$$\begin{tikzcd} & C \arrow[dl] \arrow[dr] & \\ A & & B \end{tikzcd}$$

We omit the prefix $\mathbb{P}$- when it is obvious from a context. The $\mathbb{P}$-open morphisms form a subcategory of $\mathbb{M}$, as they include all the identity morphisms and are closed under composition. It can be easily checked that all isomorphisms are open. In general, open-map bisimilarity is not necessarily an equivalence relation (transitivity may fail). A sufficient condition for bisimilarity to be an equivalence is given by:

Proposition 1.4.2 ([JNW93]) $\mathbb{P}$*-bisimilarity is an equivalence if* $\mathbb{M}$ *has all pullbacks.*

A proof of this fact, based on the observation that pullbacks of open morphisms are open, can be found in [JNW93].

As already mentioned, the notion of open-map bisimilarity generalizes the strong bisimilarity between labelled transition systems. The latter was shown in [JNW93] to coincide with bisimilarity induced by the full subcategory $Bran_L$ of transition systems consisting of finite sequences of actions:

$$i \xrightarrow{a_1} s_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} s_n. \tag{1.4}$$

Moreover, $Bran_L$-open morphisms are precisely zig-zag morphisms.

In the following we implicitly rely on the observation that the larger an observation subcategory $\mathbb{P}$ is, the fewer morphisms are $\mathbb{P}$-open: if $\mathbb{Q}$ is a subcategory of $\mathbb{P}$ then every $\mathbb{P}$-open morphism is $\mathbb{Q}$-open as well.

Chapter 2

Algebraic observational equivalence by open maps

The topic of this chapter is an application of the setting of open maps to observational equivalences of different classes of algebras.

In Sections 2.1.1 and 2.1.2 we show that the observational equivalence of standard and regular algebras w.r.t. a set OBS of observable sorts can be presented as a bisimilarity induced by the class of observation objects consisting of free objects w.r.t. a forgetful functor $|\_|_{OBS}$. To prove this we formulate an observation (Lemma 2.1.1 in Section 2.1 and Theorem 2.2.3 in Section 2.2) that bisimilarity defined by means of open maps can be transported via *multiadjunction* in both directions. The notion of *multiadjunction*, introduced in [Die79], is a generalization of adjunction where, informally, instead of one free object there exists a whole family of "jointly" free objects. While for categories of standard and regular algebras usual adjunctions are sufficient, this does not extend to partial algebras, where the class of free objects is too small to induce the observational equivalence of interest. In Section 2.2.2 the observational equivalence of partial algebras is proved to be a bisimilarity induced by the union of all families of *multifree* objects of some multiadjunction. This is done in the category of partial algebras with *strong* partial homomorphisms, similarly as in [BT96]. Such a result is not quite satisfactory, as the choice of strong homomorphisms cannot be easily justified. To do the same for partial algebras with all (weak) homomorphisms, we introduce the notion of *pre-adjunction*, being a further relaxation of requirements on multiadjunction. It turns out that every pre-adjunction gives rise to some multiadjunction, when certain necessary conditions are satisfied. We prove that bisimilarity induced by pre-free objects of a pre-adjunction and bisimilarity induced by multifree objects of the related multiadjunction coincide (Theorem 2.4.9 in Section 2.4). This allows us to characterize observational equivalence of partial algebras by means of open maps in the category with weak homomorphisms (in Section 2.4.2). It is worth noting that the observation objects inducing the observational equivalence of partial algebras (i.e., finite partial pre-free algebras) are in analogy with the finite linear transition systems used in [JNW93] to characterize strong bisimilarity, as they both seem to capture correctly a concept of a finite computation (path).

The results concerning open maps in multiadjunction and pre-adjunction are formulated for an arbitrary category, and we expect that they are also applicable in frameworks other than considered here, for example to continuous algebras (we defer a detailed analysis of other cases to further work).

A surprising observation is that all bisimilarities studied in this chapter (i.e., those induced by adjunction, multiadjunction or pre-adjunction) have a common property that each computation in a target of an open map lifts uniquely to a computation in its source. We call such a bisimilarity *deterministic*. In such a case, intuitively, open morphisms do not reduce nondeterminism of computations. We introduce the notion of deterministic bisimilarity in Section 2.2.1. Deterministic open maps correspond to the class of *etale maps* ([JM94]) in a topos. An interesting conclusion is that while open maps appeared useful for behavioural equivalence of concurrent or nondeterministic systems, the same role in models for sequential, deterministic systems is played by etale maps.

In the proofs we use the characterization of observational equivalence of algebras stated in the introductory Section 1.3 (Theorems 1.3.4 and 1.3.5). Standard, regular and partial algebras are treated separately there, as in specific frameworks some slight differences arise. Due to the high degree of generality of our considerations in this chapter, we are also able to spell out this idea in a concrete category [BT96] equipped with a suitable multiadjunction (Section 2.3). This allows us to treat uniformly open-map bisimilarities for different categories of algebras in Theorem 2.3.3.

We are also interested in how large the class of observation objects should be to induce the observational equivalence. For example, in the case of standard and regular algebras it turns out that free algebras over (at most) two-element sets are sufficient, i.e., observations by means of terms over two variables are powerful enough to distinguish non-equivalent algebras. This gives a useful tool for showing equivalence of algebras. The detailed study of observation subcategories for standard, regular and partial algebras are summarized in Corollaries 2.1.7, 2.1.8, 2.2.14 and 2.4.18. Moreover, in the case of deterministic bisimulation, the class of observation objects can be reduced by Lemma 2.2.12 from Section 2.2.1 stating that closure of the subcategory of observation objects under colimits has no influence on the induced class of open maps.

2.1 Bisimilarity in adjunction

In this section we state some general result leading to the conclusion that observational equivalence of standard and regular algebras coincide with the bisimilarity defined by means of open morphisms w.r.t. a natural class of observation objects.

Let us assume that we are given a functor $G : \mathbb{M} \to \mathbb{O}$ from a category $\mathbb{M}$ of models to a category $\mathbb{O}$, having a left adjoint $F : \mathbb{O} \to \mathbb{M}$. The category $\mathbb{O}$ can be intuitively understood here as the category of observation objects, which this time are outside $\mathbb{M}$. Hence, as the path subcategory in $\mathbb{M}$ we choose the image subcategory $F(\mathbb{P})$, for a fixed in this section subcategory $\mathbb{P}$ of $\mathbb{O}$. We will study especially the case when $\mathbb{P} = \mathbb{O}$.

We omit proofs of some facts below, as they are proved in the next section in a more general context of a multiadjunction. For example, the following facts are special cases of Theorem 2.2.3 and Lemma 2.2.4, respectively:

Lemma 2.1.1 *A morphism $h : A \to B$ in $\mathbb{M}$ is $F(\mathbb{P})$-open iff $G(h)$ is $\mathbb{P}$-open in $\mathbb{O}$.*

Corollary 2.1.2 *A morphism $h : A \to B$ in $\mathbb{M}$ is $F(\mathbb{O})$-open iff $G(h)$ is an isomorphism in $\mathbb{O}$.*

Lemma 2.1.1 will be a basic tool in analyzing bisimulations for standard and regular algebras. Essentially, it states that open maps are transported via adjunction in both directions. In Corollary 2.1.2 we deal with the case when $\mathbb{P} = \mathbb{O}$: the $F(\mathbb{O})$-open morphisms in $\mathbb{M}$ are precisely those transported by G to isomorphisms.

A weaker version of Lemma 2.1.1, concerning only coreflections (i.e., adjunctions with unit being a natural isomorphism), was already proved in [JNW93].

For any subcategory $\mathcal{S}$ of $\mathbb{M}$ let $[\mathcal{S}]$ denote the smallest full subcategory of $\mathbb{M}$ containing all the objects of $\mathcal{S}$ (in fact $\mathcal{S}$ and $[\mathcal{S}]$ can differ only on morphisms). The choice of observation objects as an image subcategory $F(\mathbb{O})$ of $\mathbb{M}$ is not the only natural one - we could also take (usually different) full subcategory $[F(\mathbb{O})]$, having the same objects but in general more morphisms. It turns out that both choices give the same result, which we prove in Lemma 2.2.5 in Section 2.2. The following is a specialization of Lemma 2.2.5 for the adjunction:

Corollary 2.1.3 *A morphism $h : A \to B$ in $\mathbb{M}$ is $F(\mathbb{O})$-open iff it is $[F(\mathbb{O})]$-open.*

The straightforward conclusion from Corollary 2.1.2:

Corollary 2.1.4 *Let A, B be objects in $\mathbb{M}$. If A and B are $F(\mathbb{O})$-bisimilar then $G(A)$ and $G(B)$ are isomorphic.*

Proof: There exists in $\mathbb{M}$ a span $A \xleftarrow{f} C \xrightarrow{g} B$, where f and g are open, thus $G(f)$, $G(g)$ are isomorphisms. Then $(G(f)^{-1}; G(g)) : G(A) \to G(B)$ is also an isomorphism. □

The implication in the opposite direction does not hold in general (standard algebras provide a counterexample, see Section 2.1.1). Let us consider a very special situation, studied in the context of open maps in [JNW93], when the adjunction between F and G is a coreflection. We have now a characterization of $F(\mathbb{O})$-bisimilarity:

Proposition 2.1.5 *If the adjunction between F and G is a coreflection then for any objects A, B in $\mathbb{M}$, A and B are $F(\mathbb{O})$-bisimilar iff $G(A)$ and $G(B)$ are isomorphic.*

Proof: The "only if" part follows easily from Corollary 2.1.4, so we only need to prove the "if" implication. Assume that $G(A)$ and $G(B)$ are isomorphic. First we show that for any object X of $\mathbb{M}$, the counit $\varepsilon_X = \mathrm{id}_{G(X)}{}^{\#}$ is $F(\mathbb{O})$-open, i.e., (by Corollary 2.1.2) that $G(\varepsilon_X)$ is iso. Let $\eta : \mathrm{Id}_{\mathbb{O}} \Longrightarrow G \circ F$ denote the unit natural transformation. We have $\eta_{G(X)}; G(\varepsilon_X) = \mathrm{id}_{G(X)}$, hence $G(\varepsilon_X) = \eta_{G(X)}{}^{-1}$ as $\eta_{G(X)}$ is iso. Thus, $G(\varepsilon_X)$ is an isomorphism as well.

Now A and B are bisimilar, as witnessed by a span of open morphisms:

$$\begin{tikzcd}[column sep=small]
& F(G(A)) \arrow[dl, "\varepsilon_A"'] & \cong & F(G(B)) \arrow[dr, "\varepsilon_B"] & \\
A & & & & B
\end{tikzcd}$$

since all isomorphisms are open. □

2.1.1 Standard algebras

We separated the special case of adjunction from the more general situation (with which we deal in Section 2.2) since there are some classical and important examples of such a situation, namely standard and regular algebras. Let Σ denote an arbitrary signature and OBS a subset of observable sorts – both fixed thorough this section.

As an example consider the category Alg_Σ of standard algebras together with the forgetful functor $|\_|_{OBS} : Alg_\Sigma \to Set^{OBS}$ returning the carrier sets of sorts in OBS. The functor $|\_|_{OBS}$ has the left adjoint T_Σ. We conclude from Corollary 2.1.2 that the $T_\Sigma(Set^{OBS})$-open morphisms of standard algebras are precisely the observational morphisms (i.e., homomorphisms bijective on OBS). In this way we obtain the characterization of observational equivalence by means of $T_\Sigma(Set^{OBS})$-open maps:

Corollary 2.1.6 *Observational equivalence of standard algebras coincides with* $T_\Sigma(Set^{OBS})$-*bisimilarity.*

Proof: The proof can be based on Theorem 4.13 from [BT96] (or Theorem 28 in the short version), which implies that standard (regular, partial) algebras are observationally equivalent iff there exists an *observational correspondence* between them. (We recall this result as Theorem 2.3.1 in Section 2.3 together with all the necessary definitions, including the notion of a correspondence.) Moreover, all spans in Alg_Σ are correspondences, hence two algebras A and B are observationally equivalent iff there exists an observational span $A \xleftarrow{f} C \xrightarrow{g} B$ to them, i.e., a span with both f and g observational. But this is precisely another formulation of A and B being $T_\Sigma(Set^{OBS})$-bisimilar, due to the fact that the $T_\Sigma(Set^{OBS})$-open morphisms of standard algebras are precisely the observational morphisms.

A more direct proof can be given, without referring to results from [BT96] (we only sketch it here to illustrate more general ideas contained in Section 2.3). Assume algebras A, B to be observationally equivalent; let v_A and v_B denote two surjective valuations from Theorem 1.3.4, with X being their common domain, $v_A : X \to |A|_{OBS}$, $v_A : X \to |A|_{OBS}$. Observe that by Theorem 1.3.4 we can assume these valuations to be bijective (hence $|A|_{OBS}$ and $|B|_{OBS}$ are related by a bijection). Now, we construct an algebra C as the quotient of the free algebra $T_\Sigma(X)$ by the smallest congruence $\sim$ containing all pairs (t, t') of terms over X of an observable sort such that

$$A[v_A] \models t = t'.$$

Clearly, the same congruence is induced by equalities satisfied in B under valuation v_B. Moreover, the canonical mappings $[t]_\sim \mapsto t_{A[v_A]}$ and $[t]_\sim \mapsto t_{B[v_B]}$ form a pair of observational (hence open) morphisms relating A and B.

For the opposite direction, let A and B be two algebras related by a span of open maps f, g from an algebra C. We use, similarly as before, the free algebra $T_\Sigma(X)$, this time generated by $X = |C|_{OBS}$ and a canonical extension of identity on X, $\mathrm{id}_X = \varepsilon_C{}^\# : T_\Sigma(X) \to C$, the counit of the adjunction.

$$\begin{tikzcd}
 & T_\Sigma(X) \arrow[d, "\mathrm{id}_X{}^\#"] & \\
 & C \arrow[dl, "f"'] \arrow[dr, "g"] & \\
A & & B
\end{tikzcd}$$

Since f and g are observational, valuations v_A and v_B from Theorem 1.3.4 can be given by $|f|_{OBS} : X \to |A|_{OBS}$ and $|g|_{OBS} : X \to |B|_{OBS}$, respectively, as both $|\mathrm{id}_X{}^\#; f|_{OBS}$ and $|\mathrm{id}_X{}^\#; g|_{OBS}$ identify the same pairs of terms over X. □

In Section 2.3 we propose a generalization of this proof in a concrete category, subsuming also the case of regular algebras as well as partial algebras with strong homomorphisms.

Interestingly, a subcategory of observation objects can be reduced while still inducing the same bisimilarity. Any subcategory $T_\Sigma(\mathbb{P})$ of $T_\Sigma(Set^{OBS})$ is a candidate when $\mathbb{P}$ is chosen large enough to "distinguish" between non-isomorphic objects in Set^{OBS}, i.e., to make them non-$\mathbb{P}$-bisimilar. Surprisingly, a full subcategory $\mathbf{2}_{OBS}$ is sufficient, consisting of only those objects X in Set^{OBS} for which X_s is empty or of cardinality two, for each $s \in OBS$. We obtain the following fact as a corollary from Lemma 2.1.1 and Corrolaries 2.1.2 and 2.1.3:

Corollary 2.1.7 *The observational equivalence of standard algebras coincides with $\mathbb{Q}$-bisimilarity, for any subcategory $\mathbb{Q}$ of $[T_\Sigma(Set^{OBS})]$ containing $T_\Sigma(\mathbf{2}_{OBS})$.*

Proof: We only have to show (by Lemma 2.1.1) that $\mathbf{2}_{OBS}$-bisimilarity in Set^{OBS} coincides with Set^{OBS}-bisimilarity, i.e., that two OBS-sorted sets X, Y are $\mathbf{2}_{OBS}$-bisimilar iff they are isomorphic. This coincidence (only one implication is non-trivial) can be easily verified, similarly as in the proof of Lemma 2.4.17 below. □

Corollary 2.1.7 can be interpreted as follows: terms of observable sorts with only two variables are powerful enough to distinguish algebras that are not observationally equivalent. Observational equivalence of standard algebras is thus very "robust" to changes of subcategory of observation objects. More generally, this is always the case when a bisimilarity is *deterministic* (the concept to be introduced in Section 2.2.1).

Note Surprisingly, when there are no Σ-algebras with empty carrier of sort $s \in OBS$ in our category, we can reduce $\mathbf{2}_{OBS}$ by omitting sets empty on s. In particular, when signature Σ excludes empty carrier set on any observable sort, $\mathbf{2}_{OBS}$ consists of precisely one object (up to isomorphism). Consequently, the path subcategory for observational equivalence can contain only one object.

2.1.2 Regular algebras

The forgetful functor $|\_|_{OBS} : RAlg_\Sigma \to Set^{OBS}$ has the left adjoint $R^\mu_\Sigma(\_)$; a free Σ-regular algebra over X is an algebra of so called regular partial $\Sigma(X)$-trees [GT74, Tiu78], where $\Sigma(X)$ extends Σ with the (many-sorted) set X of constants.

Given an algebraic signature Σ, by a Σ-*tree* we mean a tree whose nodes are labelled by operation symbols, such that the number of sons of a node is equal to the arity of the labelling symbol. Sons of a node are ordered, according to the order of arguments of the labelling operation. We require a *finite* path from the root to any node, however we allow for *infinite* paths in a tree. In a *partial* Σ-*tree* we allow the number of sons of a node to be less that arity of a labelling symbol; the non-existing sons correspond to $\perp$. A partial tree is regular if it has only finitely many different subtrees; obviously, each finite partial tree is regular. Notice that the position of each node in such a tree can be encoded by a finite sequence of naturals, each consecutive number n determining the next node as nth son on the previous node; position of root is encoded by the empty sequence. Let $\mathrm{dom}(t)$ denote the set of codes of positions of all nodes in a tree t.

Now, the carrier of $R^\mu_\Sigma(X)$ is a many-sorted set of all regular partial Σ-trees (a sort of a tree is equal to a sort of an operation symbol labelling its root node); operations are given by adjoining a new root to arguments, becoming subtrees; and the partial order is defined by

$$t_1 \leq t_2 \text{ iff } \mathrm{dom}(t_1) \subseteq \mathrm{dom}(t_2) \text{ and } t_2 \text{ has the same labels as } t_1 \text{ on } \mathrm{dom}(t_1).$$

The free regular algebra can be defined alternatively using terminology from Section 1.3. We start from the standard algebra with carrier consisting of all regular Σ-terms with free variables from X and with operations defined syntactically, in a canonical way. Now, as the first step we quotient its carrier by the smallest congruence induced by the following equalities, expressing α-conversion:

$$\mu x.t(x) = \mu y.t(y), \text{ for all variables } x, y.$$

In this way we obtain a pre-regular algebra (a notion introduced in Section 5.6); the quotient of it by the smallest pre-congruence yields the regular algebra $R^{\mu}_{\Sigma}(X)$ (cf. Section 5.6 for details of this construction). Roughly speaking, this quotient identifies all "fix-point" equations which necessarily hold in every regular algebras.

Due to the described adjunction, similarly as for standard algebras, we obtain the following:

Corollary 2.1.8 *The observational equivalence of regular algebras coincides with $\mathbb{Q}$-bisimilarity, for any subcategory $\mathbb{Q}$ of $[R^{\mu}_{\Sigma}(Set^{OBS})]$ containing $R^{\mu}_{\Sigma}(\mathbf{2}_{OBS})$.*

We omit the proof, since the argument follows immediately by Theorems 1.3.4 and 2.3.3. Moreover, another proof can be given by means of observational correspondences, similarly as in the case of standard algebras.

2.1.3 Partial algebras

Unfortunately, the above results of this section cannot be applied to partial algebras – this is the main motivation for the similar development in the framework of a multiadjunction in the next section. We will concentrate on two usually studied categories of partial algebras, different on morphisms: $PAlg^{\mathrm{w}}_{\Sigma}$ with all (weak) partial homomorphisms and its subcategory $PAlg_{\Sigma}$, containing only strong morphisms.

Category $PAlg^{\mathrm{w}}_{\Sigma}$ has free algebras with respect to the forgetful functor $|\_|_{OBS}$. The carrier set of a free partial algebra over $X \in |Set^{OBS}|$ is equal to X (i.e., empty on sorts $s \notin OBS$); all its operations are undefined (on all arguments). The free partial algebras have, intuitively, not enough structure to distinguish each pair of observationally non-equivalent partial algebras. To see this, notice that by Corollary 2.1.2, open morphisms of partial algebras for such a path subcategory are precisely observational homomorphisms. Hence a span of open maps relating a pair of partial algebras guarantees only bijective correspondence between carrier sets on OBS, but does not imply any relationship between domains of definedness of operations in both algebras.

A way out would be to switch to the category with strong homomorphisms, whose morphisms automatically reflect definedness of operations. But now there are no free algebras! However, there exist families of multifree algebras, with a universal property analogous to the free objects – the only difference is that instead of a single free object, there exists a set of "jointly" free objects. Multifree families give raise to the notion of left multiadjoint and multiadjunction, which is a topic of Section 2.2. We turn back to partial algebras in Section 2.2.2, where we instantiate and illustrate the general considerations from the next section.

2.2 Bisimilarity in multiadjunction

In Section 2.1 we considered bisimilarity when an adjunction is given between categories of models and observation objects. In this section we assume the existence of *left multiadjoint* [Die79] to $G : \mathbb{M} \to \mathbb{O}$, where instead of one free object we have a family of free objects. Let $G : \mathbb{M} \to \mathbb{O}$ be a functor, fixed throughout this section. The definition below is adapted from [Die79]:

Definition 2.2.1 (Left multiadjoint, multiadjunction) *Let X be any object of category $\mathbb{O}$. Let $F(X) = \{F_i(X)\}_{i \in \mathbb{I}_X}$ and $\eta_X = \{\eta^i_X : X \to G(F_i(X))\}_{i \in \mathbb{I}_X}$ be families of objects of $\mathbb{M}$ and morphisms of $\mathbb{O}$, respectively, both indexed by the same set $\mathbb{I}_X$. We say that a pair $\langle F(X), \eta_X \rangle$ is* a family of multifree objects over X *if for any object A in $\mathbb{M}$ and any morphism $f : X \to G(A)$ there exists a unique pair $\langle i, f^\# : F_i(X) \to A \rangle$ where $i \in \mathbb{I}_X$ and $f^\#$ is a morphism in $\mathbb{M}$, such that $\eta^i_X ; G(f^\#) = f$.*

If for every object X in $\mathbb{O}$ there exists a family $\langle F(X), \eta_X \rangle$ of multifree objects over X, we say that the pair of families $\langle F = \{F(X)\}_{X \in |\mathbb{O}|}, \eta = \{\eta_X\}_{X \in |\mathbb{O}|} \rangle$ is the left multiadjoint *to G and that $\langle F, \eta \rangle$ and G form a* multiadjunction.

Assume in this section that G has a left multiadjoint $\langle F, \eta \rangle$. In [Die79] it was proved that the left multiadjoint is determined uniquely up to isomorphism. The main difference between multiadjunction and adjunction is that we have now a bijection

$$(\_)^{\#_{X,A}} : \mathbb{O}(X, G(A)) \to \biguplus_{i \in \mathbb{I}_X} \mathbb{M}(F_i(X), A), \tag{2.1}$$

for any X in $\mathbb{O}$ and A in $\mathbb{M}$, instead of a bijection $\mathbb{O}(X, G(A)) \to \mathbb{M}(F(X), A)$. As usual we suppress the index of #, hoping that this should not lead to an ambiguity. It is easy to observe that when all index sets $\mathbb{I}_X$ are singletons we obtain a usual adjunction.

There is no unique way to define "$F(f)$" for a morphism $f : X \to Y$ in $\mathbb{O}$. Instead we can naturally define a family of morphisms $\{F_j(f)\}_{j \in \mathbb{I}_Y}$, where for $j \in \mathbb{I}_Y$,

$$F_j(f) := (f; \eta^j_Y)^\# : F_i(X) \to F_j(Y) \tag{2.2}$$

(for a necessarily unique $i \in \mathbb{I}_X$). In this way we obtain a function $\mathbb{I}_f : \mathbb{I}_Y \to \mathbb{I}_X$ (actually we obtain a functor $\mathbb{I}_\_ : \mathbb{O}^{op} \to Set$) such that $\mathbb{I}_f(j) = i$ iff the domain of $F_j(f)$ is $F_i(X)$. Moreover the following "naturality" property of η holds:

$$\eta^{\mathbb{I}_f(j)}_X ; G(F_j(f)) = f; \eta^j_Y \tag{2.3}$$

for any morphism $f : X \to Y$ in $\mathbb{O}$ and $j \in \mathbb{I}_Y$. The following lemma will be useful in the sequel. It indicates that the notion of multiadjunction is quite close to adjunction:

Lemma 2.2.2 *$f^\#; g = (f; G(g))^\#$, for any $f : X \to G(Y)$ in $\mathbb{O}$ and $g : Y \to Z$ in $\mathbb{M}$.*

Proof: Let $f^\# : F_i(X) \to Y$, i.e., $F_i(X)$ is the domain of $f^\#$. It is enough to show that $\eta^i_X ; G(f^\#; g) = f; G(g)$. We have

$$\eta^i_X ; G(f^\#; g) = \eta^i_X ; G(f^\#); G(g) = f; G(g),$$

which yields the result. □

From this lemma it follows that morphisms $f^\#$ and $(f;G(g))^\#$ have the same domain; moreover, using this fact one can easily show that a left multiadjoint is functorial in a sense that it preserves composition:

$$F_{\mathbb{I}_g(k)}(f);F_k(g) = F_k(f;g),$$

where $X \xrightarrow{f} Y \xrightarrow{g} Z$ are morphisms in $\mathbb{O}$ and $k \in \mathbb{I}_Z$.

Let $\mathbb{P}$ be a fixed subcategory of $\mathbb{O}$. As a subcategory of observation objects in $\mathbb{M}$ we take now the subcategory $F(\mathbb{P})$ defined as follows: objects are $\{F_i(X)\}_{X\in|\mathbb{P}|,i\in\mathbb{I}_X}$, morphisms are $\{F_j(f)\}_{f:X\to Y\in|\mathbb{P}|,j\in\mathbb{I}_Y}$. This obviously defines a subcategory, as the morphisms are closed under composition and contain all identities. Theorem 2.2.3 states that openness is transported via multiadjunction in both directions.

Theorem 2.2.3 *A morphism $h : A \to B$ in $\mathbb{M}$ is $F(\mathbb{P})$-open iff $G(h)$ is $\mathbb{P}$-open in $\mathbb{O}$.*

Proof: $\Leftarrow$: Let $h : A \to B$ be any morphism in $\mathbb{M}$ and assume $G(h)$ is $\mathbb{P}$-open. Consider a commuting square:

$$\begin{tikzcd} F_i(X) \arrow[r, "p^\#"] \arrow[d, "F_j(m)"'] & A \arrow[d, "h"] \\ F_j(Y) \arrow[r, "q^\#"'] & B \end{tikzcd} \tag{2.4}$$

where X, Y are objects of $\mathbb{O}$, $m : X \to Y$ is a morphism of $\mathbb{O}$. We will find a morphism $r : Y \to |A|$ such that $r^\# : F_j(Y) \to A$ makes both induced triangles commute. First observe that the following square commutes in $\mathbb{O}$:

$$\begin{tikzcd} X \arrow[r, "p"] \arrow[d, "m"'] & G(A) \arrow[d, "G(h)"] \\ Y \arrow[r, "q"'] & G(B) \end{tikzcd} \tag{2.5}$$

Indeed: since diagram (2.4) commutes, i.e., $p^\#;h = F_j(m);q^\#$, thus

$$\eta^i_X;G(p^\#;h) = \eta^i_X;G((m;\eta^j_Y)^\#;q^\#),$$

from which we conclude $p;G(h) = m;\eta^j_Y;G(q^\#)$ and then $p;G(h) = m;q$, i.e., diagram (2.5) commutes. $G(h)$ is $\mathbb{P}$-open so there exists a diagonal morphism $r : Y \to G(A)$ such that $r;G(h) = q$ and $m;r = p$:

$$\begin{tikzcd} X \arrow[r, "p"] \arrow[d, "m"'] & G(A) \arrow[d, "G(h)"] \\ Y \arrow[r, "q"'] \arrow[ur, dotted, "r"] & G(B) \end{tikzcd} \tag{2.6}$$

We will show that $r^\#$ is a diagonal morphism for diagram (2.4), i.e., that both triangles commute in:

$$\begin{tikzcd} F_i(X) \arrow[r, "p^\#"] \arrow[d, "F_j(m)"'] & A \arrow[d, "h"] \\ F_j(Y) \arrow[r, "q^\#"'] \arrow[ur, dotted, "r^\#"] & B \end{tikzcd} \tag{2.7}$$

First note that the domain of $r^\#$ is the same as the domain of $q^\#$, since $r^\#; h = (r; G(h))^\# = q^\#$ by Lemma 2.2.2 (this proves also that the bottom-right triangle in diagram (2.7) commutes). Now, we only need to show that the upper-left triangle commutes too. By (2.3) we have

$$\eta^i_X; G(F_j(m); r^\#) = m; \eta^j_Y; G(r^\#) = m; r = p = \eta^i_X; G(p^\#),$$

which implies $F_j(m); r^\# = p^\#$.

$\Rightarrow$: The idea of the proof is analogous to the "if" part. Let $h : A \to B$ be any $F(\mathbb{P})$-open morphism in $\mathbb{M}$. Consider a commuting square (2.5); now we are looking for a diagonal morphism $r : Y \to G(A)$. First, we lift the square (2.5) to the commutative square (2.4): let $F_i(X)$ denote the domain of $p^\#$, $F_j(Y)$ – the domain of $q^\#$ and $F_k(X)$ – the domain of $F_j(m)$. Again by Lemma 2.2.2 we have $\eta^i_X; G(p^\#; h) = p; G(h)$ and $\eta^k_X; G(F_j(m); q^\#) = m; q$. Since the square in (2.5) commutes, we get

$$p^\#; h = (p; G(h))^\# = (m; q)^\# = F_j(m); q^\#,$$

$k = i$ and the square (2.4) commutes. Now, since h is $F(\mathbb{P})$-open, there exists a diagonal morphism $r^\# : F_j(Y) \to A$ in diagram (2.7) being necessarily an image under bijection $(\_)^\#$ of some morphism $r : Y \to G(A)$. It is only to be shown that r makes both triangles in diagram (2.6) commute. But this is immediate since diagram (2.4) commutes:

$$r; G(h) = \eta^j_Y; G(r^\#); G(h) = \eta^j_Y; G(q^\#) = q,$$
$$m; r = m; \eta^j_Y; G(r^\#) = \eta^i_X; G(F_j(m); r^\#) = \eta^i_X; G(p^\#) = p.$$

□

For $\mathbb{P}$ being the whole category $\mathbb{O}$ we obtain the following corollary from Theorem 2.2.3:

Lemma 2.2.4 *A morphism $h : A \to B$ in $\mathbb{M}$ is $F(\mathbb{O})$-open iff $G(h)$ is an isomorphism in $\mathbb{O}$.*

Proof: We will only prove that f is an isomorphism in $\mathbb{O}$ iff f is $\mathbb{O}$-open. Then the result follows easily by Theorem 2.2.3. Assumed f is $\mathbb{O}$-open, we construct f^{-1} as a diagonal morphism in the following diagram:

$$\begin{tikzcd} A \arrow[r, "\mathrm{id}_A"] \arrow[d, "f"'] & A \arrow[d, "f"] \\ B \arrow[r, "\mathrm{id}_B"'] \arrow[ur, dotted, "f^{-1}"] & B \end{tikzcd}$$

By $\mathbb{O}$-openness of f such diagonal morphism exists and $f; f^{-1} = \mathrm{id}_A$ and $f^{-1}; f = \mathrm{id}_B$. □

Lemma 2.2.5 *A morphism $h : A \to B$ in $\mathbb{M}$ is $F(\mathbb{O})$-open iff it is $[F(\mathbb{O})]$-open.*

Proof: The "if" part is obvious, as $[F(\mathbb{O})]$-openness is stronger. Instead of proving directly the "only if" implication, we will show (based on Lemma 2.2.4) that h is $[F(\mathbb{O})]$-open whenever $G(h)$ is iso in $\mathbb{O}$. Consider any commuting (outer) square in the following diagram:

$$\begin{tikzcd} F_i(X) \arrow[r, "p^\#"] \arrow[d, "m^\#"'] & A \arrow[d, "h"] \\ F_i(Y) \arrow[r, "q^\#"'] \arrow[ur, dotted, "r^\#"] & B \end{tikzcd}$$

where X, Y are objects of $\mathbb{O}$, $m : X \to G(F_j(Y))$ is a morphism of $\mathbb{O}$. We put $r := q; G(h)^{-1}$ and show that $r^\#$ is a diagonal morphism, namely

i) $r^\#; h = q^\#$,

ii) $m^\#; r^\# = p^\#$.

i) From the definition of r we have $r; G(h) = q$. Now, by Lemma 2.2.2, we have $r^\#; h = (r; G(h))^\# = q^\#$. Note that this also guarantees that domains of $r^\#$ and $q^\#$ are the same.

ii) The square in the diagram above commutes, i.e., $m^\#; q^\# = p^\#; h$. Hence by i) we have $m^\#; r^\#; h = p^\#; h$. As G is a functor, then $G(m^\#; r^\#); G(h) = G(p^\#); G(h)$, and because $G(h)$ is iso (hence mono), we conclude: $G(m^\#; r^\#) = G(p^\#)$. Now $\eta^i_X; G(m^\#; r^\#) = \eta^i_X; G(p^\#) = p$, which implies $m^\#; r^\# = p^\#$. □

In Lemma 2.2.11 in Section 2.2.1, equipped with a notion of determinism, we prove a result similar to Lemma 2.2.5 for a wide class of subcategories $\mathbb{P}$ of $\mathbb{O}$.

2.2.1 Determinism

Consider a commuting diagram (2.4) in $\mathbb{M}$ for any $F(\mathbb{O})$-open morphism h and any morphism $m : X \to Y$ in $\mathbb{O}$. It is interesting to note that the diagonal morphism $r^\#$ making both triangles commute is unique. Moreover, this morphism is determined by only h and $q^\#$, i.e., replacing $p^\#$ and/or $F_j(m)$ by any other morphisms does not influence $r^\#$:

Observation 2.2.6 *For any $F(\mathbb{O})$-open morphism $h : A \to B$ in $\mathbb{M}$ and $q : Y \to G(B)$ in $\mathbb{O}$ there exists at most one morphism $r : Y \to G(A)$ in $\mathbb{O}$ such that $r^\#; h = q^\#$.*

This means that two different computations in A cannot be simulated by the same computation in B. Intuitively speaking, open morphisms cannot reduce nondeterminism of computations.

The diagonal morphisms need not to be uniquely determined in general; it is not so for example in the category of transition systems with path subcategory $Bran_L$. The observation above leads to the general notion of determinism of open morphisms.

Definition 2.2.7 *Consider any subcategory $\mathbb{Q}$ of a category $\mathbb{M}$ of models. We say that $\mathbb{Q}$-bisimilarity in $\mathbb{M}$ is* deterministic *if for any $\mathbb{Q}$-open morphism h there is exactly one diagonal morphism $\gamma : Q \to A$ in every commuting square*

$$\begin{tikzcd} P \arrow[r, "\alpha"] \arrow[d, "m"'] & A \arrow[d, "h"] \\ Q \arrow[r, "\beta"'] & B \end{tikzcd} \tag{2.8}$$

where $m : P \to Q$ is in $\mathbb{Q}$. Moreover $\mathbb{Q}$-bisimilarity will be called strongly deterministic *if for any $\mathbb{Q}$-open morphism $h : A \to B$ in $\mathbb{M}$ and any computation $\beta : O \to B$ in $\mathbb{M}$ there exists at most one computation $\gamma : O \to A$ in $\mathbb{M}$ making $\gamma; h = \beta$.*

Note We concentrate on the stronger notion, as bisimilarities in all categories of algebras we consider are strongly deterministic. Even more: both properties of determinism coincide under an assumption that $\mathbb{M}$ and $\mathbb{P}$ have a common initial object, i.e., $\mathbb{M}$ has the initial object $\mathcal{I}$ and $\mathcal{I}$ belongs to $\mathbb{Q}$ together with all morphisms from $\mathcal{I}$ to objects of $\mathbb{Q}$. (The unique morphism from $\mathcal{I}$ to A can be thought of as the initial "empty computation" in A.) They still coincide, when we only assume existence of a multi-initial family of objects that is a family $\{\mathcal{I}_i\}_{i\in\mathbb{I}}$ of objects indexed by some set $\mathbb{I}$, with the property that for each object A in $\mathbb{M}$, there is exactly one pair $\langle i \in \mathbb{I}, f : \mathcal{I}_i \to A\rangle$.

Proposition 2.2.8 *A deterministic $\mathbb{Q}$-bisimilarity is strongly deterministic, whenever $\mathbb{M}$ has a multi-initial family of objects $\{\mathcal{I}_i\}_{i\in\mathbb{I}}$ such that all objects $\mathcal{I}_i$, together with all morphisms from $\mathcal{I}_i$ to objects of $\mathbb{Q}$, are in $\mathbb{Q}$, for all $i \in \mathbb{I}$.*

Proof: By contradiction: assume that $\mathbb{Q}$-bisimilarity does not satisfy the condition of strong determinism, i.e., there exists an open morphism $h : A \to B$ and two different computations $p, q : Q \to A$ with $p; h = q; h$. We construct a commuting square as follows:

$$\begin{tikzcd} \mathcal{I} \arrow[r, "\iota_A"] \arrow[d, "\iota_O"'] & A \arrow[d, "h"] \\ Q \arrow[ur, dotted, "p"] \arrow[ur, dotted, "q"'] \arrow[r, "p;h=q;h"'] & B \end{tikzcd} \tag{2.9}$$

where ι_A denotes the unique morphism from one of multi-initial objects $\mathcal{I}$ to A. Observe that ι_A and ι_O are forced to have the same domains, since otherwise we would get two different candidates for ι_B, which is clearly forbidden by the universal property of the multi-initial family. Then, both p and q are diagonal morphisms, i.e., make two triangles commute, since by postcomposing with ι_O they necessarily give ι_A. But this contradicts the assumption about determinism of $\mathbb{Q}$-bisimilarity, hence concludes the proof. □

In Observation 2.2.6 we noticed that $F(\mathbb{O})$-bisimilarity in $\mathbb{M}$ is strongly deterministic for any left multiadjoint F. In Lemma 2.2.9 we deal with an arbitrary subcategory $\mathbb{P}$ of observation objects:

Lemma 2.2.9 *Assume $\mathbb{P}$ is a subcategory of $G(\mathbb{M})$. If $\mathbb{P}$-bisimilarity in $G(\mathbb{M})$ is strongly deterministic then $F(\mathbb{P})$-bisimilarity in $\mathbb{M}$ is so as well.*

Proof: Having the bijection (2.1) it is enough to prove that

$$r; G(h) = s; G(h) \quad \text{if} \quad r^{\#}; h = s^{\#}; h,$$

for any computations r and s in $\mathbb{O}$ with a domain object from $\mathbb{P}$ and any $F(\mathbb{P})$-open morphism h in $\mathbb{M}$. Now, by Lemma 2.2.4, $r^{\#}; h = (r; G(h))^{\#}$ and similarly $s^{\#}; h = (s; G(h))^{\#}$, from which the implication above follows easily. □

The following definition will be useful in this section as well as in the next chapter:

Definition 2.2.10 *We say that $\mathbb{M}$ is* initialized *(w.r.t. $\mathbb{Q}$) if for any $\mathbb{Q}$-open morphism $h : A \to B$, any object Q from $\mathbb{Q}$ and a computation $\beta : Q \to B$ there exists an object P in $\mathbb{Q}$ together with computation $\alpha : P \to A$ and a morphism $m : P \to Q$ in $\mathbb{Q}$ such that diagram (2.8) commutes.*

For example, $\mathbb{M}$ is initialized w.r.t. $\mathbb{Q}$ whenever it satisfies the assumption from Proposition 2.2.8.

In the deterministic case, some closures of the observation subcategory do not change the induced bisimilarity. First "closure" property, stated in Lemma 2.2.11, is a strengthening of Lemma 2.2.5.

Lemma 2.2.11 *Assume that $\mathbb{M}$ is initialized w.r.t. $\mathbb{Q}$ and that $\mathbb{Q}$-bisimilarity in $\mathbb{M}$ is strongly deterministic. Then a morphism in $\mathbb{M}$ is $\mathbb{Q}$-open iff it is $[\mathbb{Q}]$-open and moreover $[\mathbb{Q}]$-bisimilarity is also strongly deterministic.*

Proof: Let h be any $\mathbb{Q}$-open morphism in $\mathbb{M}$. We will show h to be $[\mathbb{Q}]$-open, i.e., we will construct a diagonal morphism r in any commuting square:

$$\begin{tikzcd} X \arrow[r, "p"] \arrow[d, "m"'] & A \arrow[d, "h"] \\ Y \arrow[r, "q"] & B \end{tikzcd}$$

where $m : X \to Y$ is in $[\mathbb{Q}]$. Category $\mathbb{M}$ is initialized and h is $\mathbb{Q}$-open, hence there exists a morphism $r : Y \to A$ satisfying $r; h = q$. It remains only to prove $m; r = p$. The diagram above commutes, i.e., $m; q = p; h$. From $r; h = q$ we conclude $m; r; h = p; h$, and from $\mathbb{Q}$-bisimilarity being strong deterministic we obtain $m; r = p$, which completes the proof. $\Box$

Deterministic bisimilarity has another interesting property, namely that the closure of observation subcategory under colimits does not change the bisimilarity induced. In Lemma 2.2.12 we show that $\mathbb{Q}$-bisimilarity coincides with $\mathbb{Q}^{\to}$-bisimilarity, where subcategory $\mathbb{Q}^{\to}$ of $\mathbb{M}$ is defined as follows. It contains as objects colimits of all diagrams in $\mathbb{Q}$. Let P and Q now be colimits of two diagrams D_P and D_Q over the same graph (i.e., D_P and D_Q are functors into $\mathbb{Q}$ with a common domain category G). Let $\{p_i : P_i \to P\}_{i\in I}$ and $\{q_i : Q_i \to Q\}_{i\in I}$ denote colimit cocones on D_P and D_Q, respectively, where I denotes the set of objects of G. Every natural transformation $\xi : D_P \Longrightarrow D_Q$ induces a cocone $\{(\xi_i; q_i) : P_i \to Q\}_{i\in I}$ and results in the unique morphism $!_\xi : P \to Q$ such that $p_i; !_\xi = \xi_i; q_i$, for $i \in I$. Let category $\mathbb{Q}^{\to}$ be the smallest subcategory of $\mathbb{M}$ containing all morphisms induced in this way by any two diagrams (over the same graph, i.e., corresponding to functors from the same category) and any natural transformation (between these functors). In other words, morphisms of $\mathbb{Q}^{\to}$ are all compositions $!_{\xi^1}; \ldots ; !_{\xi^k}$ of finite number of morphisms that can be obtained in the way described above.

Lemma 2.2.12 *For any subcategory $\mathbb{Q}$ of $\mathbb{M}$ such that $\mathbb{Q}$-bisimilarity is deterministic, a morphism of $\mathbb{M}$ is $\mathbb{Q}$-open iff it is $\mathbb{Q}^{\to}$-open. Moreover, if $\mathbb{Q}$-bisimilarity is (strongly) deterministic, then $\mathbb{Q}^{\to}$-bisimilarity is also so.*

Proof: It is enough to prove that every $\mathbb{Q}$-open morphism is also $\mathbb{Q}^{\to}$-open, as $\mathbb{Q}$ is a subcategory of $\mathbb{Q}^{\to}$. Consider any commuting (outer) square:

$$\begin{tikzcd} P_0 \arrow[r, "\iota"] \arrow[d, "f"'] & A \arrow[d, "h"] \\ P_k \arrow[r, "\kappa"'] \arrow[ur, dotted, "r"] & B \end{tikzcd} \tag{2.10}$$

such that h is $\mathbb{Q}$-open and $f = !_{\xi^1}; \ldots ; !_{\xi^k}$, where $!_{\xi^i} : P_{i-1} \to P_i$ is the unique morphism induced by a natural transformation ξ^i and $P_0, \ldots, P_k$ are colimits of some diagrams in $\mathbb{Q}$. We are going to construct a diagonal morphism r in diagram above.

It is enough to construct such a diagonal r for $k = 1$, i.e., for $f = !_\xi : P_0 \to P_1$ — the general case follows then by an easy induction on k. Let D and D' denote functors from the same category G such that P_0 and P_1 are colimits of D and D', respectively, and let $\xi : D \Longrightarrow D'$. Let I denote the class of objects of G and let $\{p_i : D_i \to P_0\}_{i \in I}$ and $\{p'_i : D'_i \to P_1\}_{i \in I}$ be the colimit cocones. We define a family of morphisms $\{r_i : D'_i \to A\}_{i \in I}$ as diagonal morphisms in squares $\xi_i; (p'_i; \kappa) = (p_i; \iota); h$. As $\mathbb{Q}$-bisimilarity is deterministic, $\{r_i : D'_i \to A\}_{i \in I}$ is a cocone over D'. To show this, consider a morphism $m : D_l \to D_n$ in D and the corresponding $m : D'_l \to D'_n$ in D', for some $l, n \in I$.

$$\begin{tikzcd}
D_l \arrow[r, "m"'] \arrow[rr, bend left, "p_l"] \arrow[d, "\xi_l"] & D_n \arrow[r, "p_n"'] \arrow[d, "\xi_n"] & P_0 \arrow[r, dotted, "\iota"] \arrow[d, dotted, "!_\xi"] & A \arrow[d, dotted, "h"] \\
D'_l \arrow[r, "m'"] \arrow[rr, bend right, "p'_l"'] \arrow[rrru, "r_l"] & D'_n \arrow[r, "p'_n"] \arrow[rru, "r_n"] & P_1 \arrow[r, dotted, "\kappa"] & B
\end{tikzcd} \tag{2.11}$$

We should show $r_l = m'; r_n$; to this aim we check that $m'; r_n$ (as well as r_l) is a diagonal for the outermost square in diagram above; then we apply determinism of $\mathbb{Q}$-bisimilarity. First observe that $r_l; h = (m'; r_n); h$, as $r_l; h = p'_l; \kappa = m'; p'_n; \kappa = m'; r_n; h$, since $r_n; h = p'_n; \kappa$. Second, we have $\xi_l; (m'; r_n) = \xi_l; r_l$, since by naturality of ξ $\xi_l; m'; r_n = m; \xi_n; r_n = m; p_n; \iota = p_l; \iota = \xi_l; r_l$. Now, as $\mathbb{Q}$-bisimilarity is deterministic, all this implies $r_l = m'; r_n$, hence $\{r_i : D'_i \to A\}_{i \in I}$ is a cocone over D'. This gives us a unique morphism $r : P_1 \to A$, making $p'_i; r = r_i$, for all $i \in I$. It is easy now to verify that r is a diagonal morphism in diagram (2.10), by the universal property of colimits. Equality $r; h = \kappa$ follows from the fact that morphisms of a colimit cocone are "jointly epi": we have $p'_i; (r; h) = p'_i; \kappa$, for all $i \in I$ (since $p'_i; r = r_i$), hence $r; h = \kappa$. Similarly, to show that $!_\xi; r = \iota$, observe that $p_i; (!_\xi; r) = p_i; \iota$, for all $i \in I$, since morphisms $r_i = p'_i; r$ are diagonal and squares $\xi_i; p'_i = p_i; !_\xi$ commute by the definition of $!_\xi$.

Finally, (strong) determinism of $\mathbb{Q}^{\to}$-bisimilarity follows easily from $\mathbb{Q}$-bisimilarity being (strongly) deterministic and from the fact that morphisms of a colimit cocone are jointly epi. □

A notion of deterministic bisimilarity is especially interesting since open maps in this case correspond to so called *etale* maps [JM94]. This can be easily seen when studying embedding of a category $\mathbb{M}$ into the category of presheaves over the path subcategory $\mathbb{P}$. We will come back to deterministic bisimilarity in Section 4.5.2, where we derive some further properties to hold in this case.

2.2.2 Partial algebras

We are going to demonstrate that observational equivalence in the category of partial algebras can be defined as a bisimulation in the setting of open morphisms. In fact we will characterize the observational equivalence twice: first (in this section) in the category $PAlg_\Sigma$

of partial algebras with strong homomorphisms and then (in Section 2.4.2) in the richer category $PAlg^{\mathrm{w}}_{\Sigma}$ of weak partial homomorphisms. Restricting to only strong homomorphisms makes the task easier, so we start from that case, following the lines of [BT96], where observational equivalence for partial algebras was defined in the category of partial algebras with strong morphisms.

The situation now is not so nice as in the case of standard algebras, where the forgetful functor has a left adjoint. Instead, there exists a left multiadjoint defined as follows. For any X in Set^{OBS}, the set $T^{\Sigma}(X) = \{T^{\Sigma}_i(X)\}_{i \in \mathbb{I}_X}$ consists of all partial algebras $T^{\Sigma}_i(X)$ whose carrier is a subset of the total algebra $T_{\Sigma}(X)$, which satisfy (for every $i \in \mathbb{I}_X$):

- $X \subseteq |T^{\Sigma}_i(X)| \subseteq |T_{\Sigma}(X)|$,
- carrier sets of $T^{\Sigma}_i(X)$ are closed under subterms, i.e., $t \in |T^{\Sigma}_i(X)|$ implies that all subterms of t are in $|T^{\Sigma}_i(X)|$,
- for any function symbol $f : s_1, \ldots, s_n \to s$ in Σ and terms $t_1, \ldots, t_n$ of appropriate sorts in $|T^{\Sigma}_i(X)|$, the function $f^{T^{\Sigma}_i(X)}$ is defined on $t_1, \ldots, t_n$ iff the term $f(t_1, \ldots, t_n)$ belongs to $|T^{\Sigma}_i(X)|$, and if so, then $f^{T^{\Sigma}_i(X)}(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$.

For any X in Set^{OBS}, we define the family $\eta_X = \{\eta^i_X : X \to |T^{\Sigma}_i(X)|_{OBS}\}_{i \in \mathbb{I}_X}$ of inclusions in Set^{OBS}.

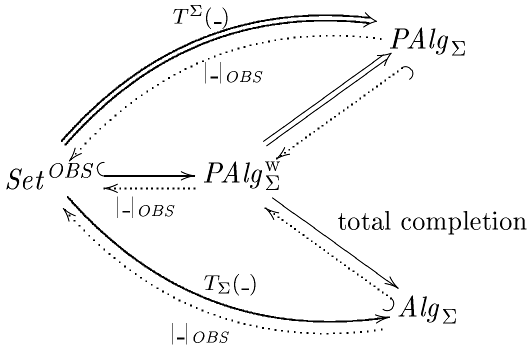
Proposition 2.2.13 *The pair $\langle T^{\Sigma} = \{T^{\Sigma}(X)\}_{X \in |Set^{OBS}|}, \eta = \{\eta_X\}_{X \in |Set^{OBS}|}\rangle$ is the left multiadjoint to the forgetful functor $|\_|_{OBS} : PAlg_{\Sigma} \to Set^{OBS}$.*

(Analogously, a left multiadjoint to a functor $|\_|_R$ could be defined, for any subset R of sorts.) Having defined the multiadjoint, we easily obtain a conclusion from Lemmas 2.2.4, 2.2.5 and Theorem 2.2.3:

Corollary 2.2.14 *The observational equivalence of partial algebras in category $PAlg_{\Sigma}$ coincides with the $\mathbb{Q}$-bisimilarity in $PAlg_{\Sigma}$, for any subcategory $\mathbb{Q}$ of $[T^{\Sigma}(Set^{OBS})]$ containing $T^{\Sigma}(\mathbf{2}_{OBS})$; subcategory $\mathbf{2}_{OBS}$ of Set^{OBS} was defined in Section 2.1.1.*

We omit the proof – the argument is essentially the same as for standard algebras in Corollary 2.1.6.

We end up with the following diagram of (multi)adjunctions between categories of standard and partial algebras (right adjoints are depicted by dotted arrows and multiadjoints by double arrows):



both triangles of which commute (up to natural isomorphism). For instance, the left multiadjoint from $PAlg^{\mathrm{w}}_{\Sigma}$ to $PAlg_{\Sigma}$ is defined similarly as $T^{\Sigma}(\_)$, with the only difference that values of an operation are freely generated only for those arguments, for which this operation is undefined. In a total completion of a partial algebra, all undefined terms yield freely generated values (cf. [Gra79]). Note that we could also use this multiadjunction to show that observational equivalence of partial algebras is definable by open maps in $PAlg_\Sigma$, having proved that it is definable in $PAlg_\Sigma^{\mathrm{w}}$; actually we proceed in the reversed order, and the latter is the subject of Section 2.4.2, where we rely on the fact that observational equivalence is definable in $PAlg_\Sigma$.

2.3 Behavioural equivalence in a concrete category

We introduce here a categorical setting, borrowed from [BT96], enabling one to define observational equivalences uniformly across different algebraic frameworks. We will intensively use this setting in Chapter 3. All the definitions below, although very abstract, follow usual algebraic intuitions. We formulate abstractly some algebraic constructions, like quotients and subobjects, sufficient for defining behavioural equivalence.

Let $\mathbb{M}$ be an arbitrary category, equipped with a concretization functor $|\_| : \mathbb{M} \to Set^S$ to the category of S-sorted sets and functions (although in many examples S will be a singleton set). We will always require $|\_|$ to be faithful. For an object A, $|A|$ can be seen as a carrier of A, hence we can think of objects of $\mathbb{M}$ as "sets with structure". Moreover, faithfulness of $|\_|$ allows us to treat morphisms as (structure preserving) functions between carrier sets.

For instance, the category TS_L of transition systems over a fixed labelling set L can be equipped with the forgetful concretization functor $|\_| : TS_L \to Set$, defined on objects by $|(S, i, t)| = S$ and on morphisms by $|f| = f$.

Similarly as in previous sections, for a subset $R \subseteq S$ of sorts and an object X of $\mathbb{M}$ we will write $|X|_R$ to denote the evident R-sorted set. Notice that the mapping $X \mapsto |X|_R$ can be extended to a functor $|\_|_R : \mathbb{M} \to Set^R$ in the obvious way.

Subobjects For an object $A \in |\mathbb{M}|$, a *subobject* of A is an object B together with a morphism $\iota_{B\hookrightarrow A} : B \to A$ such that $|\iota_{B\hookrightarrow A}|$ is an inclusion. A subobject B is *full* if for any morphism $f : C \to A$ with $|f|(|C|) \subseteq |B|$, there exists a (necessarily unique) morphism $\widehat{f} : C \to B$ satisfying $\widehat{f}; \iota_{B\hookrightarrow A} = f$. Intuitively speaking, the full subobject inherits the whole "structure" of A restricted to $|B|$. Given an object A and a set $X \subseteq |A|$, a *full subobject of A generated by X*, denoted by $\langle X\rangle_A$, is a full subobject B of A such that $X \subseteq |B|$ and such that for any other subobject C of A either $|B| \subseteq |C|$ or $X \nsubseteq |C|$. $\langle X\rangle_A$, if exists, is determined up to *an exact isomorphism* (that is, an isomorphism i with $|i|$ being identity). $\mathbb{M}$ *has full subobjects* if for each object A and a set $X \subseteq |A|$, there exists a full subobject generated by X.

Congruences and quotients For any morphism $f : A \to B$ in $\mathbb{M}$, by a *kernel* of f we mean an equivalence $\{(a, a') \in |A| \times |A| : |f|(a) = |f|(a')\}$, denoted by $\ker(|f|)$. Given an object A in $\mathbb{M}$, by a *congruence* on A we mean the kernel $\ker(|f|)$ of any morphism $f : A \to B$.

A *quotient* of an object A is an object B together with an epimorphism $\pi : A \to B$, called the *quotient projection*, such that for any morphism $f : A \to C$ with $\ker(|\pi|) \subseteq \ker(|f|)$ there exists a (necessarily unique) morphism $\bar{f} : B \to C$ satisfying $\pi; \bar{f} = f$. For a congruence $\sim$ on A, a quotient of A by $\sim$ is a quotient B of A whose projection π satisfies $\ker(|\pi|) = \sim$.

Given $\sim$, this quotient projection will be also denoted by $\pi_{A/\sim}$, and B will be written as $A/\sim$. Quotients by congruences are determined up to isomorphism. We say that $\mathbb{M}$ *has surjective quotients* if for any A and any congruence $\sim$ on A, the quotient $A/\sim$ exists and the projection $|\pi_{A/\sim}|$ is surjective.

Note The above definitions of subobjects and quotients are deliberately chosen asymmetrically, even though these concepts are usually considered as duals. The reason is that it seems natural to fix a carrier of the full subobject generated by a set, which is sometimes impossible for quotients. However, the carrier of a surjective quotient can be chosen to be a quotient set, when $\mathbb{M}$ *admits renaming of elements*, that is, for each object $A \in |\mathbb{M}|$ and a bijection $i : |A| \to X$ there is an object B together with an isomorphism $\hat{\imath} : A \to B$ meeting $|\hat{\imath}| = i$ (and $|B| = X$).

In the sequel, as well as in the next chapter, we assume that $\mathbb{M}$ has full subobjects and surjective quotients – this assumption guarantees existence of the algebraic constructions necessary to abstractly articulate the concept of behavioural equivalence.

Indistinguishability structures By a *partial congruence* on an object A we mean a congruence on a full subobject of A, called its *domain subobject*. In the sequel we often deal with a family of partial congruences consisting, for each object A in $\mathbb{M}$, of a partial congruence $\doteq_A \subseteq |A| \times |A|$. We will write $\doteq$ for $\{\doteq_A\}_{A\in|\mathbb{M}|}$, and then, for each object A, we will write $A_{\doteq}$ for a domain subobject of $\doteq_A$. When no danger of confusion arises we will omit subscripts and write $A/\doteq$ instead of $A_{\doteq}/\doteq_A$, $\pi_{A/\doteq}$ instead of $\pi_{A_{\doteq}/\doteq_A}$, etc. We call such a family $\doteq$ an *indistinguishability structure*, since $\doteq_A$ is meant to represent a (behavioural) indistinguishability in A. We say that two objects A and B are *behaviourally equivalent* (w.r.t. $\doteq$), which we denote by $A \equiv_{\doteq} B$, if $A/\doteq$ and $B/\doteq$ are isomorphic.

2.3.1 Observational equivalence

As a justification for this abstract setting, we reformulate uniformly the observational equivalences of standard, partial and regular algebras using the framework introduced above (note the similarity to Section 1.3). In the following, let OBS denote a subset of observable sorts, $OBS \subseteq S$, fixed in this section.

The indistinguishability structure $\sim^{OBS} = \{\sim_A^{OBS}\}_{A\in|\mathbb{M}|}$ is defined as follows. The domain of $\sim_A^{OBS}$ is the carrier $|A_{OBS}|$ of $A_{OBS} = \langle |A|_{OBS}\rangle_A$, the full subobject of A generated by $|A|_{OBS}$; recall that $|A|_{OBS}$ denotes the carrier sets of observable sorts. Then we define $\sim_A^{OBS}$ as the greatest congruence on A_{OBS} being identity on observable sorts (the *greatest* such congruence does not have to exist in general; it does in the categories of standard, regular and partial algebras with strong homomorphisms). The observational equivalence is an equivalence relation $\equiv_{OBS}$ *factorized* by $\sim^{OBS}$:

$$A \equiv_{OBS} B \quad \text{iff} \quad A/_{\sim_A^{OBS}} \cong B/_{\sim_B^{OBS}}.$$

Notice some informal duality in definitions of A_{OBS} and $\sim_A^{OBS}$. Let us call a subobject (quotient, congruence) *observational* whenever subobject inclusion (quotient projection) is observational (i.e., bijective on OBS). Then A_{OBS} is the smallest observational full subobject of A while $\sim_A^{OBS}$ is the greatest observational congruence on A_{OBS}.

In previous sections we based some proofs on a result from [BT96], giving a characterization of observational equivalence in a concrete category in terms of observational correspondences (Corollaries 2.1.6, 2.1.8 and 2.2.14). In fact, this is a lifting of the idea of [Sch87] (cf. definition of an observational correspondence in Section 1.3.4) to a general setting of a concrete category. By a correspondence (between A and B, objects of $\mathbb{M}$) we mean a span of morphisms

$$\begin{tikzcd} & C \arrow[dl, "f"'] \arrow[dr, "g"] & \\ A & & B \end{tikzcd} \tag{2.12}$$

where both f and g are *structural*[1]; a morphism $f : C \to A$ is structural if the full subobject $f(C) = \langle |f|(|C|)\rangle_A$ of A generated by the image of f, is a surjective quotient of C (then $f(C) = C/\mathrm{ker}(|f|)$ up to isomorphism). A correspondence is observational if both f and g are observational. We have the following characterization of $\sim^{OBS}$ that hold under some assumptions on category $\mathbb{M}$ satisfied in all categories of algebras we considered (for details, we refer to [BT96] [2]):

Theorem 2.3.1 ([BT96]) *Two objects A, B from $\mathbb{M}$ are observationally equivalent iff they are related by an observational correspondence.*

Our proofs based on this fact are valid due to an observation that in the categories of algebras we were considering, all spans are structural; this need not hold in general, hence a span of observational morphisms does not ensure the existence of an observational correspondence. This is the reason why we did not state a general result, concentrating on specific models separately. In order to look closer at the problem, assume that the functor $|\_|_{OBS} : \mathbb{M} \to Set^{OBS}$ has a left adjoint $F : Set^{OBS} \to \mathbb{M}$. This immediately gives us, by Corollary 2.1.2 in Section 2.1, the coincidence that can be informally presented as follows:

$$\text{observational} \quad \Longleftrightarrow \quad F(\mathbf{2}_{OBS})\text{-open} \quad \Longleftrightarrow \quad [F(Set^{OBS})]\text{-open},$$

analogously as in previous sections (e.g. in Corollary 2.1.7). Now, assume two objects A and B to be related by a span of observational morphisms (2.12). In [BT96] it was proved that $|f(C_{OBS})| \subseteq |A_{OBS}|$ for each morphism $f : C \to A$; on the other hand for f in a correspondence (2.12), it holds necessarily $|A|_{OBS} \subseteq |f(C_{OBS})|$, hence $|A_{OBS}| \subseteq |f(C_{OBS})|$. From all this we can conclude that

$$|A_{OBS}| = |f(C_{OBS})|,$$

but this does not prove the composition $\iota_{C_{OBS} \hookrightarrow C}; f : C_{OBS} \to A$ to be structural – to show that the span

$$\begin{tikzcd} & C_{OBS} \arrow[dl, "\iota_{C_{OBS} \hookrightarrow C}; f"'] \arrow[dr, "\iota_{C_{OBS} \hookrightarrow C}; g"] & \\ A & & B \end{tikzcd}$$

is a correspondence we would need to know that A_{OBS} is a surjective quotient of C_{OBS}. Similar technical difficulties arise also in Section 3.3, while considering factorization of open-map bisimilarity, especially in Lemma 3.3.6.

[1] We slightly changed the terminology of [BT96].

[2] According to the terminology of [BT96], we assume here that $\mathbb{M}$ is finitely directed complete, admits renaming of elements, has pullbacks which preserve quotients and that quotients are fully compatible with subobjects.

The proofs of Corollaries 2.1.6, 2.1.8 and 2.2.14 could be equivalently given without referring to the concept of observational correspondences (which involves verifying the necessary conditions for Theorem 2.3.1 above to hold) – this was done for standard algebras in Corollary 2.1.6, and omitted (as routine) for partial and regular algebras. The main tool used there was the characterization of observational equivalences in Theorems 1.3.4 and 1.3.5 by means of pairs of surjective valuations (of observable variables) identifying the same pairs of observable terms. It turns out that spans of observational maps are conceptually closer to this idea, but unfortunately ideas formulated in Theorems 1.3.4 and 1.3.5 have no generalization to an arbitrary (concrete) category of models. In the following we present an attempt for such a generalization, together with a result (Theorem 2.3.3 below) subsuming the categories of standard, regular and partial algebras with strong homomorphisms. Theorem 2.3.3 gives immediately the proofs of the three corollaries mentioned above.

We consider an arbitrary concrete category $\mathbb{M}$ with concretization functor $|\_| : \mathbb{M} \to Set^S$ such that the functor $|\_|_{OBS}$ has a left adjoint $F : Set^{OBS} \to \mathbb{M}$. Following algebraic intuitions, the basic idea is to consider a function $v : X \to |A|_{OBS}$ as a valuation of variables (of observable sorts) in an object A. Furthermore, by an observable equation in A we mean abstractly a pair (t_1, t_2) of elements of the carrier $|F(X)|_{OBS}$ of the free object over the set X of variables. Moreover, an equation is satisfied in A under v if $|v^\#|_{OBS}$ identifies t_1 and t_2, where $v^\# : F(X) \to A$. Having this, we are able to state the following definition: two objects A, B in $\mathbb{M}$ are *equationally equivalent* (w.r.t. the set OBS of observable sorts) if for some set X in Set^{OBS} there exists a span of surjective functions:

$$\begin{tikzcd} & X \arrow[dl, "v_A"'] \arrow[dr, "v_B"] & \\ |A|_{OBS} & & |B|_{OBS} \end{tikzcd} \tag{2.13}$$

such that $\ker(|v_A^\#|_{OBS}) = \ker(|v_B^\#|_{OBS})$.

For regular algebras, we are treating a carrier $|F(X)|_{OBS}$ of a free object as an abstract syntax, identifying, as we mentioned in Section 2.1.2, all α-equivalent terms as well as all "fix-point equivalent" ones. However, one can easily check that our general definition is equivalent in the case of regular algebras to that from Theorem 1.3.4. A slightly different approach to an abstract syntax of regular algebras, taking into account only α-equivalence, turns out to be convenient in Section 5.6.

One can easily adapt (2.13) for a multiadjunction; assume then the forgetful functor $|\_|_{OBS}$ to have a left multiadjoint F:

Definition 2.3.2 *Two objects A, B in $\mathbb{M}$ are* equationally equivalent *if there exists a span (2.13) of surjective functions such that:*

- *both $v_A^\#$ and $v_B^\#$ have the same domain (say $F_i(X)$) and*
- $\ker(|v_A^\#|_{OBS}) = \ker(|v_B^\#|_{OBS})$.

Keeping in mind the intuition that the set $|F_i(X)|_{OBS}$ corresponds to the set of terms of observable sorts "allowed" in A and B under a valuation v_A and v_B, respectively, this obviously subsumes equational characterization of observational equivalence of partial algebras.

Theorem 2.3.3 *Assume that for each object A in $\mathbb{M}$ and each binary relation $\sim \subseteq |A| \times |A|$, there exists the smallest congruence on A including $\sim$. Then the equational equivalence* *coincides with open-map bisimilarity induced by* $F(\mathbb{Q})$, *for any subcategory* $\mathbb{Q}$ *of* $[F(Set^{OBS})]$ *containing* $F(\mathbf{2}_{OBS})$.

Proof: Assume A and B to be related by a span of open (i.e., observational, by Lemmas 2.2.4 and 2.2.5) maps:

$$\begin{tikzcd} & C \arrow[dl, "f"'] \arrow[dr, "g"] & \\ A & & B \end{tikzcd}$$

Transporting this span via $|\_|_{OBS}$ we get a span (2.13) of bijections, that we take as candidates for valuations v_A and v_B; we put $X = |C|_{OBS}$. To verify the first condition in Definition 2.3.2, we need to show that extensions of v_A and v_B have the same domain. Let $F_i(X)$ denote the domain of $\mathrm{id}_X{}^\#$, i.e., the counit $\varepsilon_C : F_i(X) \to C$ (for a fixed C, $i \in \mathbb{I}_X$ is determined uniquely). To prove that both $v_A^\#$ and $v_B^\#$ have the same domain $F_i(X)$, we apply Lemma 2.2.2: $v_A = |f|_{OBS}$, hence we have

$$v_A^\# = (\mathrm{id}_X; v_A)^\# = (\mathrm{id}_X; |f|_{OBS})^\# = \mathrm{id}_X{}^\#; f$$

and similarly for $v_B^\#$ and g. But this means that both $v_A^\#$ and $v_B^\#$ have the domain $F_i(X)$ and the following diagram commutes:

$$\begin{tikzcd} & F_i(X) \arrow[dl, bend right, "v_A^\#"'] \arrow[d, "\varepsilon_C = \mathrm{id}_X{}^\#"] \arrow[dr, bend left, "v_B^\#"] & \\ A & C \arrow[l, "f"] \arrow[r, "g"'] & B \end{tikzcd} \tag{2.14}$$

Moreover, by transporting this diagram via $|\_|_{OBS}$ we get the following one:

$$\begin{tikzcd} & |F_i(X)|_{OBS} \arrow[dl, bend right, "|v_A^\#|_{OBS}"'] \arrow[d, "|\varepsilon_C|_{OBS}"] \arrow[dr, bend left, "|v_B^\#|_{OBS}"] & \\ |A|_{OBS} & X \arrow[l, "v_A"] \arrow[r, "v_B"'] & |B|_{OBS} \end{tikzcd}$$

from which we read out that $\ker(|v_A^\#|_{OBS}) = \ker(|v_B^\#|_{OBS})$, since $\ker(v_A) = \ker(v_B)$ (both v_A and v_B are bijections). Interestingly, for this direction we did not need the assumption on the existence of the smallest congruences generated by a relation.

For the opposite direction, assume A and B to be equationally equivalent. For X from (2.13), let $\sim$ be the smallest congruence on $F_i(X)$ containing $\ker(|v_A^\#|_{OBS})$ $(= \ker(|v_B^\#|_{OBS}))$ and let C denote a quotient $\overline{F_i(X)/\sim}$ of $F_i(X)$ by $\sim$. By the definition of quotient, there exist (unique) morphisms $\overline{v_A^\#}$ and $\overline{v_B^\#}$ making the following diagram commute:

$$\begin{tikzcd} & F_i(X) \arrow[dl, "v_A^\#"'] \arrow[d, "\pi"] \arrow[dr, "v_B^\#"] & \\ A & C \arrow[l, "\overline{v_A^\#}"] \arrow[r, "\overline{v_B^\#}"'] & B \end{tikzcd}$$

where π is the quotient projection. Our aim is now to show that $\overline{v_A^\#}$ and $\overline{v_B^\#}$ are open, i.e., observational. Notice that by the very definition of quotient

$$\ker(|v_A^\#|_{OBS}) \subseteq \ker(\pi) = \sim,$$

hence obviously

$$\ker(|v_A^\#|_{OBS}) \subseteq \ker(|\pi|_{OBS}).$$

Since the following diagram commutes (as it is obtained by transporting the last diagram via $|\_|_{OBS}$)

$$\begin{tikzcd}
 & {|F_i(X)|_{OBS}} \arrow[dl, "|v_A^\#|_{OBS}"'] \arrow[d, "|\pi|_{OBS}"] \\
{|A|_{OBS}} & {|C|_{OBS}} \arrow[l, "|\overline{v_A^\#}|_{OBS}"]
\end{tikzcd}$$

$|v_A^\#|_{OBS}$ factors through $|\pi|_{OBS}$. Moreover $|\pi|_{OBS}$ is surjective (as $\mathbb{M}$ has surjective quotients), thus we conclude easily that $|\overline{v_A^\#}|_{OBS}$ is necessarily a bijection, i.e., $\overline{v_A^\#}$ is observational. Similarly we show that $\overline{v_B^\#}$ is observational, which completes the proof. $\Box$

Note that from the above proof we can conclude that functions v_A and v_B in Definition 2.3.2 could be equivalently required to be bijections.

As we already mentioned, from Theorem 2.3.3 we can derive Corollaries 2.1.6 (standard algebras), 2.1.8 (regular algebras) and 2.2.14 (partial algebras).

2.4 Bisimilarity in pre-adjunction

In this section we study bisimilarity when the left multiadjoint to a functor $G : \mathbb{M} \to \mathbb{O}$ is not given. It is sufficient for our purposes to have a *pre-adjunction* (Definition 2.4.1) from which we extract certain multiadjunction by distinguishing the class of *ext-strong* morphisms in $\mathbb{M}$. More precisely, left pre-adjoint turns out to be the left multiadjoint to the functor G restricted to subcategory of ext-strong morphisms of $\mathbb{M}$. Then we compare the induced bisimilarities in both categories (Theorem 2.4.9) showing that they coincide for certain classes of observation objects. This coincidence gives us a tool for relating observational equivalences with bisimilarity. The results presented in this section are quite general; an illustrative example of application to partial algebras with weak homomorphisms is given in Section 2.4.2.

Definition 2.4.1 (Left pre-adjoint, pre-adjunction) *Let $G : \mathbb{M} \to \mathbb{O}$ be a functor and X be any object of category $\mathbb{O}$. Let $F(X) = \{F_i(X)\}_{i\in\mathbb{I}_X}$ and $\eta_X = \{\eta_X^i : X \to G(F_i(X))\}_{i\in\mathbb{I}_X}$ be families of objects of $\mathbb{M}$ and morphisms of $\mathbb{O}$, respectively, both indexed by the same set $\mathbb{I}_X$. We say that a pair $\langle F(X), \eta_X\rangle$ is* a family of pre-free objects over X *(with* units η_X*) if*

(i) for any object A in $\mathbb{M}$ and a morphism $f : X \to G(A)$ there exists a (not necessarily unique) pair $\langle i, f' : F_i(X) \to A\rangle$ where $i \in \mathbb{I}_X$ and f' is a morphism in $\mathbb{M}$, such that $\eta_X^i; G(f') = f$ (we call such f' an extension *of f to A),*

(ii) $\eta_X^i; G(f') = \eta_X^i; G(f'')$ implies $f' = f''$, for any $f', f'' : F_i(X) \to A$ in $\mathbb{M}$, i.e., no two distinct pairs with the same index i are allowed in (i),

(iii) for any object A in $\mathbb{M}$ and a morphism $f : X \to G(A)$, the set of indexes

$$\mathbb{I}_X^{f,A} = \{i \in \mathbb{I}_X \mid \text{ there exists } f' : F_i(X) \to A \text{ such that } \eta_X^i; G(f') = f\}$$

has the greatest element $\sqcup^{f,A}$ w.r.t. a partial order $\leq_X$ on $\mathbb{I}_X$ defined as:

$$i \leq_X j \quad \text{iff} \quad \text{there exists a morphism } \; w : F_i(X) \to F_j(X) \; \text{ such that } \; \eta_X^i; G(w) = \eta_X^j.$$

If for every object X in $\mathbb{O}$ there exists a family of pre-free objects $\langle F(X), \eta_X \rangle$ over X, we say that the pair of families $\langle F = \{F(X)\}_{X \in |\mathbb{O}|}, \eta = \{\eta_X\}_{X \in |\mathbb{O}|} \rangle$ is a left pre-adjoint *to G and that $\langle F, \eta \rangle$ and G form a* pre-adjunction.

Note Relation $\leq_X$ (for any X in $\mathbb{O}$) is a pre-order, but it is easy to observe that $\leq_X$ is a partial order up to isomorphism, namely

$$i \leq_X j \text{ and } j \leq_X i \implies F_i(X) \cong F_j(X),$$

for any $i, j \in \mathbb{I}_X$. Thus, without loss of generality we can assume that $\leq_X$ is indeed a partial order.

From now on in this section we consider a fixed pre-adjunction formed by $\langle F, \eta \rangle$ and a functor $G : \mathbb{M} \to \mathbb{O}$. Let X be an object in $\mathbb{O}$ and A be an object in $\mathbb{M}$. For any morphism $f : X \to G(A)$ in $\mathbb{O}$, by $f^\#$ we denote the unique morphism $f^\# : F_{\sqcup^{f,A}}(X) \to A$ such that $\eta_X^{\sqcup^{f,A}}; G(f^\#) = f$. We call $f^\#$ *the greatest extension* of f (to A).

Example 2.4.2 *As an example consider the category $PAlg_\Sigma^{\mathrm{w}}$ of partial algebras with weak homomorphisms. It turns out that the left multiadjoint $\langle T^\Sigma, \eta \rangle$, defined in Section 2.2.2 for category $PAlg_\Sigma$, is now a left pre-adjoint to the functor $|\_|_{OBS} : PAlg_\Sigma^{\mathrm{w}} \to Set^{OBS}$. The pre-free partial algebra $T^\Sigma_{\sqcup^{v,A}}(X)$ consists of precisely those terms over variables from X that are defined in A under a valuation $v : X \to |A|_{OBS}$. Moreover, the greatest extension $v^\# : T^\Sigma_{\sqcup^{v,A}}(X) \to A$ is the usual extension of the valuation v to terms. We expand on this example in Section 2.4.2.*

We are going now to distinguish a subclass of *ext-strong* morphisms, giving raise to a multiadjunction.

Definition 2.4.3 *A morphism $f : A \to B$ in $\mathbb{M}$ is* ext-strong *if whenever composed with the greatest extension of a morphism $p : X \to G(A)$ to A it gives the greatest extension of $(p; G(f))$ to B, i.e., if it satisfies the following*

$$p^\#; f = (p; G(f))^\#, \quad \text{for any morphism } p : X \to G(A).$$

The condition above is equivalent to: $\sqcup^{p,A} = \sqcup^{(p;G(f)),B} \in \mathbb{I}_X$, for any X and $p : X \to G(A)$ in $\mathbb{O}$ (observe that in general it only holds that $\sqcup^{p,A} \leq_X \sqcup^{(p;G(f)),B}$, since $p^\#; f$ necessarily extends $(p; G(f))$ to B). Roughly, composition of the greatest extension with ext-strong morphism gives the greatest extension as well. The opposite holds for any morphism and we state this in a separate lemma for future references:

Lemma 2.4.4 *Let $h : A \to B$ in $\mathbb{M}$ and $p' : F_i(X) \to A$ be any extension of $p : X \to G(A)$ (i.e., $p = \eta_X^i; G(p')$). If $(p'; h)$ is the greatest extension of $p; G(h)$ then p' is the greatest extension of p.*

Proof: Assume that $(p'; h)$ is the greatest extension of $p; G(h)$ and p' is not the greatest extension of p. Let $p'' : F_j(X) \to A$ be the greatest extension of p; consequently $i <_X j$. But now $(p''; h)$ is also an extension of $(p; G(h))$ as well as $(p'; h)$. Moreover, $i <_X j$, and so $(p'; h)$ is not the greatest extension of $(p; G(h))$. The contradiction completes the proof. $\square$

Lemma 2.4.5 *Every ext-strong extension of a morphism f is the greatest extension of f.*

Proof: Let a morphism $f' : F_i(X) \to A$ in $\mathbb{M}$ be an ext-strong extension of $f : X \to G(A)$. We will show that f' is necessarily the greatest extension of f.

Obviously $\mathrm{id}_{F_i(X)}; f' = f'$. Moreover, $\mathrm{id}_{F_i(X)} = (\eta^i_X)^\#$, i.e., $\mathrm{id}_{F_i(X)}$ is the greatest extension of η^i_X. Since f' is ext-strong , $\mathrm{id}_{F_i(X)}; f'$ is the greatest extension of $\eta^i_X; G(f')$. But $\eta^i_X; G(f') = f$, hence f' is the greatest extension of f. □

By this lemma, each morphism f has at most one ext-strong extension. The greatest extensions need not to be in general ext-strong and a simple counter-example can be constructed. But in many typical situations these two notions coincide (cf. Section 2.4.2), which justifies the following assumption, needed below for the results in this section:

Assumption 2.4.6 *Every greatest extension is ext-strong, i.e., for any objects X, A and $f : X \to G(A)$, $f^\#$ is ext-strong.*

It follows that the greatest extensions compose, similarly as ext-strong morphisms do. Moreover, identities (in fact, all isomorphisms) are ext-strong. Now, let $(\mathbb{M})^{\mathrm{ext-strong}}$ denote the subcategory of $\mathbb{M}$ consisting of all objects with ext-strong morphisms only. Due to Assumption 2.4.6 we recover some functorial property of a left pre-adjoint. Even without this assumption we can define "$F(f)$" $= \{F_j(f)\}_{j \in \mathbb{I}_Y}$ similarly as for multiadjunctions, where for $f : X \to Y$ in $\mathbb{O}$ and $j \in \mathbb{I}_Y$,

$$F_j(f) = (f; \eta^j_Y)^\# : F_{\bigsqcup(f;\eta^j_Y), F_j(Y)}(X) \to F_j(Y).$$

Having assumed that greatest extensions are ext-strong, we obtain the preservation of composition:

$$F_j(f); F_k(g) = F_k(f; g),$$

for $X \xrightarrow{f} Y \xrightarrow{g} Z$ in $\mathbb{O}$ and appropriately chosen $j \in \mathbb{I}_Y$. Roughly, this brings us closer to the notion of multiadjunction, and in consequence it is not surprising that we have:

Corollary 2.4.7 *The pair $\langle F, \eta \rangle$ is the left multiadjoint to G restricted to* $(\mathbb{M})^{\mathrm{ext-strong}}$.

Proof: Due to Assumption 2.4.6 all greatest extensions are in $(\mathbb{M})^{\mathrm{ext-strong}}$, hence for each object A in $\mathbb{M}$, every morphism $m : X \to G(A)$ in $\mathbb{O}$ has at least one extension to A, namely the greatest extension $m^\# : F_{\bigsqcup f, A}(X) \to A$. From Lemma 2.4.5 it follows that $m^\#$ is the only extension of m to A in $(\mathbb{M})^{\mathrm{ext-strong}}$. □

Let $\mathbb{P}$ be a fixed subcategory of $\mathbb{O}$, similarly as in Section 2.2. As a subcategory of observation objects in $\mathbb{M}$ we take the subcategory $F(\mathbb{P})$ whose objects are all pre-free objects and morphisms are all extensions of morphisms $(m; \eta^j_Y)$ for any morphism $m : X \to Y$ in $\mathbb{O}$ and any $j \in \mathbb{I}_Y$. We will deliberately use the same notation $F(\mathbb{P})$ for this subcategory of $\mathbb{M}$ as well as for different in general on morphisms subcategory $F(\mathbb{P})$ of $(\mathbb{M})^{\mathrm{ext-strong}}$ determined by the left multiadjoint (this subcategory was defined in Section 2.2).

In Theorem 2.4.9 below we relate $F(\mathbb{P})$-openness in $\mathbb{M}$ and $F(\mathbb{P})$-openness in $(\mathbb{M})^{\mathrm{ext-strong}}$. It turns out that if all $F(\mathbb{P})$-open morphisms in $\mathbb{M}$ are ext-strong (i.e., they are also in $(\mathbb{M})^{\mathrm{ext-strong}}$; obviously it is not always the case, for example for $\mathbb{P}$ being the empty category), then the notions of $F(\mathbb{P})$-openness coincide in both categories. The following easy lemma will be useful in the proof:

Lemma 2.4.8 *Let $f : F_i(X) \to A$ and $f' : F_j(X) \to A$ be two extensions of the same morphism, such that $i \leq_X j$, and let $s : F_i(X) \to F_j(X)$ satisfy $\eta^i_X; G(s) = \eta^j_X$ (existence of such a morphism is guaranteed by $i \leq_X j$). Then we have:*

1. *$f = s; f'$,*

2. *s is determined uniquely (being an extension of the unit η^j_X) and is epi.*

(The observation that every morphism s witnessing the pre-order $\leq_X$ is an epimorphism indicates that the pre-free objects over the same object X are closely related to each other.)

Theorem 2.4.9 *Let $h : A \to B$ be a morphism in $(\mathbb{M})^{\mathrm{ext-strong}}$. Then h is $F(\mathbb{P})$-open in $\mathbb{M}$ iff h is $F(\mathbb{P})$-open in $(\mathbb{M})^{\mathrm{ext-strong}}$.*

Proof: $\Leftarrow$: To show $F(\mathbb{P})$-openness of h in $\mathbb{M}$ we need to find a diagonal morphism r in every commuting square:

$$\begin{tikzcd} F_i(X) \arrow[r, "p^i"] \arrow[d, "m"'] & A \arrow[d, "h"] \\ F_j(Y) \arrow[r, "q^j"'] \arrow[ur, dotted, "r"] & B \end{tikzcd} \tag{2.15}$$

where m is an extension of $(n; \eta^j_Y)$, for some morphism $n : X \to Y$ in $\mathbb{P}$ (i.e., m is in $F(\mathbb{P})$), p^i, q^j are extensions of $p : X \to G(A)$ and $q : Y \to G(B)$, respectively. We construct another commuting square:

$$\begin{tikzcd} F_k(X) \arrow[r, "p^{\#}"] \arrow[d, "m'"'] & A \arrow[d, "h"] \\ F_l(Y) \arrow[r, "q^{\#}"'] \arrow[ur, dotted, "r'"] & B \end{tikzcd} \tag{2.16}$$

where $k = \sqcup^{p,A} \in \mathbb{I}_X$, $l = \sqcup^{q,B} \in \mathbb{I}_Y$ (i.e., $p^{\#}$ and $q^{\#}$ are the greatest extensions of p and q, respectively). By Lemma 2.4.8(1) there exist morphisms $p' : F_i(X) \to F_k(X)$ and $q' : F_j(X) \to F_l(X)$ making

$$p'; p^{\#} = p^i \quad \text{and} \quad q'; q^{\#} = q^j. \tag{2.17}$$

Let m' be the greatest extension of $(n; \eta^l_Y) = (n; \eta^j_Y; G(q'))$. We should prove that both $p^{\#}$ and m' have the same domain objects (i.e., $k = \sqcup^{(n;\eta^l_Y), F_l(Y)}$). We know that $q^{\#}$ is ext-strong (by Assumption 2.4.6), thus $(m'; q^{\#})$ is the greatest extension of the morphism $(n; \eta^l_Y; G(q^{\#})) = (n; q)$. On the other hand $(p^{\#}; h)$ is the greatest extension (as h is ext-strong) of $(p; G(h))$. It is easy to observe that $p; G(h) = n; q$ — this follows from commutativity of (2.15), thus both $(p^{\#}; h)$ and $(m'; q^{\#})$ are the greatest extensions of the same morphism. We conclude that $p^{\#}$ and m' have the same domain and moreover that square (2.16) commutes. This square consists of only ext-strong morphisms (by Assumption 2.4.6), thus from $F(\mathbb{P})$-openness of h in $(\mathbb{M})^{\mathrm{ext-strong}}$ we obtain a diagonal morphism r' in diagram (2.16). We will show that $r := q';r'$ is diagonal in (2.15). Obviously, from (2.17) we conclude $r;h = q^j$, while equality $m;r = p^i$ follows from commutativity of

$$\begin{tikzcd} F_i(X) \arrow[r, "p'"] \arrow[d, "m"'] & F_k(X) \arrow[d, "m'"] \\ F_j(Y) \arrow[r, "q'"'] & F_l(Y) \end{tikzcd}$$

(this square commutes by Lemma 2.4.8(1) since $\eta^i_X; G(p') = \eta^k_X$ and and both m' and $(m;q')$ are extensions of $(n;\eta^l_Y))$.

$\Rightarrow$: To show $F(\mathbb{P})$-openness of h in $(\mathbb{M})^{\text{ext-strong}}$ we need only to prove that in every commuting square (2.15) in $\mathbb{M}$ a diagonal morphism is ext-strong whenever p^i, q^j and m are ext-strong, i.e., (by Lemma 2.4.5) whenever p^i, q^j and m are greatest extensions. This follows from Lemma 2.4.4 (it is only necessary for q^j to be the greatest extension) and from Assumption 2.4.6. $\square$

Theorem 2.4.9 can be applied when $\mathbb{P} = \mathbb{O}$ (see Corollary 2.4.13 below), as the following lemma holds:

Lemma 2.4.10 *Every $F(\mathbb{O})$-open morphism in $\mathbb{M}$ is ext-strong.*

Proof: Let $h : A \to B$ be any $F(\mathbb{O})$-open morphism in $\mathbb{M}$, $p^\# : F_i(X) \to A$ be the greatest extension of $p : X \to G(A)$. We will show that $p^\#; h$ is the greatest extension of $(p; G(h))$, i.e., $i = j$, where $j := \sqcup^{(p;G(h)),B}$. It is easy to verify that $p^\#; h$ is an extension of $(p; G(h))$, i.e., that $i \leq_X j$. Let us consider the greatest extension $g : F_j(X) \to B$ of $(p; G(h))$. We will show $j \leq_X i$. Let $s : F_i(X) \to F_j(X)$ be a morphism meeting $\eta^i_X; G(s) = \eta^j_X$ (which exists as $i \leq_X j$). Consider a diagram:

$$\begin{tikzcd} F_i(X) \arrow[r, "p^\#"] \arrow[d, "s"'] & A \arrow[d, "h"] \\ F_j(X) \arrow[r, "g"'] \arrow[ur, dotted, "r"] & B \end{tikzcd}$$

Obviously, s is in $F(\mathbb{O})$ as it is an extension of η^j_X. The square commutes by Lemma 2.4.8(1). By $F(\mathbb{O})$-openness of h there exists a diagonal morphism r. A crucial observation is that r extends p: $\eta^j_X; G(r) = \eta^i_X; G(s;r) = \eta^i_X; G(p^\#) = p$. This gives $j \leq_X i$ and completes the proof. $\square$

As a conclusion from Theorem 2.4.9 and Lemma 2.4.10 we obtain:

Corollary 2.4.11 *A morphism $h : A \to B$ in $\mathbb{M}$ is $F(\mathbb{O})$-open iff h is $F(\mathbb{O})$-open in $(\mathbb{M})^{\text{ext-strong}}$.*

2.4.1 Determinism

By Lemma 2.2.9 we conclude that all the bisimilarities considered so far, induced by an adjunction or a multiadjunction, are strongly deterministic. Here we prove that pre-adjunction also induces strongly deterministic bisimilarity if the related multiadjunction does so.

Lemma 2.4.12 *Let $\mathbb{P}$ be a subcategory of $\mathbb{O}$ as in Theorem 2.4.9. Whenever $F(\mathbb{P})$-bisimilarity in $(\mathbb{M})^{\mathrm{ext-strong}}$ is strongly deterministic then $F(\mathbb{P})$-bisimilarity in $\mathbb{M}$ is so as well.*

Proof: Assume $r'; h = s'; h$ for certain $F(\mathbb{P})$-open morphism h in $\mathbb{M}$. We will show $r' = s'$. Let r', s' have the domain, say, $F_i(X)$. Let $r = \eta^i_X; G(r')$, $s = \eta^i_X; G(s')$. We have $r; G(h) = s; G(h)$, hence $(r; G(h))^{\#} = (s; G(h))^{\#}$. Due to the fact that h is ext-strong (as it is $F(\mathbb{P})$-open) we conclude $r^{\#}; h = (r; G(h))^{\#}$ and $s^{\#}; h = (s; G(h))^{\#}$, which gives us $r^{\#}; h = s^{\#}; h$. Morphism h is ext-strong and open in $(\mathbb{M})^{\mathrm{ext-strong}}$ by Theorem 2.4.9, so $r^{\#} = s^{\#}$ (as $F(\mathbb{P})$-bisimilarity in $(\mathbb{M})^{\mathrm{ext-strong}}$ is strongly deterministic), thus $r = s$. It means that $\eta^i_X; G(r') = \eta^i_X; G(s')$, from which we conclude $r' = s'$, as more that one extension of the same morphism is not allowed from the same pre-free object. $\Box$

Fact analogous to Observation 2.2.6 holds also in this case, and can be concluded directly from the above lemma:

Corollary 2.4.13 *$F(\mathbb{O})$-bisimilarity is strongly deterministic for any left pre-adjoint F.*

2.4.2 Partial algebras with weak homomorphisms

As discussed in Section 2.2, the forgetful functor $|\_|_{OBS} : PAlg^{\mathrm{w}}_{\Sigma} \to Set^{OBS}$ has the left adjoint, but the full subcategory of free objects (in which all operations are totally undefined) is too small to induce the expected observational equivalence of partial algebras. Instead, we will use pre-free objects of the pre-adjunction defined in Example 2.4.2 – recall that the left pre-adjoint is the same as the left multiadjoint $\langle T^{\Sigma}, \eta\rangle$, defined in Section 2.2.2 for partial algebras with strong homomorphisms. In the pre-adjunction under consideration, the greatest extension of a function $v : X \to |A|_{OBS}$ to an algebra A is an extension $v^{\#}$ of v from the multiadjunction.

In the following we will need some weaker kind of strong morphisms. We say that $f : A \to B$ in $PAlg^{\mathrm{w}}_{\Sigma}$ is *R-strong* (for $R \subseteq sorts(\Sigma)$) if it reflects definedness of operations on the carrier of the subalgebra $\langle |A|_R \rangle_A$ of A generated by sorts R. The following lemma gives an abstract characterization of OBS-strong partial homomorphisms:

Lemma 2.4.14 *A morphism $f : A \to B$ in $PAlg^{\mathrm{w}}_{\Sigma}$ is OBS-strong iff it is ext-strong w.r.t. the pre-adjunction $\langle T^{\Sigma}, \eta\rangle$.*

Proof: For any function $v : X \to |A|_{OBS}$ in Set^{OBS} let T^v_A denote the set of terms over X defined in A under the valuation v, or equivalently, the sum of carrier sets of all pre-free algebras over X from which there exists an extension of v. Consider any morphism $h : A \to B$ in $PAlg^{\mathrm{w}}_{\Sigma}$. Now we have an easy characterizations of both notions by means of sets T^v_A and $T^{(v;|h|_{OBS})}_B$: morphism h is ext-strong (respectively, OBS-strong) iff

$$T^v_A = T^{(v;|h|_{OBS})}_B, \text{ for any } \ v : X \to |A|_{OBS} \text{ in } \ Set^{OBS}.$$

$\Box$

Assumption 2.4.6 is satisfied in this case:

Lemma 2.4.15 *Every greatest extension in $PAlg^{\mathrm{w}}_{\Sigma}$ w.r.t. $\langle T^{\Sigma}, \eta\rangle$ is OBS-strong.*

Proof: For every extension $v^i : T_i^\Sigma(X) \to A$ which is not OBS-strong we can construct another ("greater") extension $v^j : T_j^\Sigma(X) \to A$, such that the carrier set of $T_j^\Sigma(X)$ is a superset of $|T_i^\Sigma(X)|$. For example $|T_j^\Sigma(X)|$ can contain precisely one element more, being application of an operation of Σ that is defined in A on the values of some terms from $|T_i^\Sigma(X)|$ under the valuation $\eta_X^i; |v^i|_{OBS}$. Such an operation and terms exist since v^i is not OBS-strong. This implies that v^i is not the greatest extension. □

Category $(PAlg_\Sigma^{\mathrm{w}})^{\mathrm{ext-strong}}$ contains more morphisms than $PAlg_\Sigma$, namely all OBS-strong partial homomorphisms. Now, by Corollary 2.4.11, the class of $T^\Sigma(Set^{OBS})$-open morphisms in category $PAlg_\Sigma^{\mathrm{w}}$ is equal to the class of $T^\Sigma(Set^{OBS})$-open morphisms in $(PAlg_\Sigma^{\mathrm{w}})^{\mathrm{ext-strong}}$. The subtle point here is that the left multiadjoint to $|\_|_{OBS}$ for category $PAlg_\Sigma$ works also for $(PAlg_\Sigma^{\mathrm{w}})^{\mathrm{ext-strong}}$, since every OBS-strong extension is a strong homomorphism. Since all pre-free algebras are generated by observable sorts, $T^\Sigma(Set^{OBS})$-bisimilarity in $(PAlg_\Sigma^{\mathrm{w}})^{\mathrm{ext-strong}}$ coincides with $T^\Sigma(Set^{OBS})$-bisimilarity in $PAlg_\Sigma$. This implies the following characterization of observational equivalence of partial algebras in terms of open morphisms in the category of weak partial homomorphisms:

Corollary 2.4.16 *The observational equivalence of partial Σ-algebras coincides with the $T^\Sigma(Set^{OBS})$-bisimilarity in the category $PAlg_\Sigma^{\mathrm{w}}$.*

Proof: By direct application of Corollary 2.4.11, $T^\Sigma(Set^{OBS})$-bisimilarities in categories $PAlg_\Sigma^{\mathrm{w}}$ and $(PAlg_\Sigma^{\mathrm{w}})^{\mathrm{ext-strong}}$ coincide. Moreover, by Corollary 2.2.14, observational equivalence of partial algebras coincides with $T^\Sigma(Set^{OBS})$-bisimilarity in $PAlg_\Sigma$. Thus we have only to prove:

$$A \sim_{T^\Sigma(Set^{OBS})} B \ \text{ in } \ PAlg_\Sigma \quad \iff \quad A \sim_{T^\Sigma(Set^{OBS})} B \ \text{ in } \ (PAlg_\Sigma^{\mathrm{w}})^{\mathrm{ext-strong}},$$

for any partial algebras A and B. The implication from left to right is trivial, as strong partial homomorphisms are OBS-strong. To show the opposite implication, observe that when there exists a span $A \xleftarrow{f} C \xrightarrow{g} B$ of observational OBS-strong morphisms, then morphisms $(\iota_{C_{OBS} \hookrightarrow C}; f)$ and $(\iota_{C_{OBS} \hookrightarrow C}; g)$ are strong, both obtained by pre-composing with the subobject inclusion. This means that the span induced by these morphisms is in $PAlg_\Sigma$; moreover since they both are obviously observational, we conclude that A and B are $T^\Sigma(Set^{OBS})$-bisimilar in $PAlg_\Sigma$, which completes the argument. □

We are able to restrict the path subcategory to the subcategory of $T^\Sigma(Set^{OBS})$ consisting of algebras having finite carrier set (i.e., the distinct union of carrier sets of all sorts is finite – this clarification is relevant when the set of sorts is infinite). Equivalently, this subcategory consists precisely of those pre-free algebras whose carrier sets are given by a closure of a finite set of terms $\{t_1, \ldots, t_n\}$, with variables of observable sorts, under subterms. In fact, we do not even need all such algebras – in the following we restrict to the subcategory $\mathbb{T}_{OBS}$, satisfying an additional requirement: terms $t_1, \ldots, t_n$ above have no *proper* subterms of observable sorts besides variable occurrences (in other words, the only subterms of observable sort of each term t_i are variable occurrences and possibly t_i itself). Such terms will be called *OBS-atomic* in the sequel. Note that $\mathbb{T}_{OBS}$ is not a full subcategory of $PAlg_\Sigma^{\mathrm{w}}$ in general.

A pre-free algebra induced by OBS-atomic terms $\{t_1, \ldots, t_n\}$ in the way described above we denote by $[t_1, \ldots, t_n]$. Recall from Section 2.2.2 that operations in $[t_1, \ldots, t_n]$ are defined canonically as in a term algebra, or are undefined. Below we prove that observational equivalence of partial algebras coincides with the bisimilarity induced in $PAlg_\Sigma^{\mathrm{w}}$ by $\mathbb{T}_{OBS}$.

Lemma 2.4.17 *A (weak) partial homomorphism is $T^{\Sigma}(Set^{OBS})$-open iff it is $\mathbb{T}_{OBS}$-open.*

Proof: As $\mathbb{T}_{OBS}$ is a subcategory of $T^{\Sigma}(Set^{OBS})$, we have only to prove that $\mathbb{T}_{OBS}$-openness implies $T^{\Sigma}(Set^{OBS})$-openness, i.e., by Corollary 2.4.11 and Lemma 2.4.14 that every $\mathbb{T}_{OBS}$-open partial homomorphism is *i)* bijective on OBS and *ii)* OBS-strong. Let h be a $\mathbb{T}_{OBS}$-open partial homomorphism.

i) Let $s \in OBS$ be a fixed observable sort. In diagrams below we ignore the carrier sets of pre-free algebras on other sorts, we can assume for example that they are empty sets. Now, assume that $|B|_s$ is non-empty.

Surjectivity of $|h|_s$ on the carrier of sort s follows from the openness condition for the right-bottom triangle of the commuting diagram (we draw only carrier sets of sort s of pre-free algebras):

$$\begin{tikzcd} {[\emptyset]} \arrow[r, "i_A"] \arrow[d, "i_{[x]}"'] & A \arrow[d, "h"] \\ {[x]} \arrow[ur, "r"] \arrow[r, "q"'] & B \end{tikzcd}$$

(x is a variable of a sort s, i_X is the unique homomorphism from the initial partial algebra $[\emptyset]$ to X), as q can point to any element of $|B|_s$.

To show injectivity of $|h|_s$, assume $h(a_1) = h(a_2)$, for $a_1, a_2 \in |A|_s$. Consider the commuting diagram:

$$\begin{tikzcd} {[x_1, x_2]} \arrow[r, "p"] \arrow[d, "m"'] & A \arrow[d, "h"] \\ {[y]} \arrow[ur, "r"] \arrow[r, "q"'] & B \end{tikzcd}$$

where $p(x_i) = a_i$, $m(x_i) = y$ $(i = 1, 2)$, $q(y) = h(a_1)$, x_1, x_2, y are variables of sort s. Any morphism $r : [y] \to A$ making the upper-left triangle commute satisfies $r(y) = a_i$, hence $a_1 = a_2$.

ii) For any $f : s_1, \ldots, s_n \to s$ in Σ, $a_1 \in A_{s_1}, \ldots, a_n \in A_{s_n}$, such that $a_i \in \langle |A|_{OBS} \rangle_A$, $i = 1, \ldots, n$, let us assume that f^B is defined on $h(a_1), \ldots, h(a_n)$. For any a_i, there exist a term t_i with variables of observable sorts only, and a valuation $v_i : var(t_i) \to |A|_{OBS}$ such that $(t_i)_{A[v_i]} = a_i$. We can assume that sets $var(t_i)$ are pair-wise disjoint, hence there exists a common valuation v making $(t_i)_{v[A]} = a_i$, for all $i = 1, \ldots, n$. We define the commuting diagram as follows.

$$\begin{tikzcd} {[t_1, \ldots, t_n]} \arrow[r, "p"] \arrow[d, "m"'] & A \arrow[d, "h"] \\ {[f(t_1, \ldots, t_n)]} \arrow[ur, "r"] \arrow[r, "q"'] & B \end{tikzcd}$$

Let $m : [t_1, \ldots, t_n] \to [f(t_1, \ldots, t_n)]$ be the canonical inclusion. Morphism p is an extension of v to A and q is an extension of $(v; |h|_{OBS})$ to B. Any diagonal morphism $r : [f(t_1, \ldots, t_n)] \to A$ (making the bottom-right triangle commute) maps term $f(t_1, \ldots, t_n)$ to $f^A(a_1, \ldots, a_n)$, thus f^A is defined on $a_1, \ldots, a_n$. $\square$

Roughly speaking, the subcategory $\mathbb{T}_{OBS}$ of finite partial pre-free algebras plays in the category $PAlg_{\Sigma}^{\mathrm{w}}$ a role of finite computations, analogously as the subcategory $Bran_L$ of finite linear transition systems in the category of transition systems [JNW93].

Let us summarize the results concerning open bisimilarity in category $PAlg_{\Sigma}^{\mathrm{w}}$, as a direct conclusion from Corollary 2.4.16, Lemma 2.4.17, Corollary 2.4.13 and Lemma 2.2.11:

Corollary 2.4.18 *The observational equivalence of partial algebras coincides with the* $\mathbb{Q}$-*bisimilarity in* $PAlg_{\Sigma}^{\mathrm{w}}$, *for any subcategory* $\mathbb{Q}$ *of* $[T^{\Sigma}(Set^{OBS})]$ *containing* $\mathbb{T}_{OBS}$.

The coincidence of $T^{\Sigma}(Set^{OBS})$- and $[T^{\Sigma}(Set^{OBS})]$-bisimilarity follows from the fact that the first one is deterministic (see Lemma 2.2.11 in Section 2.2.1).

2.4.3 Other applications

The framework of a pre-adjunction, developed in this section, seems to be applicable also in other contexts. Intuitively, our results are intended to capture all the situations, when in the category under consideration there exists a subclass of "strong" morphisms, transporting a kind of structure of objects relevant to the observational equivalence.

As an example, our results can be applied directly for instance to the so called *algebras with OK predicates* [DEGL84], designed to allow for error values of operations. They differ from the standard algebras in one respect: the carrier set has a distinguished subset of *OK values*, i.e., those which, taken as arguments of an operation, guarantee an error-free result; moreover, the OK values are preserved by morphisms.

For the observational equivalence of the algebras with OK predicates, it is reasonable that it is observable whether an operation of a non-observable sort yields an error value. This is similar to the observational equivalence of partial algebras, where the undefinedness of non-observable operations is observable.

Observational equivalence of algebras with OK predicates can be treated analogously as that of partial algebras. Namely, a similar pre-adjunction can be constructed, and similarly it can be shown that the open maps induced by the pre-free objects are the observational homomorphisms which moreover reflect OK values on the subalgebra generated by observable sorts. In that case, similarly as in the case of partial algebras, there exists a left adjoint to the forgetful functor, but the free objects are not sufficient as the path subcategory. The free objects are, roughly, the standard free algebras equipped with empty sets of OK values, hence they do not enable to observe whether the non-observable operations yield error values. We omit here further technical details.

On should expect that the general results on a pre-adjunction can be easily applied to other algebraic frameworks.

Chapter 3

Factorization

The main problem investigated in this chapter is to find a family of relations factorizing an open-map bisimilarity. That is to say, for an open-map bisimilarity induced by a fixed path subcategory, we search for a family of partial congruences $\{\sim_A\}$ such that two objects A and B are bisimilar if and only if their quotients are isomorphic:

$$\begin{tikzcd}
A \arrow[dr] & & & B \arrow[dl] \\
& A/_{\sim_A} \cong & B/_{\sim_B} &
\end{tikzcd}$$

Our motivation is the observation that for strong bisimilarity of transition systems such a factorization exists and is given by the greatest bisimulations (Section 3.1). In Section 3.3 we present a successful attempt to imitate these properties in an arbitrary concrete category. We give general conditions that are sufficient, under some mild assumptions, for the greatest congruences induced by open maps (the greatest *open congruences*) to factorize bisimilarity. These very general results apply to some important examples: transition systems (which fit into the general scheme) and presheaf models [JM94], proposed in [JNW93] as a uniform model for concurrency.

In some relevant cases the greatest open congruences need not exist, for instance in the category of event structures [NW95]. In Section 3.4 we prove that the factorization result as above holds under a weaker condition, namely when the quotients by all maximal open congruences are isomorphic. Moreover, this condition turns out to be necessary, which gives an elegant necessary and sufficient condition for the factorization of a wide class of bisimilarities defined by open maps. The crucial example is the subcategory of *simple* event structures (Section 3.5), failing to have greatest open congruences, but satisfying the weaker requirement.

This results are not the only attempt to define bisimilarity by means of co-spans. An observation central to [C85, BB88] is that two transition systems are strongly bisimilar precisely when they are related by a co-span of open maps (zig-zag maps). Our work is hence a generalization of this development specific for transition systems. Such an abstract approach was possible since we started from a better position, equipped with a notion of an open map, to our knowledge not familiar to the computer science community before early nineties. Our results give a response to the negative statement from [JNW93] that the "open co-span" approach cannot be applied to the bisimilarity of *transition systems with independence* [NW95]. Transition systems with independence generalize both transition systems and event structures via coreflective adjunctions (with left adjoints fully and faithfully embedding transition systems and event structures). Our results on event structures give some bound for this negative statement.

Our approach offers some potential for further investigations. For instance Lemmas 3.1.1 and 3.6.4 suggest some general line of establishing coincidence between bisimulations (i.e., relations represented by open spans) and open congruences, giving automatically the greatest open congruences. Another challenge is to generalize the proof of Theorem 3.5.8, specific for event structures, possibly under an assumption that open quotients are split epi.

The crucial role in our development is played by the domain subobjects (cf. Section 2.3) generated by images of all computations. In the last section we present another possibility to define a domain subobject of A, as a canonical colimit of all computations in A (more precisely, a colimit of the diagram consisting of all computation domains and computation extensions). This construction, perhaps more natural for category theorists, ensures without any special assumptions that open maps between domain subobjects are quotient projections, which is the key fact needed in the proof of Theorem 3.3.4. Unfortunately, the price to pay is that these subobjects are not necessarily full. They are full in all examples considered, since under assumptions from Theorem 3.3.4 they coincide with subobjects generated by images of computations.

3.1 Motivating example

We start in this section with presentation of some intuition based on strong bisimilarity between transition systems (see Section 1.4). This should be a preparation for Sections 3.2, 3.3 and 3.4, which are more general and abstract, hence unavoidably less intuitive.

Let us focus for a while on the category TS_L of transition systems, and try to find out an indistinguishability structure $\simeq = \{\simeq_T\}_{T\in|TS_L|}$ factorizing strong bisimilarity. For a domain subobject $T_\simeq$ of a transition system T it is reasonable to choose its part "observable" by means of computations; the good candidate is the subobject consisting of all reachable states of T and inheriting all the transitions between reachable states. Observe that $T_\simeq$ defined in this way is precisely the full subobject generated by the union of all images of computations in T (compare with (3.3) in Section 3.2).

As the indistinguishability relation $\simeq_T$ on $T_\simeq$ take the greatest bisimulation between $T_\simeq$ and $T_\simeq$ itself. The following lemma justifies this choice and implies that $\simeq_T$ is actually a congruence.

Lemma 3.1.1 *An equivalence $\sim \subseteq |T_\simeq| \times |T_\simeq|$ is a bisimulation iff it is a kernel of a zig-zag morphism from $T_\simeq$.*

Proof: $\Leftarrow$: It is enough to show that for any zig-zag morphism $f : T_\simeq \to U$, $f(s_1) = f(s_2)$ and $s_1 \xrightarrow{a} s_1'$ in T implies that $s_2 \xrightarrow{a} s_2'$ in T and $f(s_1') = f(s_2')$, for certain s_2'. But this follows easily from zig-zag property, namely $f(s_2) \xrightarrow{a} f(s_1')$ implies that exists s_2' in $T_\simeq$ such that $s_2 \xrightarrow{a} s_2'$ and $f(s_2') = f(s_1')$.

$\Rightarrow$: Assuming an equivalence $\sim$ to be a bisimulation, we construct an object U being the quotient of $T_\simeq = (S, r, tran)$ by $\sim$. We define U as $(S/_\sim, [r]_\sim, tran')$, where $([s_1]_\sim, a, [s_2]_\sim) \in tran'$ iff $(s_1, a, s_2') \in tran$ and $s_2' \sim s_2$, for some s_2'. This transition relation is well defined as $\sim$ is a bisimulation. The canonical quotient projection $\pi : T_\simeq \to U$, $\pi(s) = [s]_\sim$, is a morphism in TS_L, moreover $\sim = \ker(|\pi|)$. It is easy to verify that π satisfies the zig-zag condition. □

Due to this lemma, $\simeq_T$ is the greatest kernel of a zig-zag morphism with domain $T_{\simeq}$. In the sequel, a kernel of a zig-zag morphism is called a *zig-zag congruence.* One can check that strong bisimilarity $\sim_{Bran_L}$ is factorized by $\simeq$, i.e.,

$$T_1 \sim_{Bran_L} T_2 \ \text{ iff } \ T_1/_{\simeq} \cong T_2/_{\simeq}. \tag{3.1}$$

Equivalently, strong bisimilarity $\sim_{Bran_L}$ coincides with the behavioural equivalence $\equiv_{\simeq}$ in the category of transition systems.

In Section 3.3 we redo this scheme in the framework of an arbitrary concrete category (zig-zag congruences are replaced by kernels of open maps), and find some necessary and sufficient conditions for the factorization analogous to (3.1) to hold. Moreover, (3.1) is an immediate corollary from Theorem 3.3.4 (see Corollary 3.3.7), which is applicable due to the fact that transition systems have the greatest zig-zag congruences.

While reading Section 3.3 it may be helpful to keep in mind algebraic observational equivalence formulated abstractly in Section 2.3, where indistinguishability relation was defined as the *greatest* observational congruence on the *smallest* observational subobject. In the following section we move this subobject/quotient duality to the framework of open maps.

3.2 Open congruences and subobjects

Let $\mathbb{M}$ be an arbitrary category of models of computation with some fixed subcategory $\mathbb{P}$ of observation objects. We assume throughout the rest of this chapter that:

- category $\mathbb{M}$ has pullbacks,
- $\mathbb{M}$ is equipped with a faithful concretization functor $|\_| : \mathbb{M} \to Set^S$, for some set S,
- the concrete category $(\mathbb{M}, |\_|)$ has full subobjects (see Section 2.3).
- $(\mathbb{M}, |\_|)$ has surjective quotients.

Recall that existence of pullbacks in $\mathbb{M}$ implies that $\mathbb{P}$-bisimilarity is an equivalence, since pullbacks of open maps are open [JNW93]. We state here some general properties of open maps, useful in the following sections.

Proposition 3.2.1 *If $(f;g)$ is $\mathbb{P}$-open and g is mono, then f is also $\mathbb{P}$-open.*

Proof: To show that $f : A \to B$ is $\mathbb{P}$-open, we construct for any commuting square

$$\begin{tikzcd} O_1 \arrow[r, "p"] \arrow[d, "m"'] & A \arrow[d, "f"] \\ O_2 \arrow[r, "q"'] & B \end{tikzcd} \tag{3.2}$$

(with $m : O_1 \to O_2$ in $\mathbb{P}$) a diagonal morphism $r : O_2 \to A$ making both triangles commute. A good candidate is a diagonal morphism r in the commuting diagram:

$$\begin{tikzcd} O_1 \arrow[r, "p"] \arrow[d, "m"'] & A \arrow[d, "f;g"] \\ O_2 \arrow[r, "q;g"'] \arrow[ur, dotted, "r"] & B \end{tikzcd}$$

As $(f;g) : A \to C$ is open such a morphism exists and makes: $m;r = p$ and $r;f;g = q;g$; from g being mono it follows now that $r;f = q$ and so indeed f is open. $\square$

Recall from Section 2.2.1 in the previous chapter that if $\mathbb{M}$ is *initialized* then for any $\mathbb{P}$-open morphism $f : A \to B$ and a computation $q : O \to B$ there exists a computation $p : O \to A$ such that $p; f = q$. The most usual situation is when $\mathbb{M}$ has an initial object $\mathcal{I}$ being also initial in $\mathbb{P}$ – in such a case, the unique morphism from $\mathcal{I}$ can be thought of as the empty (initial) computation.

Proposition 3.2.2 *Assume that $\mathbb{M}$ is initialized. If g and $(g; f)$ are $\mathbb{P}$-open, then f is $\mathbb{P}$-open as well.*

Proof: We are going to construct a diagonal morphism for any commuting diagram (3.2). As g is open and $\mathbb{M}$ is initialized, there exists a morphism $s_1 : O_1 \to C$ (let C be the domain of g) such that $s_1; g = p$. Now consider a commuting square:

$$\begin{tikzcd}
O_1 \arrow[r, "s_1"] \arrow[dd, "m"'] & C \arrow[d, "g"] \\
 & A \arrow[d, "f"] \\
O_2 \arrow[r, "q"'] \arrow[uur, dotted, "s_2"] \arrow[ur, dashed, "r"'] & B
\end{tikzcd}$$

and take any diagonal morphism $s_2 : O_2 \to C$ such that $m; s_2 = s_1$ and $s_2; g; f = q$. We will show that $r = (s_2; g)$ is a diagonal morphism of diagram (3.2). We have $r; f = s_2; g; f = q$. Moreover, $m; r = m; s_2; g = s_1; g = p$. $\Box$

Recall from Section 2.3 the universal properties of quotients and subobjects that determine uniquely, for certain morphisms g, morphisms $\bar{g}$ or $\hat{g}$.

Proposition 3.2.3 *For any $\mathbb{P}$-open morphism $f : A \to B$, the quotient projection $\pi_{A/\ker(|f|)}$ is open. Moreover, when $\mathbb{M}$ is initialized, then for any $\mathbb{P}$-open morphism $g : A \to C$ meeting $\ker(|f|) \subseteq \ker(|g|)$, the unique morphism $\bar{g}$ satisfying $g = \pi_{A/\ker(|f|)}; \bar{g}$ is also $\mathbb{P}$-open.*

Proof: The quotient projection $\pi_{A/\ker(|f|)}$ is open by Proposition 3.2 as $|\bar{f}|$ is mono. The openness of $\bar{g}$ follows now from Proposition 3.2.2. $\Box$

Let $\mathbb{COMP}_A \subseteq |A|$ stand for the union of images of all computations in A:

$$\mathbb{COMP}_A := \bigcup_{O \xrightarrow{p} A,\ O \in |\mathbb{P}|} |p|(|O|). \tag{3.3}$$

Proposition 3.2.4 *For any $A \in |\mathbb{M}|$ and any full subobject $\iota_{B \hookrightarrow A} : B \to A$ including $\mathbb{COMP}_A$ (i.e., $\mathbb{COMP}_A \subseteq |B|$), the inclusion $\iota_{B \hookrightarrow A} : B \to A$ is $\mathbb{P}$-open. Moreover, for any open $g : C \to A$ such that $|g|(|C|) \subseteq |B|$, the unique morphism $\hat{g}$ satisfying $\hat{g}; \iota_{B \hookrightarrow A} = g$ is also $\mathbb{P}$-open.*

Proof: For any computation $q : O \to A$ (O in $\mathbb{P}$), $|q|(|O|) \subseteq |B|$, so there exists a computation $\hat{q} : O \to B$ such that $\hat{q}; \iota_{B \hookrightarrow A} = q$, as B is a full subobject. Now for any commuting diagram $p; \iota_{B \hookrightarrow A} = m; q$, where $p : U \to B$ (U in $\mathbb{P}$), $m : U \to O$ in $\mathbb{P}$, we will show that $\hat{q}$ is a diagonal morphism. We already have $\hat{q}; \iota_{B \hookrightarrow A} = q$; moreover $m; \hat{q}; \iota_{B \hookrightarrow A} = m; q = p; \iota_{B \hookrightarrow A}$, so (as $\iota_{B \hookrightarrow A}$ is mono) we obtain $m; \hat{q} = p$.

Openness of $\hat{g}$ follows immediately by Proposition 3.2.1. $\Box$

By the above two propositions, quotients by kernels of open maps as well as subobjects including $\mathbb{COMP}_A$ are also quotients and subobjects in the subcategory of open maps. These quotients and subobjects will play an important role in the following sections, hence for convenience we introduce the following naming convention:

open congruence on A:	a kernel of an open map from A,
open quotient:	a quotient by an open congruence,
open subobject of A:	a full subobject including $\mathbb{COMP}_A$.

Notice that by Proposition 3.2.3 we know that a congruence is open if and only if it has an open quotient projection. Moreover, when $\mathbb{M}$ is initialized, which is usually the case, we easily notice that a (full) subobject is open if and only if its inclusion is open. This leads to the following conclusion (for initialized $\mathbb{M}$):

open congruence	=	a congruence with an open projection,
open subobject	=	a full subobject with open inclusion.

Note the analogy to the observational congruences and subobjects in Section 2.3.

In the following sections we only use quotients by open congruences. Hence, instead of the fourth assumption in the beginning of this section, it is sufficient to assume that

- there exist in $\mathbb{M}$ quotients by all open congruences and moreover the quotient projections are surjective.

This will be crucial in Section 3.5.

3.3 Factorization by the greatest open congruences

In this section we address the following problem: for an arbitrary category $\mathbb{M}$, under the same assumptions as in the previous section, and for its arbitrary subcategory $\mathbb{P}$,

> *Find an indistinguishability structure (cf. Section 2.3), i.e., a family of partial congruences $\simeq = \{\simeq_A\}_{A\in|\mathbb{M}|}$, inducing the same abstract equivalence as $\mathbb{P}$-open morphisms, that is, factorizing $\mathbb{P}$-bisimilarity. That is, we want $\equiv_{\simeq} = \sim_{\mathbb{P}}$, or equivalently*
>
> $$A/_{\simeq}\cong B/_{\simeq} \iff A\sim_{\mathbb{P}}B, \quad \text{for all } A, B \in |\mathbb{M}|.$$

In the following we prove that under Assumption 3.3.3 the family of the greatest open congruences on the smallest open subobjects is the solution to this problem (Theorem 3.3.4). The domain subobjects will be defined as the full subobjects $\langle\mathbb{COMP}_A\rangle_A$ generated by the set-theoretic sum of images of all computations. We denote the subobject $\langle\mathbb{COMP}_A\rangle_A$ in short by $\widetilde{A}$, as it will be often used in the sequel.

Why do we choose the smallest open subobjects and the greatest open congruences? Restricting to open maps is suggested by Propositions 3.2.3 and 3.2.4, stating that the subcategory of open maps inherits all the relevant algebraic structure of $\mathbb{M}$, i.e., open subobjects and surjective quotients. The choice of *the smallest* open subobjects follows the intuition that a domain of an indistinguishability relation (i.e., the observable part of an object) should be equal to its part "visible" via computations. An argument for considering only open quotients is our aim to have $A/\simeq$ bisimilar to A. Another justification for *the smallest* subobjects and *the greatest* quotients is given by the observation that the equivalence induced by any of indistinguishability structures under consideration is finer than bisimilarity:

Lemma 3.3.1 *Let $\approx$ be an indistinguishability structure in $\mathbb{M}$ consisting, for each object A, of an open congruence $\approx_A$ on an open subobject of A. Then $\equiv_\approx \subseteq \sim_\mathbb{P}$.*

Proof: Assume that $A \equiv_\approx B$, that is $A/_\approx$ and $B/_\approx$ are isomorphic. Then we obtain a span of $\mathbb{P}$-open morphisms $f' = f; \iota_{A_\approx \hookrightarrow A}$ and $g' = g; \iota_{B_\approx \hookrightarrow B}$ from a pullback of the pair of quotient projections:

$$\begin{tikzcd}
 & & C \arrow[dl, "f"'] \arrow[dr, "g"] & & \\
A & A_\approx \arrow[l, hook'] \arrow[dr, "\pi"'] & & B_\approx \arrow[r, hook] \arrow[dl, "\pi"] & B \\
 & & A/_\approx \cong B/_\approx & &
\end{tikzcd}$$

Both f', g' are $\mathbb{P}$-open since f and g are open (because pullbacks of open morphisms are open – proof of this fact can be found in [JNW93]) and the subobject inclusions $\iota_{A_\approx \hookrightarrow A}$ and $\iota_{B_\approx \hookrightarrow B}$ are open maps (by Proposition 3.2.4). □

Thus, choosing the smallest open subobjects and the greatest open congruences we aim, intuitively, at staying as close as possible to the bisimilarity.

In addition, note that whenever $\equiv_\approx$ and $\sim_\mathbb{P}$ coincide, the indistinguishability structure $\approx$ is especially well-behaved:

Proposition 3.3.2 *If $\equiv_\approx = \sim_\mathbb{P}$, then all quotients $A/\approx$ are* weakly fully abstract, *i.e.,*

$$(A/\approx)/\approx \cong A/\approx.$$

Proof: Let A be an arbitrary object of $\mathbb{M}$. Since the quotient projection $\pi_{A_\approx/\approx_A}$ as well as the subobject inclusion $\iota_{A_\approx \hookrightarrow A}$ are $\mathbb{P}$-open, they form a span of open maps relating $A/\approx$ and A, i.e., we have $A/\approx \sim_\mathbb{P} A$. This implies immediately $A/\approx \equiv_\approx A$, i.e., $(A/\approx)/\approx \cong A/\approx$. □

Now we are going to introduce formally the indistinguishability structure $\doteq$ to factorize $\mathbb{P}$-bisimilarity. By $\mathbb{CONGR}_A$ we denote, for any object A, the set of all open congruences on A:

$$\mathbb{CONGR}_A := \{\ker(|f|) : A \xrightarrow{f} B \text{ is } \mathbb{P}\text{-open}\}.$$

We need in the sequel the following assumption, justified by some important examples, like transition systems (cf. Section 3.3.1) or presheaves (Section 3.6):

Assumption 3.3.3 *For each object A in $\mathbb{M}$, there exists the greatest (w.r.t. inclusion) open congruence on $\widetilde{A}$, i.e., the poset $(\mathbb{CONGR}_{\widetilde{A}}, \subseteq)$ has the greatest element.*

For any object A in $\mathbb{M}$, let $\doteq_A$ be a partial congruence on A such that its domain subobject $A_\doteq$ is $\widetilde{A}$ and $\doteq_A$ is the greatest congruence on $A_\doteq$. Consider then the indistinguishability structure $\doteq = \{\doteq_A\}_{A \in \mathbb{M}}$. Note the analogy to the indistinguishability structure for observational equivalence in Section 2.3.

Before stating the main result (Theorem 3.3.4), we need some more definitions. First, we say that *computations create subobjects* in $\mathbb{M}$ if, for any A in $\mathbb{M}$,

$$|\widetilde{A}| = \mathbb{COMP}_A$$

(each element of a domain subobject is in image of some computation). Second, we say that the concretization functor $|\_|$ *lifts isomorphisms of domain subobjects* if for any open morphism $f : \widetilde{A} \to \widetilde{B}$, $|f|$ is iso (bijection) implies f is iso. This means, roughly speaking that open maps are powerful enough to transport and reflect the whole structure of observable parts of objects. In the following theorem we formulate conditions to guarantee a factorization of $\mathbb{P}$-bisimilarity. All these requirements are verified in Sections 3.3.1, 3.5 and 3.6 to be satisfied by some standard frameworks.

Theorem 3.3.4 (Factorization) *Assume that*

- $|\_|$ *lifts isomorphisms of domain subobjects,*
- *computations create subobjects in* $\mathbb{M}$*,*
- $\mathbb{M}$ *is initialized.*

Then $\equiv_{\simeq} = \sim_{\mathbb{P}}$*.*

Before embarking on the proof, we formulate and prove two lemmas which will be needed afterwards. In the first lemma we exhaustively check all the four possibilities of consecutively applying one of two operations: taking a domain subobject $(\_)_{\simeq}$ and quotient $\_/\simeq$ by the partial congruence. The lemma states, roughly speaking, that all four collapse, i.e., are equivalent to a single application of one of the operations, up to isomorphism:

Lemma 3.3.5 *If* $\mathbb{M}$ *is initialized and computations create subobjects, then for any object* A *in* $\mathbb{M}$ *it holds:*

1. $(A_{\simeq})_{\simeq} \cong A_{\simeq}$,
2. $(A_{\simeq})/\simeq \cong A/\simeq$,
3. $(A/\simeq)_{\simeq} \cong A/\simeq$,
4. $(A/\simeq)/\simeq \cong A/\simeq$.

Moreover, isomorphisms in 1 and 3 are exact, i.e., their concretizations are identities.

Proof: First, observe that it is sufficient to show 1 and 3; 2 is a direct conclusion of 1 and 4 follows from 3, since the surjective quotient $A/\simeq$ of $A_{\simeq}$ by the greatest open congruence can have no open congruences but identity.

For 1 it is sufficient to show that

$$\mathbb{COMP}_{A_{\simeq}} = |A_{\simeq}|. \tag{3.4}$$

Since the subobject inclusion $\iota_{A_{\simeq} \hookrightarrow A}$ is full, each computation in A factorizes through $A_{\simeq}$. From this observation we conclude

$$\mathbb{COMP}_{A_{\simeq}} = \mathbb{COMP}_A.$$

Now equality (3.4) follows immediately, since $\mathbb{COMP}_A = |A_{\simeq}|$, which holds due to the fact that computations create subobjects in $\mathbb{M}$.

For 3, observe that $A/\simeq$ is a surjective quotient of $A_{\simeq}$. Using (3.4), by surjectivity of the quotient projection we conclude $\mathbb{COMP}_{A/\simeq} = |A/\simeq|$, which implies 3. □

Second lemma is crucial for the proof of Theorem 3.3.4:

Lemma 3.3.6 *Under the assumptions of Theorem 3.3.4, any $\mathbb{P}$-open morphism $h : \widetilde{A} \to \widetilde{B}$ is a (surjective) quotient projection and consequently, $\widetilde{B}$ is a quotient of $\widetilde{A}$ by* $\ker(|h|)$.

Proof: Consider a quotient C of $\widetilde{A}$ by $\ker(|h|)$, with the quotient projection $\pi : \widetilde{A} \to C$.

$$\begin{tikzcd} & C \arrow[d, "\bar{h}"] \\ \widetilde{A} \arrow[ur, "\pi"] \arrow[r, "h"'] & \widetilde{B} \end{tikzcd}$$

The unique morphism $\bar{h} : C \to \widetilde{B}$ making $\pi; \bar{h} = h$ is open by Proposition 3.2.3. We will show $\bar{h}$ to be iso. By the assumption that the concretization functor lifts isomorphisms of domain subobjects, it is enough to show that $|\bar{h}|$ is a bijection. Obviously, $|\bar{h}|$ is injective as $\ker(|\pi|) = \ker(|h|)$ and $|\pi|$ is surjective, hence it remains only to show $|\bar{h}|$ to be surjective.

We start by showing that $|h|$ is surjective. For any computation $q : O \to \widetilde{B}$, there exists a computation $p : O \to \widetilde{A}$ such that $p; h = q$, as h is open and $\mathbb{M}$ is initialized. Hence $|q| = |p; h|$ and $|q|(|O|) = |p; h|(|O|) \subseteq |h|(|\widetilde{A}|)$. As q is an arbitrary computation in $\widetilde{B}$, we conclude $\mathbb{COMP}_{\widetilde{B}} \subseteq |h|(|\widetilde{A}|)$. But $\mathbb{COMP}_{\widetilde{B}} = |\widetilde{\widetilde{B}}|$ because computations create subobjects in $\mathbb{M}$, and $|\widetilde{\widetilde{B}}| = |\widetilde{B}|$ by Lemma 3.3.5(1). From this we conclude $|\widetilde{B}| \subseteq |h|(|\widetilde{A}|)$, i.e., $|h|$ is surjective. This immediately implies $|\bar{h}|$ to be surjective as well, since $|\pi|; |\bar{h}| = |h|$. $\square$

Proof of Theorem 3.3.4: Inclusion $\equiv_{\simeq} \subseteq \sim_{\mathbb{P}}$ follows directly from Lemma 3.3.1, hence we only need to show $\sim_{\mathbb{P}} \subseteq \equiv_{\simeq}$. Consider two objects A, B in $\mathbb{M}$ such that $A \sim_{\mathbb{P}} B$, i.e., A and B are related by a span

$$\begin{tikzcd} & C \arrow[dl, "f"'] \arrow[dr, "g"] & \\ A & & B \end{tikzcd}$$

of open morphisms. The subobject inclusion $\iota_{C_{\simeq} \hookrightarrow C} : C_{\simeq} \to C$ is open by Proposition 3.2.4.

First we aim at showing that f induces an open map between domain subobjects $C_{\simeq} \to A_{\simeq}$. Since for any computation $p : O \to C_{\simeq}$, the morphism $(p; \iota_{C_{\simeq} \hookrightarrow C}; f)$ is a computation in A, we obtain trivially $|\iota_{C_{\simeq} \hookrightarrow C}; f|(|C_{\simeq}|) = |\iota_{C_{\simeq} \hookrightarrow C}; f|(\mathbb{COMP}_C) \subseteq \mathbb{COMP}_A$. The subobject $A_{\simeq}$ is full, thus there exists the unique morphism $g = \widehat{(\iota_{C_{\simeq} \hookrightarrow C}; f)} : C_{\simeq} \to A_{\simeq}$ making the square in the following diagram commutative (i.e., $g; \iota_{A_{\simeq} \hookrightarrow A} = \iota_{C_{\simeq} \hookrightarrow C}; f$):

$$\begin{tikzcd} & C_{\simeq} \arrow[r, hook] \arrow[dl, "\pi_{C/\simeq}"'] \arrow[d, "g"] & C \arrow[d, "f"] \\ C/\simeq & A_{\simeq} \arrow[l, "h"] \arrow[r, hook] & A \end{tikzcd}$$

By Proposition 3.2.4 we know that g is open (since both f and the subobject inclusion $\iota_{C_{\simeq} \hookrightarrow C} : C_{\simeq} \to C$ are open), hence by Lemma 3.3.6, $A_{\simeq}$ is a quotient of $C_{\simeq}$ and g is a quotient projection. Since $\simeq_C$ is the greatest congruence on $C_{\simeq}$, there is the unique morphism $h = \overline{\pi_{C/\simeq}} : A_{\simeq} \to C/\simeq$ satisfying $g; h = \pi_{C/\simeq}$, open by Proposition 3.2.3.

By Lemma 3.3.5(3), $C/\simeq$ is necessarily its own domain subobject $(C/\simeq)_\simeq$ (up to exact isomorphism), hence by Lemma 3.3.6, h is also a quotient projection and $C/\simeq$ is a quotient of $A_\simeq$. From $\simeq_C$ being the greatest among open congruences on $C_\simeq$ it follows that $C/\simeq$ has no open congruences but identity, i.e., $C/\simeq$ is *fully abstract* (since otherwise we would construct an open congruence on $C_\simeq$ strictly containing $\simeq_C$). But this implies that $\ker(|h|)$ is a maximal open congruence on $A_\simeq$, hence the greatest one. This gives $A_\simeq/\simeq_A \cong C_\simeq/\simeq_C$.

Analogously we obtain the quotient projection $h' : B_\simeq \to B_\simeq/\simeq_B \cong C_\simeq/\simeq_C$; thus we conclude that $A_\simeq/\simeq_A \cong B_\simeq/\simeq_B$, which completes the proof. □

3.3.1 Strong bisimilarity

We apply here Theorem 3.3.4 to strong bisimilarity between transition systems, i.e., $Bran_L$-bisimilarity in category TS_L. First, we should verify that this category satisfies all assumptions needed there. Category TS_L has pullbacks [NW95]. We will describe full subobjects and quotients, omitting easy verification that they satisfy the appropriate requirements. For a transition system $T = (S, i, t)$, and a subset $X \subseteq S$ of states, we define $\langle X \rangle_T$ as $(X \cup \{i\}, i, t')$ with transition relation t' obtained by restriction of t to $(X \cup \{i\}) \times (X \cup \{i\})$. For a morphism $f : T \to U$, we define the quotient of T by $\sim := \ker(|f|)$ as $(T/\sim, [i]_\sim, t')$, where a transition $[s]_\sim \xrightarrow{a} [r]_\sim$ is in t' if there are some states s', r', such that $s \sim s'$, $r \sim r'$, $s' \xrightarrow{a} r'$ in T. The canonical transformation $s \mapsto [s]_\sim$ is a (surjective) quotient projection.

For any transition system T, $\mathbb{COMP}_T$ is the set of its reachable states, hence trivially computations create subobjects in TS_L. Moreover, TS_L is initialized, as it has an initial object $(\{i\}, i, \emptyset)$, which is also initial in $Bran_L$. To prove that the forgetful functor $|\_|$ restricted to zig-zag morphisms between reachable transition systems lifts isomorphisms, it suffices to show that f^{-1} is a morphism of transition systems, which follows easily from the zig-zag property of f. Moreover, Lemma 3.1.1 from Section 3.1 ensures the existence of the greatest open congruence on a reachable subobject of a transition system. Having verified all the requirements, we conclude (as announced earlier):

Corollary 3.3.7 *The strong bisimilarity is factorized by the greatest bisimulations.*

3.4 Maximal open congruences

A careful examination of the proof of Theorem 3.3.4 leads to an observation that the requirement of existence of the greatest open congruences is slightly too strong – a weaker, but still sufficient assumption is formulated in Theorem 3.4.1 below. It turns out to be also necessary, hence we obtain a characterization of open-map bisimilarities that are factorizable by open congruences. This result is applied in the following section to event structures, failing to have the greatest open congruences.

Let $\approx$ denote an arbitrary indistinguishability structure consisting exclusively of partial open congruences with $A_\approx = \widetilde{A}$, for all objects A. The consideration below are done under the assumption that the posets $\mathbb{CONGR}_{\widetilde{A}}$ are *bounded*, that is every increasing chain in $\mathbb{CONGR}_{\widetilde{A}}$ has an upper bound (we will say in short that *congruences are bounded* in $\mathbb{M}$). Note that this guarantees that for each open congruence, there exists a maximal open congruence including it. We will say that $\mathbb{M}$ *is normalized by* $\approx$ if quotients of $A_\approx$ by all maximal open congruences are isomorphic to $A/_\approx$ (for all A in $\mathbb{M}$). Equivalently, this can be divided into two separate conditions:

- quotients of A by all maximal congruences are isomorphic,
- $\approx_A$ is a maximal congruence.

The first condition, intuitively, amounts to saying that there is essentially only one way to behaviourally quotient any object.

Theorem 3.4.1 *Assume that congruences are bounded in $\mathbb{M}$ and moreover, similarly as in Theorem 3.3.4, that $|\_|$ lifts isomorphisms of domain subobjects, computations create subobjects in $\mathbb{M}$ and $\mathbb{M}$ is initialized. Then the following conditions are equivalent:*

1. $\equiv_{\approx} = \sim_{\mathbb{P}}$,
2. $\mathbb{M}$ *is normalized by* $\approx$.

Proof: $1 \Leftarrow 2$: A slight adaptation of the proof of Theorem 3.3.4. Similarly as there, we construct the quotient projection $g : C_{\approx} \to A_{\approx}$ by application of Lemma 3.3.6. $C/\!\approx$ need not to be a quotient of $A_{\approx}$ – instead, take a maximal open congruence $\sim_{\max}$ on $C_{\approx}$ including $\ker(|g|)$.

$$\begin{tikzcd} & C_{\approx} \arrow[dl, "\pi_{C/\sim_{\max}}"'] \arrow[d, "g"] \\ C/{\sim_{\max}} & A_{\doteq} \arrow[l, "h"] \end{tikzcd}$$

By an argument similar to that in Theorem 3.3.4 we will construct an open quotient projection $h : A_{\approx} \to C/{\sim_{\max}}$. Since $\sim_{\max}$ is maximal, $\ker(|h|)$ is maximal too. Applying twice the fact that $\mathbb{M}$ is normalized by $\approx$, we obtain $C/\!\approx \,\cong C/{\sim_{\max}} \cong A/\ker(|h|) \cong A/\!\approx$, which, after the analogous argument for B, implies $A/\!\approx\, \cong B/\!\approx$.

$1 \Rightarrow 2$: Let $A_{\max}$ be a quotient of $A_{\approx}$ by any of maximal open congruences. Since $A \sim_{\mathbb{P}} A_{\max}$ (as the quotient projection $\pi : A \to A_{\max}$ is open by Proposition 3.2.3), we conclude

$$A_{\max}/_{\approx} \cong A/_{\approx}. \tag{3.5}$$

One can easily check that $(A_{\max})_{\approx} = A_{\max}$, analogously as in Lemma 3.3.5(3) (in general, each object being an open quotient of a domain subobject is a domain subobject itself). From maximality of the kernel of π (in $\mathbb{CONGR}_{A_{\approx}}$) we conclude that $\approx_{A_{max}}$ is identity on $A_{\max}$, hence $A_{\max} \cong A_{\max}/_{\approx}$; now by (3.5) we get $A_{\max} \cong A/_{\approx}$, which completes the proof. $\square$

When all $\approx_A$ are maximal, we can strengthen Proposition 3.3.2:

Proposition 3.4.2 *Assume that computations create subobjects. If $\approx$ consists exclusively of maximal open congruences and $\equiv_{\approx} = \sim_{\mathbb{P}}$, then all quotients $A/\!\approx$ are* fully abstract, *i.e., $\approx_{A/\approx}$ is an identity, for every object A of $\mathbb{M}$.*

Proof: Let A be an arbitrary object of $\mathbb{M}$. First, observe that $\approx_{A/\approx}$ is total, and moreover $\mathbb{COMP}_{A/\approx} = |A/\!\approx|$, which is guaranteed by the assumption that computations create subobjects and by surjectivity of the quotient projection $\pi_{A/\approx} : A_{\approx} \to A/\!\approx$.

Now, assume that $\approx_{A/\approx}$ is not identity. By composing the quotient projections for $\approx_A$ and $\approx_{A/\approx}$, we obtain an open congruence $\ker(|\pi_{A_{\approx}/\approx}; \pi_{(A/\approx)/\approx}|)$ on $A_{\approx}$ strictly coarser than $\approx_A$, which contradicts the fact that $\approx_A$ is maximal in $\mathbb{CONGR}_{A_{\approx}}$. $\square$

Consequently, by Theorem 3.4.1 and Proposition 3.4.2 we conclude that under assumptions from Theorem 3.4.1, $\equiv_{\approx} = \sim_{\mathbb{P}}$ implies that all quotients $A/\!\approx$ are fully abstract.

3.5 Shp-bisimilarity

The following definitions, cited from [JNW93] and [NW95], are specialized for a fixed set of labels. A *labelled event structure* (over L) is a structure $E = \langle |E|, \leq_E, Con_E, l_E \rangle$ consisting of a set $|E|$ of events partially ordered by the *causal dependency relation* $\leq_E$, a family $Con_E \subseteq \mathcal{P}^{fin}(|E|)$ of finite subsets of events (the *consistency relation*), and a labelling function $l_E : |E| \to L$ satisfying:

- $\lceil e \rceil = \{e' : e' \leq_E e\}$ is finite, for every $e \in |E|$,
- $\{e\} \in Con_E$,
- $Y \subseteq X \in Con_E \Rightarrow Y \in Con_E$,
- $X \in Con_E \wedge e \leq_E e' \in X \Rightarrow X \cup \{e\} \in Con_E$.

Two events e_1 and e_2 are called *concurrent*, denoted by $e_1\ co_E\ e_2$, when they are consistent, but causally not related: $\{e_1, e_2\} \in Con_E\ \wedge\ e_1 \not\leq_E e_2\ \wedge\ e_2 \not\leq_E e_1$.

A *configuration* of E, representing a state of computation, is any subset C of events being:

- *downward closed*: $e' \in C \wedge e \leq_E e' \Rightarrow e \in C$,
- *consistent*: $\forall X \subseteq^{fin} C \,.\, X \in Con_E$.

Each configuration C of E gives raise to an event structure, with consistency relation containing all finite subsets of C and causality and labelling inherited from E. In the sequel, an event structure whose consistency relation contains all finite subsets of events is called a *pomset*. When convenient, we identify a configuration with a pomset it gives raise to.

A morphism of event structures $h : E \to F$ is a function $h : |E| \to |F|$ satisfying:

- $l_F(h(e)) = l_E(e)$, for any $e \in |E|$ (preservation of labels),
- if C is a configuration of E then $h(C)$ is a configuration of F and h restricted to C is 1-1.

(Equivalently, in the last point one could only consider *finite* configurations C.) Event structures with their morphisms form a category ES_L. For configurations C, C' of E and any $a \in L$, by $C \xrightarrow{a} C'$ we mean that for some $e \notin C$, $C' = C \cup \{e\}$ and $l_E(e) = a$. Note that morphisms of event structures preserve transitions, namely given a morphism $h : E \to F$, $C \xrightarrow{a} C'$ in E implies $h(C) \xrightarrow{a} h(C')$ in F.

Definition 3.5.1 *A* history preserving bisimulation *(hp-bisimulation in short) between event structures E_1, E_2 is a set H of triples (C_1, ξ, C_2), where C_1 is a finite*[1] *configuration of E_1, C_2 is a finite configuration of E_2 and ξ is an isomorphism of pomsets*[2] *between them, such that $(\emptyset, \emptyset, \emptyset) \in H$, and whenever $(C_1, \xi, C_2) \in H$,*

- *if $C_1 \xrightarrow{a} C_1'$ in E_1 then $C_2 \xrightarrow{a} C_2'$ in E_2 and $(C_1', \xi', C_2') \in H$, with $\xi \subseteq \xi'$, for some C_2', ξ',*

[1] In [JNW93] infinite configurations are also allowed in bisimulations. Following [vGG89], we consider here finite configurations only, as this has no influence on the bisimilarity relation induced.

[2] i.e., a function which preserves and reflects causality and labelling.

- *if* $C_2 \xrightarrow{a} C_2'$ *in* E_2 *then* $C_1 \xrightarrow{a} C_1'$ *in* E_1 *and* $(C_1', \xi', C_2') \in H$, *with* $\xi \subseteq \xi'$, *for some* C_1', ξ'.

(By C_1' *and* C_2' *we denote finite configurations of* E_1 *and* E_2*, respectively.) An hp-bisimulation* H *is* strong *(shp-bisimulation in short) when it further satisfies*

- $(C_1, \xi, C_2) \in H$ *and* $C_1' \subseteq C_1$ *implies* $(C_1', \xi', C_2') \in H$, *for some* $\xi' \subseteq \xi$ *and* $C_2' \subseteq C_2$,
- $(C_1, \xi, C_2) \in H$ *and* $C_2' \subseteq C_2$ *implies* $(C_1', \xi', C_2') \in H$, *for some* $\xi' \subseteq \xi$ *and* $C_1' \subseteq C_1$.

Event structures E_1, E_2 *are* shp-bisimilar *if there exists an shp-bisimulation between them.*

History preserving bisimilarity was invented by many authors independently, among others in [RT88] and [vGG89]. The shp-bisimilarity has been first introduced in [Bed91] (called there *hereditary* history preserving bisimilarity) and rediscovered independently in [JNW93]. To obtain an open-map characterization of shp-bisimilarity we need some branching structure in observation objects: shp-bisimilarity coincides with Pom_L-bisimilarity [JNW93], where Pom_L is the full subcategory of ES_L consisting of finite pomsets. A morphism $f : E \to F$ is Pom_L-open iff

1. f satisfies the "zig-zag" property: for any finite configuration C of E, if $f(C) \xrightarrow{a} C'$ in F then $C \xrightarrow{a} C''$ in E and $f(C'') = C'$, for some configuration C'' of E.
2. f reflects concurrency: for all different events $e_1, e_2 \in |E|$, such that $\{e_1, e_2\} \in Con_E$, $f(e_1)\ co_F\ f(e_2)$ implies $e_1\ co_E\ e_2$.

Note that the implication opposite to 2 holds for every (not necessarily open) morphism of event structures: $e_1\ co_E\ e_2$ implies $f(e_1)\ co_F\ f(e_2)$.

3.5.1 Factorization

Our aim is now to apply the results from the previous section to shp-bisimilarity in the category ES_L of event structures (over a fixed, throughout this section, labelling set L). We feel free to omit prefixes Pom_L- and shp-, obvious in the present context. As the notation $|E|$ suggests, category of event structures can be equipped with a concretization functor $|\_|$, taking an event structure E to its set $|E|$ of events, and a morphism f to the underlying function $|f|$. Since morphisms of event structures are functions, we write sometimes f instead of $|f|$, if this does not lead to confusion.

First we should prove that the category of event structures satisfies all assumptions needed in Section 3.4. Category ES_L has pullbacks [JNW93] and the concrete category $(ES_L, |\_|)$ has full subobjects – the set of events of the full subobject of an event structure E generated by $X \subseteq |E|$ is the (downward) closure of X under the causal dependency, while the labelling, causal dependency and consistency relation are inherited from E. However, we will not need subobjects in the sequel, since $\mathbb{COMP}_E$ is the set $|E|$ of all events of E, hence $\widetilde{E} = E$ (and obviously computations create subobjects in ES_L). The category of event structures is initialized, which follows, similarly as for transition systems, from the existence of the initial event structure, carrying empty set of events. The forgetful functor $|\_|$ lifts isomorphisms of domain subobjects: for any bijective morphism f in ES_L, its openness implies that f^{-1} is also a morphism of event structures.

Instead of showing event structures to have surjective quotients, we will only show that quotients by open congruences exist and are surjective – this is sufficient for Theorem 3.4.1 to hold, by the remark at the end of Section 3.4. Surprisingly, the following proposition holds:

Proposition 3.5.2 *Every open morphism of event structures is a surjective quotient projection.*

Proof: First, observe that each open morphism is surjective, due to the zig-zag condition. Let E, F be a domain and a codomain of an open morphism f. We are to show that for an arbitrary morphism $g : E \to G$ satisfying $\ker(g) \supseteq \ker(f)$, the unique function $\bar{g}$ defined by $\bar{g}(f(e)) = g(e)$:

$$\begin{tikzcd} & G \\ E \arrow[ur, "g"] \arrow[r, "f"'] & F \arrow[u, "\bar{g}"'] \end{tikzcd}$$

is a morphism of event structures. But this follows immediately from the fact that an open map restricted to a configuration is an isomorphism of pomsets, cf. [JNW93], Proposition 8. Let C' be a finite configuration in F; by the zig-zag property of f we conclude that there exists a configuration C in E with $f(C) = C'$. By the fact cited above, C and C' are pomset-isomorphic; moreover we have $g(C) = \bar{g}(C')$. All this implies that $\bar{g}(C')$ is a configuration and $\bar{g}$ restricted to C' is 1-1. $\square$

Observe that the kernel $\sim$ of any morphism *respects consistency*, i.e., it does not identify consistent events: $e_1 {\sim} e_2 \wedge e_1 \neq e_2 \Rightarrow \{e_1, e_2\} \notin Con_E$. Moreover, it is *compatible with labels*, in a sense that it contains only pairs of equally labelled events: $e_1 {\sim} e_2 \Rightarrow l_E(e_1) = l_E(e_2)$. Furthermore, each equivalence $\sim$ on the set of events of any event structure, satisfying these two conditions, is a congruence (i.e., the kernel of a morphism).

A surjective quotient by an arbitrary congruence does not necessarily exist in general – as a counterexample consider an event structure consisting of four events $\{e_1, e_2, f_1, f_2\}$ labelled identically, whose consistency relation consists only of subsets of $\{e_i, f_i\}$, for $i = 1, 2$, and causality is given by $e_1 \leq f_1, e_2 \leq f_2$. Let $\sim$ be a congruence identifying f_1 with f_2 (given, for example, as a kernel of a morphism to a tree-event pomset with discrete causality relation).

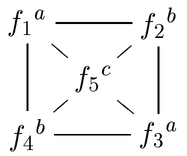
$$\begin{tikzcd} f_1 & f_2 & \longmapsto & {[f_1]_\sim} & = & {[f_2]_\sim} \\ e_1 \arrow[u, mapsto, "\leq"] & e_2 \arrow[u, mapsto, "\leq"] & \longmapsto & {[e_1]_\sim} \arrow[u, mapsto, "\leq"] & & {[e_2]_\sim} \arrow[u, mapsto, "\leq"] \end{tikzcd}$$

There is no quotient by $\sim$ in ES_L. To see this, note that in the potential quotient structure, it is not possible to have $[f_1]_\sim$ $(= [f_2]_\sim)$ causally dependent on $[e_1]_\sim$ $([e_2]_\sim)$, since the image of $\{e_2, f_2\}$ $(\{e_1, f_1\})$ should be a configuration. On the other hand, letting $[f_1]_\sim$ to be causally non-dependent on $[e_1]_\sim$ as well as on $[e_2]_\sim$ doesn't solve the problem, since a morphism $e_i \mapsto e, f_i \mapsto f$ to a two-events pomset $e \leq f$ would not factorize through the quotient projection then.

Event structures fail to have the greatest open congruences in general. A counterexample is given by an event structure with a set of events $\{e_1, \ldots, e_6, f_1, \ldots, f_6\}$ labelled by three labels a, b, c as indicated by superscripts, build up of two identical six-event components:

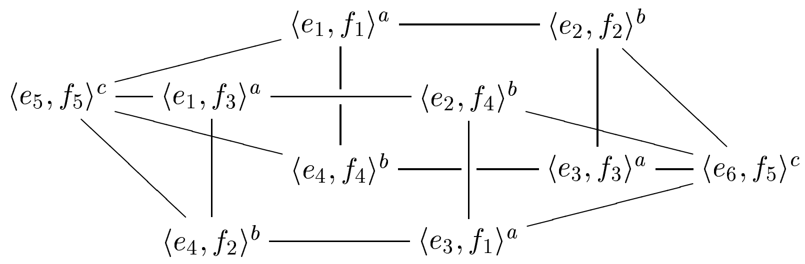
$$\begin{tikzcd} & {e_1}^a \arrow[r, dash] \arrow[dl, dash] \arrow[dd, dash] & {e_2}^b \arrow[dr, dash] \arrow[dd, dash] & & & & {f_1}^a \arrow[r, dash] \arrow[dl, dash] \arrow[dd, dash] & {f_2}^b \arrow[dr, dash] \arrow[dd, dash] & \\ {e_5}^c \arrow[dr, dash] & & & {e_6}^c \arrow[dl, dash] & & {f_5}^c \arrow[dr, dash] & & & {f_6}^c \arrow[dl, dash] \\ & {e_4}^b \arrow[r, dash] & {e_3}^a & & & & {f_4}^b \arrow[r, dash] & {f_3}^a & \end{tikzcd}$$

Its consistency relation contains all pairs of events connected in the picture by edges as well as all triples of events forming a triangle. The event structure has no *auto-concurrency* (no two concurrent events carry the same label) and has trivial, discrete causality relation. One can easily check that no open congruence can identify events from the same component and that there are two maximal open congruences: $\{e_1 \leftrightarrow f_1, e_2 \leftrightarrow f_2, e_3 \leftrightarrow f_3, e_4 \leftrightarrow f_4, e_5 \leftrightarrow f_5, e_6 \leftrightarrow f_6\}$ and $\{e_1 \leftrightarrow f_3, e_2 \leftrightarrow f_4, e_3 \leftrightarrow f_1, e_4 \leftrightarrow f_2, e_5 \leftrightarrow f_6, e_6 \leftrightarrow f_5\}$. In this case not everything is lost: one can check that both maximal open congruences give isomorphic quotients, which follows from auto-symmetry. Unfortunately, this observation cannot be generalized to all event structures (we correct here the result stated erroneously in Theorem 5.3 from [Las98a] for all event structures) – to see this, consider a pair of event structures: E_1, one component (full substructure) of the above structure consisting of only six events $\{e_1, \ldots, e_6\}$, and E_2, the following event structure:



(3.6)

obtained by identifying events f_5 and f_6 in the second component above, i.e., with consistency relation containing all pairs of events connected by an edge together with triples $\{f_1, f_4, f_5\}$ and $\{f_2, f_3, f_5\}$. These two structures are bisimilar, e.g. via a bisimulation containing all pairs of pomset-isomorphic configurations, but they are not related by a co-span of open quotient projections. Hence maximal open congruences of an event structure do not have to give isomorphic quotients. More explicitly, considering the following event structure E, which represents the bisimulation between E_1 and E_2 (consistency is determined, as previously, by all edges and triangles),



(3.7)

kernels of two projections to E_1 and E_2 are maximal open congruences giving non-isomorphic quotients E_1 and E_2. (This event structure represents the bisimulation between E_1 and E_2 in a sense that its configurations correspond bijectively to all triples (C_1, ξ, C_2) in the bisimulation; pomset isomorphisms ξ are not explicit here, but they are determined uniquely, since both event structures under consideration have no autoconcurrency.)

These considerations motivate the searching for a subcategory of event structures whose quotients by maximal open congruences are isomorphic. In the sequel we investigate the full subcategory of *simple* structures, i.e., those $E = \langle |E|, \leq_E, Con_E, l_E\rangle$ which satisfy the following two conditions:

- *coherency*: $\forall X \subseteq^{fin} |E|.(\forall \{e, e'\} \subseteq X.\{e, e'\} \in Con_E) \implies X \in Con_E$ (i.e., consistency is essentially a binary relation),
- *uniformity*: $\forall e, e', e'' \in |E|$, whenever $e\ co_E\ e'$ and $\{e, e''\} \in Con_E$, then $\{e', e''\} \in Con_E$.

While coherent structures were widely studied (see for instance [NW95]), second condition amounts to saying that concurrent events are related in the same way to other events. This requirements rules out some not well behaved structures, for example $e_1 \; co_{E_1} \; e_2$ and $\{e_1, e_5\} \in Con_{E_1}$ but $\{e_2, e_5\} \notin Con_{E_1}$. By the way, notice that E_1 is coherent while E_2 is not.

Restricting to only simple structures does not guarantee the greatest open congruences, which is caused by the possibility of auto-concurrency. In the rest of this section we show that simple event structures have isomorphic quotients by maximal open congruences (Theorem 3.5.8). Before proving this fact, we need to know that every open congruence is included in a maximal one, which we prove for the whole category ES_L in Lemma 3.5.5. The crucial role in the proof of this lemma is played by another Lemma 3.5.3, formulated below after some preparation.

By *the concretization* of an shp-bisimulation H between two event structures E_1 and E_2 we mean a relation $|H| \subseteq |E_1| \times |E_2|$ containing all pairs of events related by any of pomset-isomorphisms from H. Formally

$$(e_1, e_2) \in |H| \Leftrightarrow_{\text{def}} \exists \; (C_1, \xi, C_2) \in H \,.\, \xi(e_1) = e_2.$$

It is not possible, in general, to recover a bisimulation H from its concretization, as there can be more than one bisimulation with the same concretization. By a *lifting* $\text{Lift}(\sim)$ of a relation $\sim \subseteq |E_1| \times |E_2|$ to configurations we mean a relation consisting (similarly as shp-bisimulations) of triples (C_1, ξ, C_2), where C_1 and C_2 are finite configurations of E_1 and E_2, respectively, and ξ is an isomorphism of pomsets between them, defined by

$$(C_1, \xi, C_2) \in \text{Lift}(\sim) \;\; \Leftrightarrow_{\text{def}} \;\; \xi \subseteq \sim.$$

Intuitively, $\text{Lift}(\sim)$ is the greatest possible candidate to have a concretization $\sim$, but $\text{Lift}(\sim)$ is not necessarily a bisimulation, even if $\sim$ is a concretization of some bisimulation (for a counterexample, consider lifting of a concretization of any bisimulation in E_1 (from a counterexample above), which relates configuration $\{e_1\}$ with $\{e_3\}$ and $\{e_2\}$ with itself). We call a bisimulation H *recoverable* if $H = \text{Lift}(|H|)$. Any bisimulation satisfies $H \subseteq \text{Lift}(|H|)$ and $|\text{Lift}(|H|)| = |H|$, hence $\text{Lift}(\_)$ and $|\_|$, being monotonic, form a Galois connection between concretizations of bisimulations on one side and bisimulations together with liftings of their concretizations on the other side.

Now we are ready to state some correspondence between shp-bisimulations and open congruences. Unlike in the case of transition systems, where all bisimulation equivalences are congruences and an equivalence relation is an open congruence iff it is a bisimulation:

$$\textit{open congruence} \quad \Longleftrightarrow \quad \textit{bisimulation},$$

in the category of event structures we do not even have the following correspondence

$$\textit{open congruence} \quad \Longleftrightarrow \quad \textit{concretization of a bisimulation} \wedge \textit{congruence}.$$

A counterexample can be given by an equivalence relation on $|E_2|$ in (3.6) containing pairs (f_1, f_3) and (f_2, f_4), which is a concretization of a bisimulation and a congruence, but not an open congruence. Open congruences on an event structure correspond to recoverable bisimulations:

$$\textit{open congruence} \quad \Longleftrightarrow \quad \textit{concretization of a recoverable bisimulation} \wedge \textit{congruence}.$$

Since a congruence is any equivalence which is compatible with labels and respects consistency, this can be spelled out equivalently as follows:

Lemma 3.5.3 *Let* $\sim \subseteq |E| \times |E|$ *be an equivalence on the set of events of an event structure* E*;* $\sim$ *is an open congruence on* E *if and only if*

(i) $\sim$ *is a concretization of some recoverable bisimulation, and*

(ii) $\sim$ respects consistency*:* $e_1 {\sim} e_2 \wedge e_1 \neq e_2 \Rightarrow \{e_1, e_2\} \notin Con_E$*.*

Proof: "only if": Let $\sim$ be a kernel of an open map f, i.e., $\sim\, = f^{-1} \circ f$. Applying the definition of Lift(_) to $\sim$ we get:

$$(C_1, \xi, C_2) \in \mathrm{Lift}(\sim) \;\text{ iff }\; f(C_1) = f(C_2) \;\text{ and }\; \xi = (f \restriction_{C_2})^{-1} \circ (f \restriction_{C_1}). \tag{3.8}$$

Recall that f restricted to a configuration is a pomset isomorphism (which guarantees that ξ is a pomset isomorphism as well). This, together with the zig-zag property of f guarantees that Lift($\sim$) is an shp-bisimulation. Moreover, we easily conclude that $|\mathrm{Lift}(\sim)| = \sim$, since each pair of events $e_1 {\sim} e_2$ identified by $\sim$ is related by some ξ from (3.8), for example for $C_1 = \lceil e_1 \rceil$ and $C_2 = \lceil e_2 \rceil$. Finally, kernel of any morphism respects consistency.

"if": Let $\sim$ be any binary equivalence satisfying (i) and (ii). Since $\sim$ is a concretization of a recoverable bisimulation, Lift($\sim$) is a bisimulation and $|\mathrm{Lift}(\sim)| = \sim$. To show that $\sim$ is an open congruence, we will prove that the function $f : |E| \to |E|/{\sim} \,:\, e \mapsto [e]_\sim$ is an open morphism. But first, as the codomain of f we take an event structure $F = (|E|/{\sim}, \leq_F, Con_F, l_F)$ defined by:

$$\begin{aligned} \{[e_1]_\sim, \ldots, [e_k]_\sim\} \in Con_F \;\Leftrightarrow_{\mathrm{def}}\; & \exists\, e_1' {\sim} e_1, \ldots, e_k' {\sim} e_k . \{e_1', \ldots, e_k'\} \in Con_E; \\ [e_1]_\sim \leq_F [e_2]_\sim \;\Leftrightarrow_{\mathrm{def}}\; & \exists\, e_1' {\sim} e_1, e_2' {\sim} e_2 . e_1' \leq_E e_2'; \\ & l_F([e]_\sim) = l_E(e). \end{aligned}$$

F is an event structure – it can be easily checked due to the following fact:

> **Fact 3.5.4** *Whenever* $e_1 {\sim} e_2$ *and* $e_1' \leq_E e_1$*, there exists some* e_2' *with* $e_2' \leq_E e_2$ *and* $e_1' {\sim} e_2'$*.*
>
> **Proof:** Follows immediately since lifting of $\sim$ is a bisimulation and relates $\lceil e_1 \rceil$ with $\lceil e_2 \rceil$. □

Now we show that f is a morphism of event structures. It is 1-1 restricted to a configuration, since $\sim$ respects consistency. To prove that the image of a configuration is a configuration we only need to show downward closure. It is sufficient to check that whenever $[e_1]_\sim \leq_F [e_2]_\sim$, then for any e_2' such that $e_2' {\sim} e_2$, there exists some e_1' such that $e_1' \leq_E e_2'$ and $e_1' {\sim} e_1$. But this follows from Fact 3.5.4.

Finally, we show that f is open. The very definition of the quotient structure, together with Fact 3.5.4, implies that any configuration C of E is pomset-isomorphic to its image $f(C)$. From this we conclude that f reflects consistency.

For the zig-zag condition of f, choose an arbitrary finite configuration C in E such that $f(C) \xrightarrow{a} C'$ in F. We should find some C'' in E with $C \xrightarrow{a} C''$ and $f(C'') = C'$. Since Lift($\sim$) is a recoverable bisimulation, (3.8) also holds now. By the definition of Con_F, there is some $D' \subseteq |E|$ such that $D' \in Con_E$ and $f(D') = C'$. By the definition of $\leq_F$, D' is downward closed, i.e., D' is a configuration of E. Moreover, D' has a subconfiguration D isomorphic to C, such that $f(D) = f(C)$. Now, since Lift($\sim$) is a bisimulation, by (3.8) we conclude existence of an appropriate C'', which completes the proof. □

Lemma 3.5.5 *In the category of event structures, congruences are bounded.*

Proof: Consider an increasing chain of open congruences of an event structure, $\sim_1 \subseteq \sim_2 \subseteq \ldots$. By Lemma 3.5.3 the corresponding recoverable shp-bisimulations $H_1 := \mathrm{Lift}(\sim_1), H_2 := \mathrm{Lift}(\sim_2), \ldots$, satisfy $|H_i| = \sim_i$, for $i \geq 1$.

Let $H := \cup_{i\geq 1} H_i$ be the sum of all these bisimulations. Obviously H, union of a family of bisimulations, is itself a bisimulation. We will show that $\sim := |H|$ is an open congruence, being a bound of $\sim_1 \subseteq \sim_2 \subseteq \ldots$. To use Lemma 3.5.3, we are going to show first that H, union of a chain of recoverable bisimulations, is itself recoverable, i.e.,

$$H = \mathrm{Lift}(\sim). \tag{3.9}$$

Since $|\_|$ and $\mathrm{Lift}(\_)$ form a Galois connection, we have

$$\sim = |H| = |\cup_{i\geq 1} H_i| = \cup_{i\geq 1}|H_i| = \cup_{i\geq 1}\sim_i, \tag{3.10}$$

Assume (3.9) does not hold, i.e., there exists some triple (C, ξ, C') in $\mathrm{Lift}(\sim) \setminus H$. By the very definition of $\mathrm{Lift}(\_)$ and by (3.10) we conclude that there exists some j with $(C, \xi, C') \in \mathrm{Lift}(\sim_j)$; moreover, $(C, \xi, C') \notin H_j$, since $H_j \subseteq H$, from which we conclude that $\mathrm{Lift}(\sim_j) \setminus H_j$ is nonempty. This obviously contradicts H_j being recoverable, and thus proves (3.9).

From (3.10) it follows also that $\sim$ respects consistency, hence satisfies both conditions in in Lemma 3.5.3, i.e., is an open congruence, which completes the proof.

Note that $\sim$ is in fact the least upper bound of the chain $\sim_1 \subseteq \sim_2 \subseteq \ldots$; thus we have actually shown that open congruences are closed under unions of ω-chains. □

Having checked congruences to be bounded, we prove one more lemma, assuring that factorizability we are aiming at (see Corollary 3.5.9) takes place in the subcategory of simple structures:

Lemma 3.5.6 *Whenever E is simple, $E/\sim$ is also simple, for any open congruence $\sim$ on E. (Or equivalently: for an open map $f : E \to F$, whenever E is simple, F is simple too.)*

Proof: In the sequel we will need a lemma, following directly from the fact that $\sim$ is a concretization of some bisimulation:

> **Fact 3.5.7** *Whenever $e_1 \sim e_2$ and $\{e_1, e_1'\} \in Con_E$, there exists some e_2' with $\{e_2, e_2'\} \in Con_E$ and $e_1' \sim e_2'$.*

Coherency: Let $C_2 = \{[e_1]_\sim, \ldots, [e_k]_\sim\} \subseteq^{fin} |E/\sim|$ be any finite subsets of events of $E/_\sim$ with the property that each pair $[e_i]_\sim, [e_j]_\sim \in C_2$ is consistent. It follows immediately from openness of the quotient projection that for each such pair there exists a consistent pair e_i', e_j' in E with $e_i' \sim e_i$ and $e_j' \sim e_j$. Now, consecutive applications of Fact 3.5.7 together with uniformity of E guarantee the existence of a set $C_1 = \{e_1', \ldots, e_k'\}$ of events of E with $e_i' \sim e_i$, such that each pair of its elements is consistent. Since E is coherent, we conclude that C_1 is consistent, thus C_2, its image, is consistent too.

Uniformity: Assume $[e_1]_\sim \; co_{E/\sim} \; [e_2]_\sim$ and $\{[e_1]_\sim, [e_3]_\sim\} \in Cons_{E/\sim}$. By openness of the quotient projection and by Fact 3.5.7 we conclude that there exist $e_1' \sim e_1, e_2' \sim e_2, e_3' \sim e_3$ in E satisfying: $e_1' \; co \; e_2'$ and $\{e_1', e_3'\} \in Cons_E$. Since E is uniform we have $\{e_2', e_3'\} \in Cons_E$, which implies $\{[e_2]_\sim, [e_3]_\sim\} \in Cons_{E/\sim}$. □

We are able now to state the main result of this section: all simple quotients of an event structure by maximal open congruences are isomorphic.

Theorem 3.5.8 *Let $\sim$, $\sim'$ be two maximal open congruences of an event structure E, such that both quotients $E/{\sim}$, $E/{\sim'}$ are simple. Then the quotients are isomorphic.*

Proof: The proof will be based on an observation that both open quotient projections are split epi (i.e., have one-side inverse). (In fact all open maps with a simple codomain are split epi, since they are necessarily quotient projections.)

Let π, π' denote the quotient projections by $\sim$ and $\sim'$, respectively. We will construct an injective morphism $\iota : E/{\sim} \to E$ satisfying $\iota ; \pi = \mathrm{id}_{E/\sim}$. Even more, we will construct a full subobject (substructure) F of E isomorphic to $E/{\sim}$. This will be done by consecutive applications of the fact that π, being open, reflects transitions (zig-zag property). Let F_0 be the empty event structure, and let $\iota_0 : \emptyset \to F_0$ be the empty function. We define inductively the family $F_1, F_2, \ldots$ of full substructures of E, having the sets of events determined by the following rule:

$$\begin{array}{c} |F_{k+1}| := |F_k| \cup \iota_{k+1}(S_k), \text{ where} \\ S_k := \{e \in |E/{\sim}| : e \notin \pi(|F_k|) \wedge \lceil e \rceil \setminus \{e\} \subseteq \pi(|F_k|)\} \end{array} \tag{3.11}$$

(intuitively, S_k contains k-th "level" of $|E/{\sim}|$, i.e., all events of "height" k) where

$$\iota_{k+1} : \pi(|F_k|) \cup S_k \to |E|$$

is a function extending ι_k and returning, for each $e \in S_k \subseteq |E/{\sim}|$, some event $f \in |E|$ satisfying

$$\pi(\lceil f \rceil) = \lceil e \rceil \text{ and } \lceil f \rceil \setminus \{f\} \subseteq |F_k|, \tag{3.12}$$

and moreover ι_{k+1} satisfies the following condition, crucial for correctness of induction:

$$\iota_{k+1}(C) \text{ is a configuration, for each configuration } C \text{ in } \pi(|F_k|) \cup S_k. \tag{3.13}$$

Observe that some value $\iota_{k+1}(e)$ satisfying (3.12), for each $e \in S_k$, exists by the zig-zag property of π, although it is not determined uniquely, and in general Axiom of Choice is needed here. Moreover, requirement (3.13) can be fulfilled since $E/{\sim}$ is simple – this guarantees that (binary) consistency relation in S_k (coinciding in S_k with concurrency relation, since no two events from the same "level" can be related by causality) is an equivalence (it is transitive due to uniformity of $E/{\sim}$), hence value of ι_{k+1} can be chosen independently in each equivalence class.

In the sequel we will impose some additional requirements on ι_{k+1} – they are postponed, by now, for simplicity reason.

Recall that π restricted to any configuration $\lceil e \rceil$, for $e \in |E|$, is an isomorphism of pomsets (cf. [JNW93]). Hence the full substructure F, of E, being the union of all F_k, i.e., determined by $|F| = \cup_{k \geq 0} |F_k|$, is isomorphic to $E/{\sim}$. Indeed, taking $\iota = \cup_{k \geq 0} \iota_k : E/{\sim} \to E$, we obtain $\widehat{\iota} : E/{\sim} \to F$ as the unique morphism making $\widehat{\iota} ; \iota_{F \hookrightarrow E} = \iota$. $\widehat{\iota}$ is an isomorphism with inverse $\iota_{F \hookrightarrow E} ; \pi$.

$$\begin{tikzcd}
& F \arrow[d, hook] \arrow[dr, "\pi'\restriction_F"] & \\
& E \arrow[dl, bend left, "\pi"] \arrow[dr, "\pi'"] & \pi'(F) \arrow[d, hook] \\
E/{\sim} \arrow[ur, bend left, "\iota"] \arrow[uur, bend left, "\widehat{\iota}"] & & E/{\sim'}
\end{tikzcd} \tag{3.14}$$

We will show now that π' restricted to $|F|$, i.e., considered as a morphism $F \to \pi'(F)$ and denoted by $\pi' \restriction_F$, is an open map; note that its codomain is a full substructure of $E/\sim'$. Obviously it reflects concurrency, since π' does so on whole E. To show the zig-zag property, consider any configuration C in F with $\pi'(C) \xrightarrow{a} C'$ in $\pi'(F)$ – we search for a configuration C'' in F satisfying

$$C \xrightarrow{a} C'' \text{ and } \pi'(C'') = C'. \tag{3.15}$$

In the following we rely on the fact that the projections, being open, are pomset isomorphisms on configurations. Moreover, we implicitly exploit injectivity of ι.

Consider the family $\mathcal{C}$ of all those configurations D in E with $\pi'(D) = C'$, necessarily nonempty. We know that $C' \subseteq \pi'(F)$, hence some $D \in \mathcal{C}$ must be included in F, $D \subseteq F$. By now, none of the assumptions guarantee that the unique subconfiguration of any such D, say D', satisfying $\pi'(D') = \pi'(C)$, is equal to C. Thus we strengthen requirements on construction of ι_{k+1} in the inductive definition above. We require additionally that ι_{k+1} is chosen to satisfy the following condition:

> For any two isomorphic configurations $B_1, B_2 \subseteq \pi(F_k)$ in $E/\sim$ with $\pi'(\iota_k(B_1)) = \pi'(\iota_k(B_2))$, whenever $B_1 \xrightarrow{a} B_1'$ and $B_1' \nsubseteq \pi(F_k)$, there exists some B_2' with $B_2 \xrightarrow{a} B_2'$, $B_2' \nsubseteq \pi(F_k)$ and $\pi'(\iota_{k+1}(B_1)) = \pi'(\iota_{k+1}(B_2))$.

To ensure that this can be done, note that π and π' *represent* an shp-bisimulation, say H, containing all pairs $(\pi(C), \pi'(C))$ of configurations, for any configuration C of E, linked by pomset isomorphisms induced by $\pi^{-1}; \pi'$ (like an event structure in (3.7) together with the projections to E_1 and E_2):

$$H = \{(\pi(C), (\pi' \restriction_C) \circ (\pi \restriction_C)^{-1}, \pi'(C)) : C \text{ is a configuration in } E\}.$$

Having this, observe that $\pi'(\iota(B_1)) = \pi'(\iota(B_2))$ implies that B_1 and B_2 are related (by H) to the same configuration in $E/\sim'$, from which we conclude that some extensions of B_1 and B_2 (via $\_ \xrightarrow{a} \_$) are also related to the same configuration of $E/\sim'$, hence some (not necessarily unique) ι_{k+1} can be chosen to satisfy the condition above.

This additional requirement guarantees existence of a configuration C'' satisfying (3.15): put $B_1' := \pi(D)$ (for any $D \in \mathcal{C}$), $B_1 := \pi(D')$ (where the configuration $D' \subseteq D$ is such that $\pi'(D') = \pi'(C)$), $B_2 := \pi(C)$, and now $C'' := \iota_{k+1}(B_2')$.

Returning to diagram (3.14), by the above considerations we know that $\pi' \restriction_F$ is open, hence

$$(\widehat{\iota}; \pi' \restriction_F) : E/\sim \to \pi'(F)$$

is also open by Proposition 3.2.4. Moreover, this morphism is necessarily injective, since otherwise we would obtain an open congruence $\ker(|\pi; \widehat{\iota}; (\pi' \restriction_F)|)$ strictly coarser than $\ker(|\pi|)$, which is impossible by assumption that $\ker(|\pi|)$ is maximal. By Lemma 3.3.6 we conclude also that $(\widehat{\iota}; \pi' \restriction_F)$ is surjective. All this implies that it is an isomorphism between $E/\sim$ and $\pi'(F)$, since the functor $|\_|$ lifts isomorphisms of domain subobjects.

Repeating analogously the whole argument we construct another inverse $\iota' : E/\sim' \to E$ and the full subobject F' of E such that $(\widehat{\iota'}; \pi \restriction_{F'}) : E/\sim' \to \pi(F')$ is an isomorphism. This gives the following picture,

$$\begin{tikzcd}
E/\sim \arrow[r, phantom, "\cong"] & \pi'(F) \arrow[d, hook] \\
\pi(F') \arrow[u, hook] \arrow[r, phantom, "\cong"] & E/\sim'
\end{tikzcd}$$

with $E/\!\sim$ being isomorphic to a full subobject of $E/\!\sim'$ and vice versa. Now our aim is to show that (at least) one of subobjects inclusions is an isomorphism. For this, we need in the construction of ι' one more requirement: whenever an event e of $E/\!\sim'$ is in $\pi'(F)$, i.e., $e = \pi' \restriction_F (e')$ for some e' in F, put $\iota'(e) = e'$. This is always possible, since $(\widehat{\iota}; \pi' \restriction_F)$ is an isomorphism and that each pair of configurations from $E/\!\sim$ and $\pi'(F)$, respectively, related by this isomorphism, belongs to the bisimulation H represented by π and π' (together with the evident pomset-isomorphism). Now, we have $F \subseteq F'$, hence $\pi(F') \cong E/\!\sim$, which implies $E/\!\sim \,\cong E/\!\sim'$ and completes the proof. □

Lemma 3.5.5 and Theorem 3.5.8 enable us to apply Theorem 3.4.1:

Corollary 3.5.9 *Shp-bisimilarity of simple event structures is factorized by any indistinguishability structure consisting of maximal open congruences.*

We also considered other possibilities for a concretization functor, for instance functor Conf(_) transporting an event structure E to the set of all configurations of E, and on morphism $f : E_1 \to E_2$ giving a function transporting a configuration C to $f(C)$. For Conf(_) it is no longer true that computations create subobjects, but this problem disappears when we change to a functor giving only the set of *finite* configurations. The latter one could be also used here, although it would not allow us to easily keep track of concretizations of bisimulations, appearing crucial when dealing with open congruences.

3.6 Presheaves

Category of presheaves, i.e., set-valued functors, gives another example of application of results from the previous sections. Presheaves were proposed as a general and uniform model for concurrency [JNW93, CW96]. For an arbitrary category $\mathbb{P}$, the category $\widehat{\mathbb{P}}$ of presheaves over $\mathbb{P}^{op}$ consists of all contravariant functors from $\mathbb{P}$ to *Set* with natural transformations as morphisms between them.

It seems appropriate to consider presheaves over $\mathbb{P}^{op}$, for a subcategory $\mathbb{P}$ of observation objects. For example, category $\widehat{Bran_L}$ corresponds to synchronization forests, i.e., collections of *synchronization trees* (unfoldings of transition systems) (cf. [JNW93]). A presheaf F over $\mathbb{P}^{op}$ is a particularly simple model of behaviour. Its value $F(P)$ on an object P from $\mathbb{P}$ corresponds bijectively, by Yoneda Lemma, to the set of computations $P \to F$ (we treat here $\mathbb{P}$ as a subcategory of $\widehat{\mathbb{P}}$, as it embeds fully and faithfully into $\widehat{\mathbb{P}}$ via Yoneda embedding $P \mapsto Hom_{\mathbb{P}}(\_, P)$), i.e., the set of natural transformations $Hom_{\mathbb{P}}(\_, P) \to F$. Moreover, $F(m)$, for a morphism $m : P \to Q$ in $\mathbb{P}$, describes how "longer" computations from Q restrict uniquely to "shorter" ones (via m). This demonstrates uniformity and simplicity of presheaf models, which consist (up to the natural bijection) exclusively of their own computations and computation extensions.

For any category $\mathbb{M}$ and an observation subcategory $\mathbb{P}$, we can consider a canonical functor $\mathbb{M} \to \widehat{\mathbb{P}}$, taking an object X of $\mathbb{M}$ to the representable functor $Hom_{\mathbb{M}}(\_, X) : \mathbb{P}^{op} \to Set$. For transition systems we have a pleasing situation: the subcategory of synchronization trees is equivalent via such a canonical functor to subcategory of *rooted* presheaves, i.e., those presheaves that assign a singleton set to the initial object in $Bran_L$.[3] For event structures the picture is more complicated and until recently there was no convenient characterization

[3]This notion is meaningful for any category $\mathbb{P}$, assumed that $\mathbb{P}$ has an initial object.

of those presheaves over Pom_L which correspond to event structures – only quite recently such a characterization was proposed in [Win99].

One of advantages of considering presheaves $\mathbb{P}^{op} \to Set$ instead of concrete models is that many universal constructions like limits or colimits are obtained by a point-wise construction, hence similarly as in Set. However, results obtained for presheaves, like $\widehat{Bran_L}$ or $\widehat{Pom_L}$, can not always be translated to concrete categories.

From now on we consider presheaves over $\mathbb{P}^{op}$, for an arbitrary fixed small category $\mathbb{P}$. In the following we consider the class of open maps induced in $\widehat{\mathbb{P}}$ by $\mathbb{P}$ (when the latter is identified with a subcategory of $\widehat{\mathbb{P}}$); we often omit prefix $\mathbb{P}$-, and write in short "open" instead of "$\mathbb{P}$-open". Yoneda Lemma gives a nice characterization of open morphisms in $\widehat{\mathbb{P}}$ [JM94]: a natural transformation $f : F \to G$ is open iff all naturality squares[4]

$$\begin{tikzcd} FD \arrow[r, "f_D"] \arrow[d, "F\alpha"'] & GD \arrow[d, "G\alpha"] \\ FC \arrow[r, "f_C"] & GC \end{tikzcd}$$

are weak pullbacks in Set, for all $\alpha : C \to D$ in $\mathbb{P}$ (that is, for all $x \in FC$ and $y \in GD$ with $f_C(x) = G\alpha(y)$, there exists a not necessarily unique $z \in FD$ such that $F\alpha(z) = x$ and $f_D(z) = y$). We will rely on this characterizations in the proofs below.

At the expense of slight artificiality, we introduce a concretization functor $\|\_\|$ from $\widehat{\mathbb{P}}$ to $Set^{|\mathbb{P}|}$, category of sets indexed by $|\mathbb{P}|$. (Recall that $|\mathbb{P}|$ denotes the set of objects of $\mathbb{P}$.) This functor essentially forgets about values of presheaves on morphisms, and assigns to F the family $\|F\| = \{F(C)\}_{C \in |\mathbb{P}|}$ of sets.

Unique morphisms from the empty initial presheaf are always open, hence they should be ruled out by any appropriate concept of bisimilarity in $\widehat{\mathbb{P}}$ built up on spans of open maps. There are two ways to obtain a meaningful and nontrivial notion of bisimilarity [JNW93]. One is to restrict attention to rooted presheaves. Another way around, preferred here (as more general), is to consider only spans of *surjective* open maps in the category of all presheaves. Hence, we say that two presheaves are bisimilar if they are related by a span of surjective open maps. For synchronization trees and event structures both approaches are equivalent; they are also equivalent for the subcategory of rooted presheaves, since open maps between rooted presheaves are necessarily surjective.

We will show that every presheaf has the greatest open congruence. To this aim we first establish a correspondence between open congruences and bisimulation equivalences (Lemma 3.6.4 below).

Definition 3.6.1 *By ($|\mathbb{P}|$-sorted)* bisimulation relation *we mean any relation represented by a span of surjective open maps, where a span*

$$\begin{tikzcd} & H \arrow[dl, "f"'] \arrow[dr, "g"] & \\ F & & G \end{tikzcd}$$

represents the relation: $\{(\|f\|(s), \|g\|(s)) : s \in \|H\|\}$.

[4] Throughout this section, to make the notation more concise, we omit brackets and write FP for application.

Following the standard lines, we say that F and G are bisimilar if there exists any bisimulation relating them (or equivalently, if they are related by a span of surjective open maps).

The following two lemmas concerning colimits in $\widehat{\mathbb{P}}$ will be useful in the sequel. The first lemma is a slight generalization, in the setting of presheaves, of the fact that quotients by open congruences are open (see Proposition 3.2.3).

Lemma 3.6.2 *For any parallel pair of open maps $f, g : F \to G$ between presheaves, their coequalizer $c : G \to C$ is open. Moreover, for any other open morphism $d : G \to D$ making $f; d = g; d$, the universal map $\bar{d}$ meeting $c; \bar{d} = d$ is also open.*

Proof: In the proof we rely on the weak-pullback characterization of open maps. Moreover, we use the quotient construction of colimits in Set, as colimits in $\widehat{\mathbb{P}}$ are point-wise.

To prove that $c : G \to C$ is open, we should show that the right-most square in the following diagram is a weak pullback, for any $h : Y \to X$ in $\mathbb{P}$:

$$\begin{array}{ccccc} FX & \xrightarrow{f_X, g_X} & GX & \xrightarrow{c_X} & CX = GX/{\sim_X} \\ {\scriptstyle Fh}\big\downarrow & & {\scriptstyle Gh}\big\downarrow & & {\scriptstyle Ch}\big\downarrow \\ FY & \xrightarrow{f_Y, g_Y} & GY & \xrightarrow{c_Y} & CY = GY/{\sim_Y} \end{array}$$

By the construction of coequalizers in Set, $\sim_X$ (and similarly $\sim_Y$) is the smallest equivalence relation on GX induced by the pair of functions f_X, g_X, that is, the smallest equivalence including $\{(f_X(z), g_X(z)) : z \in FX\}$. For instance, $\sim_Y = (\sim^1_Y)^*$ is a reflexive-transitive closure of a one-step relation $\sim^1_Y$ defined by:

$$y_1 \sim^1_Y y_2 \Leftrightarrow \exists z \in FY \,.\, (f_Y(z) = y_1 \wedge g_Y(z) = y_2) \vee (f_Y(z) = y_2 \wedge g_Y(z) = y_1).$$

By the point-wise construction of colimits in $\widehat{\mathbb{P}}$ we have

$$Ch([x]_{\sim_X}) = [y]_{\sim_Y} \quad \text{iff} \quad \forall x' \in [x]_{\sim_X} \,.\, Gh(x') \in [y]_{\sim_Y}.$$

Hence, in other words, we should prove the following:

$$\forall y \in GY \,.\, \forall x' \in GX \,.\, Gh(x') \in [y]_{\sim_Y} \Rightarrow \exists x \in GX \,.\, x \sim_X x' \wedge Gh(x) = y. \tag{3.16}$$

For arbitrary $y \in GY$ and $x' \in GX$, let $y' := Gh(x')$. We have $y' \sim_Y y$, i.e., $y'(\sim^1_Y)^* y$. Assume $y' \sim^1_Y y$ (the general case follows by an easy induction), that is we have $z \in FY$ with $f_Y(z) = y' \wedge g_Y(z) = y$. Since f is open there exists $z' \in FX$ satisfying $Fh(z') = z$ and $f_x(z') = x'$. Now $x := g_X(z')$ satisfies $Gh(x) = y$ by commutativity of the square $g_X; Gh = Fh; g_Y$ and moreover $x' \sim_X x$ due to z' as $f_X(z') = x'$ and $g_X(z') = x$. This proves (3.16) and consequently implies that $c : G \to C$ is open.

To show openness of the unique mediating morphism $\bar{d}$, for an arbitrary open map $d : G \to D$, consider a diagram consisting of two commuting squares:

$$\begin{array}{ccccc} GX & \xrightarrow{c_X} & CX & \xrightarrow{\bar{d}_X} & DX \\ {\scriptstyle Gh}\big\downarrow & & {\scriptstyle Ch}\big\downarrow & & {\scriptstyle Dh}\big\downarrow \\ GY & \xrightarrow{c_Y} & CY & \xrightarrow{\bar{d}_Y} & DY \end{array}$$

We need to show that the right-most square is a weak pullback assumed that the left-most square and the whole rectangle are weak pullbacks. Take any $x \in DX$ and $y \in CY$ with $Dh(x) = \bar{d}_Y(y)$. As c_Y is surjective and $d = c; \bar{d}$ is open, there exists $z \in GX$ such that $c_Y(Gh(z)) = y$ and $d_X(z) = x$. But now $z' := c_X(z) \in CX$ satisfies $\bar{d}_X(z') = x$ and $Ch(z') = c_Y(Gh(z)) = y$, hence the right-most square is a weak pullback. $\square$

Lemma 3.6.3 *For an arbitrary family $\{F_i\}_{i\in\mathbb{I}}$ of presheaves in $\widehat{\mathbb{P}}$, the coproduct injections $\iota^i : F_i \to (\coprod_{i\in\mathbb{I}} F_i)$ are open, and moreover the universal mediating morphism*

$$\bar{g} : (\coprod_{i\in\mathbb{I}} F_i) \to G$$

is open, for any cocone $\{g_i : F_i \to G\}_{i\in\mathbb{I}}$ of open maps.

Proof: One can easily check that coproduct injections are open. The proof of openness of $\bar{g}$ is analogous to the second part of proof of Lemma 3.6.2 and amounts to showing that the right-most square in the diagram below is a weak pullback. This can be done similarly as before after observation that coproduct injections are "jointly surjective" that is for each $y \in \coprod_{i\in\mathbb{I}} F_i Y$ there exists some $k \in \mathbb{I}$ with $y \in F_k Y$:

$$\begin{tikzcd}
F_k X \arrow[r, "\iota^k{}_X"] \arrow[d, "F_k h"'] & \coprod_{i\in\mathbb{I}} F_i X \arrow[r, "\bar{g}_X"] \arrow[d, "\coprod_{i\in\mathbb{I}} F_i h"'] & GX \arrow[d, "Gh"'] \\
F_k Y \arrow[r, "\iota^k{}_Y"] & \coprod_{i\in\mathbb{I}} F_i Y \arrow[r, "\bar{g}_Y"] & GY
\end{tikzcd}$$

□

The above two lemmas can be generalized to all colimits: the colimit cocone of any diagram consisting of open maps is open; the universal morphism to any cocone (over the same diagram) consisting of open maps is also open.

Lemma 3.6.4 *Let $\sim \subseteq \|F\| \times \|F\|$ be a ($|\mathbb{P}|$-sorted) equivalence relation. $\sim$ is a bisimulation between F and F itself (cf. Definition 3.6.1) iff it is an open congruence.*

Proof: An arbitrary open congruence $\sim$ on F is a bisimulation since it is represented by a kernel pair of the projection $\pi_{F/\sim}$ that is by a pullback of $\pi_{F/\sim}$ along itself. The pullback morphisms are open (cf. [JNW93]) and obviously surjective.

For the opposite direction, having a bisimulation equivalence $\sim$, we apply Lemma 3.6.2 to the coequalizer $c : F \to H$ of a pair $f, g : G \to F$ of open surjective maps representing $\sim$. Since colimits are point-wise in $\widehat{\mathbb{P}}$, we conclude that $\sim\, = \ker(\|c\|)$, i.e., $\sim$ is the kernel of the coequalizer c, and $H = F/\sim$. □

As a straightforward corollary from Lemma 3.6.3 we obtain the greatest bisimulations in $\widehat{\mathbb{P}}$:

Lemma 3.6.5 *There exists the greatest bisimulation between any two bisimilar presheaves.*

Proof: For any two presheaves F and G, we will construct a pair $F \xleftarrow{f} H \xrightarrow{g} G$ of open maps to represent the greatest bisimulation. Let H be a coproduct of all bisimulations between F an G, i.e., $H := \coprod_{i\in\mathbb{I}} H_i$, where $\mathbb{I}$ indexes the family $\{A \xleftarrow{f_i} H_i \xrightarrow{g_i} B\}_{i\in\mathbb{I}}$ of all spans representing bisimulations, i.e., all spans of surjective open maps relating A and B. To keep this family a small set observe that it is sufficient to take one span for each bisimulation, for example, by Lemma 3.6.4, pullbacks of open quotient projections. Now f and g are constructed as the universal morphism from $\coprod_{i\in\mathbb{I}} H_i$, for the cocone $\{f_i : H_i \to A\}_{i\in\mathbb{I}}$ and $\{g_i : H_i \to B\}_{i\in\mathbb{I}}$, respectively. They are open by Lemma 3.6.3 and surjective, due to an observation that the universal morphism from a coproduct, for a cocone of open maps, is surjective whenever all cocone morphisms are surjective. □

Lemmas 3.6.4 and 3.6.5 ensure the existence of the greatest open congruences $\simeq_F$ in $\widehat{\mathbb{P}}$, which allows us to apply Theorem 3.3.4. But first we must verify whether all the assumptions from Section 3.3 are satisfied in this case. One can easily construct full subobjects and surjective quotients, hence we have the domain subobjects $\widetilde{F}$ required in Assumption 3.3.3. Similarly as in the case of event structures, there holds

Proposition 3.6.6 *For all presheaves F, $\mathbb{COMP}_F = \|F\|$ (which implies $\widetilde{F} = F$ and computations create subobjects in $\widehat{\mathbb{P}}$).*

Proof: Follows from the proof of Yoneda Lemma: the 1-1 correspondence between the set of natural transformations $Nat(Hom_{\mathbb{P}}(\_, X), F)$ and FX ensures that a computation from $Hom_{\mathbb{P}}(\_, X)$ to F can map id_X to any element of FX. $\square$

Moreover, $\|\_\|$ lifts isomorphisms of domain subobjects, but in general $\widehat{\mathbb{P}}$ is not necessarily initialized. This is caused by the empty initial presheaf, hence the solution is to restrict to only rooted presheaves:

Corollary 3.6.7 *When $\mathbb{P}$ has the initial object, bisimilarity in the full subcategory of rooted presheaves is factorized by the greatest bisimulations.*

Proof: Theorem 3.3.4 applies, since Lemmas 3.6.2–3.6.5, although formulated for the whole category $\widehat{\mathbb{P}}$, hold also for the subcategory of rooted presheaves. $\square$

Recall that Corollary 3.6.7 applies immediately to strong bisimilarity in the category of synchronization trees, equivalent to the category of rooted presheaves over $Bran_L$.

It is worthwhile to mention that the result can be easily adapted to any suitable subcategory S of $\widehat{\mathbb{P}}$ satisfying requirements from Section 3.3. In addition, we need S to be closed on those universal constructions which are involved in the proofs that is pullbacks, coproducts and coequalizers, or we need appropriate colimits in S, possibly different that in $\mathbb{P}$, to still satisfy Lemma 3.6.3 and 3.6.2 (and pullbacks of open maps to be open). For example, when $\mathbb{P}$ has the initial object ι, all considerations (and Corollary 3.6.7) stay valid for the full subcategory of presheaves having a nonempty value on ι.

3.7 Domain subobjects as colimits of computations

Similarly as in Section 3.3, let $\mathbb{M}$ denote an arbitrary category with a subcategory $\mathbb{P}$ of observation objects; we assume moreover $\mathbb{P}$ to be small. We investigate a different possibility to define domain subobjects than in Section 3.3. A domain subobject of an object A will be produced now, roughly speaking, as the colimit of all computations in A. The justification for this study is that now the analog of Lemma 3.3.6, crucial in Theorem 3.3.4, holds without assumption needed there. On the other hand, the domain subobjects constructed here are not necessarily full, hence they can not be directly used in Theorem 3.3.4. Fortunately, if they are full, they necessarily coincide with those considered in Section 3.3.

In the sequel we use the following notation. For any object A in $\mathbb{M}$ we consider the *category of $\mathbb{P}$-computations* of A, denoted below by $\int_{\mathbb{P}}^{\mathbb{M}} A$. Its objects are all pairs (P, p), where P is an object of $\mathbb{P}$ and $p : P \to A$ in $\mathbb{M}$. Its morphisms $(P, p) \to (Q, q)$ are those morphisms $m : P \to Q$ from $\mathbb{P}$ for which $p = m; q$. This category can be equivalently defined as the comma category of the inclusion $\mathbb{P} \hookrightarrow \mathbb{M}$ and the constant functor $1 \mapsto A : \mathbf{1} \to \mathbb{M}$.

Since $\mathbb{M}$ and $\mathbb{P}$ are fixed in this section, we omit sometimes sub- and superscripts and write $\int A$ for the category of computations. In the following we need the projection functor $\Delta_A : \int A \to \mathbb{P}$, taking a pair (P,p) to P and a morphism $(P,p) \xrightarrow{m} (Q,q)$ to the same morphism $P \xrightarrow{m} Q$ in $\mathbb{P}$. This functor is/induces a diagram in $\mathbb{P}$ and hence also in $\mathbb{M}$. Throughout this section we need the following assumption, slightly weaker than cocompleteness of $\mathbb{M}$:

Assumption 3.7.1 *For any object A in $\mathbb{M}$, the diagram*

$$(\int_{\mathbb{P}}^{\mathbb{M}} A) \xrightarrow{\Delta_A} \mathbb{P} \hookrightarrow \mathbb{M}$$

has a colimit, denoted in the sequel by $\mathrm{colim}\Delta_A$.

Similarly as before, we assume that category $\mathbb{M}$ is initialized (see Section 2.2.1) and is equipped with a concretization functor $|\_| : \mathbb{M} \to Set^S$, for some set S; moreover, we assume that the concrete category $(\mathbb{M}, |\_|)$ has full subobjects and surjective quotients and admits renaming of elements (see Section 2.3).

Colimit of the diagram Δ_A is intuitively a "computational image" of A, containing, roughly speaking, the whole information about A, which can be gained by observing it via computations (see Example 3.7.2 below). Let $\tau^A = \{\tau^A_\alpha\}_{\alpha \in |\int A|}$ denote the colimit cocone, consisting of a family of morphisms $\tau^A_\alpha : \Delta_A(\alpha) \to \mathrm{colim}\Delta_A$, being automatically computations in $\mathrm{colim}\Delta_A$. Let δ_A denote here the projection function, taking a pair (P,p) to the computation p. An object A together with all its computations form the canonical cocone over Δ_A, which induces the unique morphism $!_A : \mathrm{colim}\Delta_A \to A$ making $\tau^A_\alpha ; !_A = \delta_A(\alpha)$, for all objects α in $\int A$. Consider the quotient $\overline{A}$ of $\mathrm{colim}\Delta_A$ by $\ker(|!_A|)$, with the quotient projection

$$\pi_A = \pi_{\mathrm{colim}\Delta_A/\ker(|!_A|)} : \mathrm{colim}\Delta_A \to \overline{A}.$$

Object $\overline{A}$ is necessarily a subobject of A with the injection $\iota_{\overline{A}\hookrightarrow A} = \overline{!_A}$; the function $|\iota_{\overline{A}\hookrightarrow A}|$ is necessarily injective, since the following diagram commutes:

$$\begin{tikzcd} & \overline{A} \arrow[d, hook, "\iota_{\overline{A}\hookrightarrow A}"] \\ \mathrm{colim}\Delta_A \arrow[ur, "\pi_A"] \arrow[r, "!_A"'] & A \end{tikzcd} \tag{3.17}$$

As $\mathbb{M}$ admits renaming of elements, we can assume without loss of generality that $|\overline{A}| \subseteq |A|$ and that $|\iota_{\overline{A}\hookrightarrow A}|$ is an inclusion.

Example 3.7.2 *The construction of the canonical colimit arises naturally in categories considered in previous sections.*

In the category of transition systems, operation $\mathrm{colim}\Delta_{\_}$ *amounts to unfolding of a (reachable part of) transition system to a synchronization tree. For event structures, we have an even more pleasing situation – each event structure E is (up to isomorphism) equal to the canonical colimit* $\mathrm{colim}\Delta_E$. *In other words, Pom_L is a dense subcategory of event structures.*

Turning to the category of standard algebras with the path subcategory $T_\Sigma(\mathbf{2}_{OBS})$, the canonical colimit $\mathrm{colim}\Delta_A$ *there, for an algebra A, is the free algebra $T_\Sigma(|A|_{OBS})$ over*

$|A|_{OBS}$, being the colimit of the free (path) objects. In the case of partial algebras, the colimit yields similarly the appropriate member $T_i^{\Sigma}(|A|_{OBS})$ of a multifree family over $|A|_{OBS}$.

In all categories discussed here, subobjects $\overline{A}$ and $\widetilde{A}$ coincide – for a more detailed investigation of this issue, see Section 3.7.1. Interestingly, in the case of observational equivalence of (partial) algebras, the counit $\varepsilon_A = \mathrm{id}_{|A|_{OBS}}{}^{\#}$ factors through the quotient projection π_A and the domain inclusion, which gives the following picture:

$$\begin{tikzcd} & \mathrm{colim}\Delta_A \arrow[d, "\pi_A"'] \arrow[dr, "!_A"] & \cong \; T_i^{\Sigma}(|A|_{OBS}) \arrow[d, "\varepsilon_A"] \\ \widetilde{A} \;\cong & \overline{A} \arrow[r, hook] & A \end{tikzcd} \tag{3.18}$$

The colimit $\mathrm{colim}\Delta_A$ is given by $T_i^{\Sigma}(|A|_{OBS})$ and (up to the canonical isomorphism between them) ε_A is equal to $!_A$.

In all the examples above, the inclusion $\iota_{\overline{A}\hookrightarrow A}$ is open – in fact this is always the case, and follows directly from the fact that categories $\int A$ and $\int \overline{A}$ are isomorphic (see the proof of Lemma 3.7.4 below). On the other hand, two other morphisms in the triangle (3.17) need not be open in general; instead, we formulate the following:

Observation 3.7.3 *Composition with $!_A$ transforms bijectively colimit inclusions τ_α^A to computations in A; moreover,*

$$\tau_\alpha^A = m;\tau_\beta^A \quad \textit{iff} \quad \tau_\alpha^A;!_A = m;\tau_\beta^A;!_A,$$

for any $\alpha, \beta \in |\int A|$ and any morphism $m : \Delta_A(\alpha) \to \Delta_A(\beta)$ in $\mathbb{P}$.

Now we are going to prove the fact analogous to the key Lemma 3.3.6, but without assumptions needed here (we only require $\mathbb{M}$ to be initialized) – besides this, the only difference is that subobjects $\widetilde{A}$ are replaced by $\overline{A}$.

Lemma 3.7.4 *For any objects A, B in $\mathbb{M}$ and any $\mathbb{P}$-open morphism $h : \overline{A} \to \overline{B}$, $\overline{B}$ is a quotient of $\overline{A}$, with the quotient projection h.*

Before embarking on the proof we need some more notation. Assumption 3.7.1 gives raise to an endofunctor $\mathcal{K} : \mathbb{M} \to \mathbb{M}$, taking an object A to $\mathrm{colim}\Delta_A$ and a morphisms $f : A \to B$ to $!_f : \mathrm{colim}\Delta_A \to \mathrm{colim}\Delta_B$ defined as follows. A morphism f induces a cocone ρ over Δ_A consisting of morphisms $\tau_\alpha^A;!_A;f$, for all α in $|\int A|$. But from this cocone we can construct another one, over the same diagram, consisting of precisely those morphisms β, having the codomain $\mathrm{colim}\Delta_B$, for which $\beta;!_B$ is in ρ. This cocone is unique by Observation 3.7.3 and gives a unique universal morphism $!_f : \mathrm{colim}\Delta_A \to \mathrm{colim}\Delta_B$. Moreover, by uniqueness of the universal morphism the following diagram commutes:

$$\begin{tikzcd} \mathrm{colim}\Delta_A \arrow[r, "!_A"] \arrow[d, "!_f"'] & A \arrow[d, "f"] \\ \mathrm{colim}\Delta_B \arrow[r, "!_B"] & B \end{tikzcd} \tag{3.19}$$

for all $f : A \to B$ in $\mathbb{M}$, that is to say the family $\{!_A\}_{A\in|\mathbb{M}|}$ is a natural transformation.

Remark Functor $\mathcal{K}$ defined above is actually the left Kan extension of the inclusion $\mathbb{P} \hookrightarrow \mathbb{M}$ along itself and is determined up to a natural isomorphism, as all colimits $\mathrm{colim}\Delta_A$ are so.

Proof of Lemma 3.7.4: For a moment, let C be an arbitrary object in $\mathbb{M}$. By the construction of $\overline{C}$ it follows that $\iota_{\overline{C}\hookrightarrow C}$ is mono, which together with commutativity of (3.17) and Observation 3.7.3 implies that categories $\int C$ and $\int \overline{C}$ are isomorphic – an isomorphism is given by the composition with $\iota_{\overline{C}\hookrightarrow C}$. Since the composition with $\iota_{\overline{C}\hookrightarrow C}$ preserves a domain of a computation, we conclude that diagrams induced by Δ_C and $\Delta_{\overline{C}}$ are identical, hence $\mathrm{colim}\Delta_C \cong \mathrm{colim}\Delta_{\overline{C}}$. Moreover, a canonical isomorphism $i_C : \mathrm{colim}\Delta_{\overline{C}} \to \mathrm{colim}\Delta_C$ between both colimit objects, induced by the injection $\iota_{\overline{C}\hookrightarrow C}$ (i.e., $i_C = !_{\iota_{\overline{C}\hookrightarrow C}}$ according to the notation from (3.19) above), factors $!_{\overline{C}}$ through π_C. Summing up, both triangles commute in the following diagram:

$$\begin{tikzcd}
\mathrm{colim}\Delta_{\overline{C}} \arrow[r, "!_{\overline{C}}"] \arrow[d, "i_C"'] & \overline{C} \arrow[d, hook, "\iota_{\overline{C}\hookrightarrow C}"] \\
\mathrm{colim}\Delta_C \arrow[r, "!_C"'] \arrow[ur, "\pi_C"] & C
\end{tikzcd} \tag{3.20}$$

Let $h : \overline{A} \to \overline{B}$ be an open morphism in $\mathbb{M}$. By combining two copies of diagram (3.20), for A and B, respectively, with an instantiation of diagram (3.19) for h, we obtain the following:

$$\begin{tikzcd}
\mathrm{colim}\Delta_{\overline{A}} \arrow[r, "!_{\overline{A}}"] \arrow[d, "i_A"'] \arrow[drr, "!_h"] & \overline{A} \arrow[d, hook, dotted, "\iota_{\overline{A}\hookrightarrow A}"] \arrow[drr, "h"] & & \\
\mathrm{colim}\Delta_A \arrow[r, dotted, "!_A"] \arrow[ur, dotted, "\pi_A"] \arrow[drr, "i_A{}^{-1};!_h;i_B"'] & A & \mathrm{colim}\Delta_{\overline{B}} \arrow[r, "!_{\overline{B}}"] \arrow[d, "i_B"'] & \overline{B} \arrow[d, hook, "\iota_{\overline{B}\hookrightarrow B}"] \\
& & \mathrm{colim}\Delta_B \arrow[r, "!_B"'] \arrow[ur, "\pi_B"] & B
\end{tikzcd} \tag{3.21}$$

from which we choose the square determined by two diagonals π_A and π_B:

$$\begin{tikzcd}
\mathrm{colim}\Delta_A \arrow[r, "\pi_A"] \arrow[d, "k:=i_A{}^{-1};!_h;i_B"'] & \overline{A} \arrow[d, "h"] \\
\mathrm{colim}\Delta_B \arrow[r, "\pi_B"] & \overline{B}
\end{tikzcd} \tag{3.22}$$

It necessarily commutes, since all squares commute in (3.21). We begin by showing that

1. k is epi, and

2. k is a quotient projection.

(1) As h is open and $\mathbb{M}$ is initialized, for every computation $(p; !_{\overline{B}})$ in $\overline{B}$ (with p a cocone injection) there exists another computation $(p'; !_{\overline{A}})$ in $\overline{A}$ (with p' a cocone injection, again)

such that $p'; !_{\overline{A}}; h = p; !_{\overline{B}}$:

$$\begin{tikzcd}
 & \mathrm{colim}\Delta_{\overline{A}} \arrow[r, "!_{\overline{A}}"] & \overline{A} \arrow[d, "h"] \\
{} \arrow[ur, "p'"] \arrow[urr, dotted] \arrow[r, "p"] & \mathrm{colim}\Delta_B \arrow[r, "!_{\overline{B}}"] & \overline{B}
\end{tikzcd}$$

(we implicitly used here Observation 3.7.3). The upper side of diagram (3.21) commutes, hence $p'; !_h; !_{\overline{B}} = p; !_{\overline{B}}$. Moreover, since the composition with $!_{\overline{B}}$ cannot identify different cocone computations in $\mathrm{colim}\Delta_B$ by Observation 3.7.3, we conclude that for every cocone computation p in $\mathrm{colim}\Delta_{\overline{B}}$ there exists another cocone computation p' in $\mathrm{colim}\Delta_{\overline{A}}$ meeting $p'; !_h = p$. This statement can be "translated" through isomorphisms i_A and i_B as follows: for every cocone computation p in $\mathrm{colim}\Delta_B$ there exists another cocone computation p' in $\mathrm{colim}\Delta_A$ meeting $p'; k = p$.

Now assume $k; f = k; g$, from which it follows $p'; (k; f) = p'; (k; g)$, for all computations p' in $\mathrm{colim}\Delta_A$; this implies $p; f = p; g$ for all computations p in $\mathrm{colim}\Delta_B$, hence $f = g$.

(2) Consider any morphism $f : \mathrm{colim}\Delta_A \to D$ such that $\ker(|k|) \subseteq \ker(|f|)$. It is easy to observe that D together with morphisms $(p'; f)$, for all cocone computations p' in $\mathrm{colim}\Delta_A$, form a cocone over Δ_B. It is so, since for any pair of cocone computations p_1, p_2 in $\mathrm{colim}\Delta_B$ satisfying $m; p_1 = p_2$ (where m is a morphism in $\mathbb{P}$), we have two cocone computations p'_1, p'_2 in $\mathrm{colim}\Delta_A$ satisfying $m; p'_1; k = p'_2; k$, from which we conclude $m; (p'_1; f) = (p'_2; f)$. The described cocone induces the unique morphism $g : \mathrm{colim}\Delta_B \to D$ satisfying $p; g = p'; f$, for all cocone computations p, where p' is a corresponding cocone computation satisfying $p'; k = p$. Hence we obtain $p'; (k; g) = p'; f$, for all cocone computations p' in $\mathrm{colim}\Delta_A$, from which we conclude $k; g = f$ (and such g is unique, since k is epi), which completes the proof of (2).

Now we should show that h is the quotient projection. By (1) h is epi, as diagram (3.22) commutes and $(\pi_A; h) = (k; \pi_B)$ is epi, which follows from the fact that k is epi and epimorphisms compose. By (2) we conclude that $(\pi_A; h) = (k; \pi_B)$ is a quotient projection, since surjective quotient projections compose (this is routine to verify). Now we are able to prove that h is a quotient projection. Let $f : \overline{A} \to C$ be a morphism with $\ker(|h|) \subseteq \ker(|g|)$. As $|\pi_A|$ is surjective, we have $\ker(|\pi_A; h|) \subseteq \ker(|\pi_A; f|)$, hence there exists a (unique) morphism $\bar{f} : \overline{B} \to C$ making $(\pi_A; h); \bar{f} = (\pi_A; f)$. Now it easily follows from π_A being epi that $h; \bar{f} = f$, which completes the proof. $\square$

Remark In the proof of (1) we actually argued that k, being the image of $\mathcal{K}$ on an open morphism h (up to isomorphisms i_A, i_B), is open itself. This is related to the result of Cattani and Winskel [CW96], stating that all left Kan extensions along Yoneda embeddings between presheaf categories preserve openness. Notice that we showed more than this – namely that k is a surjective quotient projection.

3.7.1 Relating domain subobjects

The technical problem concerned with constructing domain subobjects from colimits of computations is that $\overline{A}$ need not to be a full subobject of A, for example for empty subcategory $\mathbb{P}$. Nevertheless, some links between $\overline{A}$ and $\widetilde{A}$ can be established. We start by relating the carrier sets:

Lemma 3.7.5 $\mathbb{COMP}_A \subseteq |\overline{A}| \subseteq |\widetilde{A}|$, *for each object* A *in* $\mathbb{M}$.

Proof: In the proof of Lemma 3.7.4 we showed that categories $\int A$ and $\int \overline{A}$ are isomorphic; this together with injectivity of the inclusion $\iota_{\overline{A}\hookrightarrow A}$ implies the first inclusion $\mathbb{COMP}_A \subseteq |\overline{A}|$. To prove the second one, notice that $\widetilde{A}$ together with all computations in $\widetilde{A}$ forms a cocone over Δ_A, which follows from $\widetilde{A}$ being the full subobject. Since $\mathbb{M}$ is initialized, by openness and monotonicity of the subobject inclusion $\iota_{\widetilde{A}\hookrightarrow A}$, categories of computations $\int A$ and $\int \widetilde{A}$ are necessarily isomorphic. Thus the induced universal morphism $g : \mathrm{colim}\Delta_A \to \widetilde{A}$ satisfies $\tau^A_\alpha; (g; \iota_{\widetilde{A}\hookrightarrow A}) = \delta_A(\alpha)$, for all α in $|\int A|$. From this we conclude that $g; \iota_{\widetilde{A}\hookrightarrow A} = !_A$, i.e., $g; \iota_{\widetilde{A}\hookrightarrow A} = \pi_A; \iota_{\overline{A}\hookrightarrow A}$:

$$
\begin{tikzcd}
 & \overline{A} \arrow[d, hook, "\iota_{\overline{A}\hookrightarrow A}"] \\
\mathrm{colim}\Delta_A \arrow[ur, "\pi_A"] \arrow[r, "!_A"'] \arrow[d, "g"'] & A \\
\widetilde{A} \arrow[ur, hook, "\iota_{\widetilde{A}\hookrightarrow A}"'] &
\end{tikzcd}
\tag{3.23}
$$

As both $|\iota_{\widetilde{A}\hookrightarrow A}|$ and $|\iota_{\overline{A}\hookrightarrow A}|$ are injective and $|\pi_A|$ is surjective, we obtain $|\overline{A}| \subseteq |\widetilde{A}|$. $\square$

From this lemma it follows that whenever $\overline{A}$ is full, it necessarily coincides with $\widetilde{A}$. Moreover, both subobjects coincide, when the assumptions from Theorem 3.3.4 are fulfilled:

Proposition 3.7.6 *When* $|\_|$ *lifts isomorphisms of domain subobjects and computations create subobjects in* $\mathbb{M}$, *then* $\overline{A} \cong \widetilde{A}$.

Proof: By the previous lemma, there exists an injective morphism $\xi : \overline{A} \to \widetilde{A}$, given by $\xi := \widehat{\iota_{\overline{A}\hookrightarrow A}}$ in diagram (3.23).

$$
\begin{tikzcd}
 & \overline{A} \arrow[dl, "\xi"'] \arrow[d, hook, "\iota_{\overline{A}\hookrightarrow A}"] \\
\widetilde{A} \arrow[r, hook, "\iota_{\widetilde{A}\hookrightarrow A}"'] & A
\end{tikzcd}
$$

Now, knowing that $\mathbb{COMP}_A = |\widetilde{A}|$, we conclude that ξ is also surjective. Moreover, ξ is open by Proposition 3.2 since $(\xi; \iota_{\widetilde{A}\hookrightarrow A}) = \iota_{\overline{A}\hookrightarrow A}$ is open and $\iota_{\widetilde{A}\hookrightarrow A}$ is mono. And finally, by the assumption that $|\_|$ lifts isomorphisms of domain subobjects we conclude that ξ is iso; since $\mathbb{M}$ admits renaming of elements, this completes the proof. $\square$

As a conclusion from the above considerations, Lemma 3.7.4 can be used in Theorem 3.3.4 only when all the assumptions needed there are satisfied. Hence, the colimit construction does not enable, in general, to drop any of assumptions from Theorem 3.3.4.

3.8 Observational equivalence

We turn back to categories of algebras, studied in Chapter 2, in order to relate the factorizability results from previous sections with the evident factorizability of algebraic observational equivalences. To stay at an appropriate level of generality, we consider an observational equivalence induced by $OBS \subset S$ in an arbitrary concrete category $\mathbb{M}$ with a concretization functor $|\_| : \mathbb{M} \to Set^S$. We also assume existence of the greatest observational congruences. The forgetful functor $|\_|_{OBS}$ is assumed to have a left multiadjoint F, so that observational morphisms are open maps. We assume moreover, as usual, that $\mathbb{M}$ has full subobjects and surjective quotients.

First we compare domain subobjects: for any object A in $\mathbb{M}$, we have:

$$A_{OBS} = \widetilde{A}. \tag{3.24}$$

Both subobjects are full, hence it is sufficient to compare their carrier sets, i.e., to show $|A_{OBS}| = |\widetilde{A}|$. We will need two inclusions:

$$|A|_{OBS} \subseteq \mathbb{COMP}_A \subseteq |A_{OBS}|.$$

The first one is immediate, if we recall that computations in A are extensions of functions $v : X \to |A|_{OBS}$ in Set^{OBS}. On the other hand, second inclusion follows directly from openness of the subobject inclusion $\iota_{A_{OBS} \hookrightarrow A}$ and from the fact, that $\mathbb{M}$ is necessarily initialized; these two facts together ensure that any computation p in A factorizes through the injective subobject inclusion:

$$\begin{tikzcd}
& A_{OBS} \arrow[d, hook] \\
F_i(X) \arrow[ur] \arrow[r, "p"'] & A
\end{tikzcd}$$

Now, first inclusion implies $|A_{OBS}| \subseteq |\widetilde{A}|$ while second $|\widetilde{A}| \subseteq |A_{OBS}|$, which concludes the proof of (3.24).

Alternatively, (3.24) could be proved using the fact that observational morphisms are open maps and recalling that A_{OBS} is the smallest full subobject having an observational inclusion while $\widetilde{A}$ is the smallest full subobject with an open inclusion.

By (3.24) we conclude that observational equivalence fits into general setting of Section 3.3, that is,

$$\text{observational congruence} \quad \Longleftrightarrow \quad \text{open congruence}$$

and the observational equivalence is factorized by the greatest open congruences on the smallest open subobjects.

Chapter 4

Coalgebra morphisms and open maps

So far we concentrated on a generalization of bisimulation by means of (spans of) open maps. An alternative direction of abstraction, aiming at a generalization of bisimulation, is to turn to a category of coalgebras, induced by an endofunctor on an arbitrary category. Among multiple references on application of coalgebras to model state-based dynamical systems we mention [Acz88, Jac95, Jac96a, Jac96b, Rei95, Rut96]. Coalgebra morphisms can be understood then as functional bisimulations, while their spans (again) give an abstract notion of a bisimulation. This can be done for different endofunctors, and a relevant special case is the category of transition systems (although without a distinguished initial state) with the zig-zag morphisms (for a slightly modified zig-zag condition, not restricted to only reachable states), equivalent to the category of coalgebras induced by the powerset functor.

Similarly as in previous chapters, a basic motivation for the work presented in this chapter is to clarify relationships between both approaches considered distinct (at least technically) and not shown explicitly to be comparable so far. Since the central notion of bisimulation is captured in both cases by spans of appropriate morphisms, the crucial point is to relate open maps and coalgebra morphisms. To this aim, we define an embedding of any category with a fixed class of open maps to an appropriate category of coalgebras with so called lax cohomomorphisms [CGH98]. A relevant observation is that under this embedding open maps correspond to coalgebra morphisms. We also demonstrate that a relation-based generalization of bisimulation related to open maps, called *path bisimulation* in [JNW93], is an instance of coalgebra bisimulation. These results indicate that the coalgebraic treatment of bisimulation is the most basic one, subsuming even the setting as abstract as that of open maps.

A crucial role in our development is played by categories of presheaves over a path subcategory. On one hand, presheaves provide a uniform and simple model for concurrent systems in which open-map bisimilarity arises in a particularly natural way. On the other hand, we show that category of presheaves is isomorphic to some subcategory of well-behaved coalgebras (called *consistent* coalgebras here). This motivates a wider research in applying presheaves as models of computation, since they appear as a central and fundamental construction especially well suited for studying generalized bisimulations.

In general, category of presheaves has more objects than concrete models, like transition systems or event structures. Therefore an interesting question arises: which presheaves correspond to concrete models in each case? For strong bisimilarity of labelled transition systems this question was answered already in [JNW93], while for instance a convenient and simple characterization of presheaves corresponding to event structures was proposed only recently in [Win99].

As a natural next step in such a program, we study closer the observational equivalence of many-sorted algebras. Some attempts were made by now to present algebraic data types explicitly in terms of coalgebras, with the aim to introduce appropriate observational concepts, but to our knowledge this was successful only in a special case when signature is monadic in non-observable sorts (cf. [Mal96] or [Cir97] in the case of hidden-sorted algebra).

We show that presheaves corresponding to standard algebras are precisely those preserving finite products. Moreover, we derive a categorical equivalence of a particular subcategory of algebras and of these presheaves, which gives an alternative coalgebraic presentation of algebras (this equivalence is a slight generalization of Lawvere's characterization of many-sorted algebras [Law63]). At a more intuitive level, this gives another evidence that the algebraic observational equivalence is of essentially coalgebraic nature. Along similar lines, we prove that partial algebras correspond to presheaves preserving finite limits. In the proofs we intensively use an adjunction between a cocomplete model category and a corresponding category of presheaves, where right adjoint, roughly, gives a (coalgebraic) *behaviour* of an object, while left adjoint returns a universal *realization* of a behaviour.

4.1 Basic definitions

The notion of coalgebra is dual to that of algebra, when the latter is presented in the categorical setting. As an example, consider an algebraic (one-sorted) signature Σ. Now, a Σ-algebra on the carrier set A is given by a mapping

$$F_\Sigma(A) \to A$$

where the endofunctor F_Σ is induced by the signature in the following way:

$$F_\Sigma(A) = \coprod_{f \in \Sigma} A^{ar(f)};$$

for an operation symbol f in Σ, $ar(f)$ denotes here its arity. Moreover, the Σ-homomorphisms are precisely those functions h which make the following diagram commute:

$$\begin{tikzcd} F_\Sigma(A) \arrow[r, "F_\Sigma(h)"] \arrow[d] & F_\Sigma(B) \arrow[d] \\ A \arrow[r, "h"'] & B \end{tikzcd}$$

which is equivalent to the requirement that h preserves the algebraic structure of A.

The notion of coalgebra is obtained by dualising above definitions; moreover, for the sake of generality we switch to an arbitrary category $\mathbb{C}$ and an arbitrary endofunctor $G : \mathbb{C} \to \mathbb{C}$. A *$G$-coalgebra* is a pair $\langle A, \alpha \rangle$ consisting of an object A together with a morphism $\alpha : A \to G(A)$. A *coalgebra morphism* from $\langle A, \alpha \rangle$ to $\langle B, \beta \rangle$ is any arrow h in $\mathbb{C}$ making the following diagram commute:

$$\begin{tikzcd} G(A) \arrow[r, "G(h)"] & G(B) \\ A \arrow[u, "\alpha"] \arrow[r, "h"'] & B \arrow[u, "\beta"'] \end{tikzcd}$$

Intuitively, $G(A)$ describes possibilities for a "one-step behaviour" in a coalgebra. Coalgebras are widely used to model state-based systems, and then the coalgebra structure α corresponds to a map that, intuitively, transforms a "state" $a \in A$ to the set of states reachable in one step from a. As an example, let us redefine labelled transition systems as coalgebras induced by an endofunctor $G : \mathit{Set} \to \mathit{Set}$ given by

$$G(X) = \mathcal{P}(X)^L,$$

where $\mathcal{P}(X)$ denoted the powerset of X and L is the set of labels. Transitions are encoded in a coalgebra structure $\alpha : A \to \mathcal{P}(A)^L$ according to the following rule:

$$s \xrightarrow{a} s' \iff s' \in \alpha(s)(a),$$

for any $a \in L$. Note that category of $\mathcal{P}(\_)^L$-coalgebras is close to the category of transition systems with zig-zag morphisms (cf. Section 1.4), although they differ in two respects. First, its objects have no distinguished initial states. Second, coalgebra morphisms correspond to those morphisms of transition systems, which satisfy a slightly modified zig-zag condition – the modification amounts to not restricting to only reachable states. The latter observation justifies a definition of coalgebraic bisimulations by spans of coalgebra morphisms (see Section 4.6).

4.2 Coalgebraic characterization of open maps

Consider any locally small category $\mathbb{M}$ together with an arbitrary small path subcategory $\mathbb{P}$. We will define an embedding of $\mathbb{M}$ into the category of coalgebras of some endofunctor on the category $\mathit{Set}^{|\mathbb{P}|}$ of $|\mathbb{P}|$-sorted sets ($|\mathbb{P}|$-indexed sets), where $|\mathbb{P}|$ is the set of objects of $\mathbb{P}$. This endofunctor, called $F_{\mathbb{P}}$ in the sequel as it is determined by $\mathbb{P}$, is defined by the following rule:

$$\{X_P\}_{P\in|\mathbb{P}|} \longmapsto \{\prod_{Q\in|\mathbb{P}|} \mathcal{P}(X_Q)^{\mathit{Hom}_{\mathbb{P}}(P,Q)}\}_{P\in|\mathbb{P}|}$$

where X_P, for $P \in |\mathbb{P}|$, denotes a component of a $|\mathbb{P}|$-sorted set X, and $\mathit{Hom}_{\mathbb{P}}(P,Q)$ stands for the set of all morphisms from P to Q in $\mathbb{P}$. Intuitively, this functor resembles $\mathcal{P}(\_)^L$, for a set L of labels, with a difference that the set of states of an $F_{\mathbb{P}}$-coalgebra is indexed by objects of $\mathbb{P}$ and transitions are labelled by appropriate morphisms of $\mathbb{P}$. On morphisms, $F_{\mathbb{P}}$ is defined as usual; for $f : X \to Y$ in $\mathit{Set}^{|\mathbb{P}|}$, $(F_{\mathbb{P}}(f))_P = \prod_{Q\in|\mathbb{P}|} h_Q$, where $h_Q : \mathcal{P}(X_Q)^{\mathit{Hom}_{\mathbb{P}}(P,Q)} \to \mathcal{P}(Y_Q)^{\mathit{Hom}_{\mathbb{P}}(P,Q)}$ is given by:

$$h_Q : g \longmapsto \lambda m \in \mathit{Hom}_{\mathbb{P}}(P,Q).\{f_Q(x) : x \in g(m)\}.$$

Let $\mathcal{C}oalg_{\mathbb{P}}$ denote the category of $F_{\mathbb{P}}$-coalgebras, i.e., pairs $\langle S,t\rangle$ with S an object in $\mathit{Set}^{|\mathbb{P}|}$ and $t : S \to F_{\mathbb{P}}(S)$ a morphism in $\mathit{Set}^{|\mathbb{P}|}$, together with coalgebra morphisms from $\langle S,t\rangle$ to $\langle S',t'\rangle$ being those $\mathbb{P}$-sorted functions $f : S \to S'$ that satisfy the usual "cohomomorphism" condition, i.e., make the following square commutative:

$$\begin{tikzcd} F_{\mathbb{P}}(S) \arrow[r, "F_{\mathbb{P}}(f)"] & F_{\mathbb{P}}(S') \\ S \arrow[u, "t"] \arrow[r, "f"] & S' \arrow[u, "t'"'] \end{tikzcd}$$

Observe that since we are in the category of $|\mathbb{P}|$-sorted sets, a coalgebra structure $t : S \to F_{\mathbb{P}}(S)$ on a set S consists of a family of functions

$$\{t_P : S_P \to \prod_{Q \in |\mathbb{P}|} \mathcal{P}(S_Q)^{Hom_{\mathbb{P}}(P,Q)}\}_{P \in |\mathbb{P}|}.$$

Moreover, any element x of $\prod_{Q \in |\mathbb{P}|} \mathcal{P}(S_Q)^{Hom_{\mathbb{P}}(P,Q)}$ is also a $|\mathbb{P}|$-sorted function,

$$x = \{x_Q : Hom_{\mathbb{P}}(P,Q) \to \mathcal{P}(S_Q)\}_{Q \in |\mathbb{P}|}.$$

For convenience, in the following we feel free to omit the subscript and write $x(m)$ for a morphism $m : P \to Q$ in $\mathbb{P}$ instead of $x_Q(m)$; this is justified by the canonical bijection:

$$\prod_{Q \in |\mathbb{P}|} \mathcal{P}(S_Q)^{Hom_{\mathbb{P}}(P,Q)} \cong \prod_{m \in \biguplus_{Q \in |\mathbb{P}|} Hom_{\mathbb{P}}(P,Q)} \mathcal{P}(S_{\text{codomain}(m)}). \tag{4.1}$$

For a coalgebra $\langle S, t : S \to F_{\mathbb{P}}(S)\rangle$, a triple $\langle \alpha, m, \beta\rangle$, where $\alpha \in S_P$, $\beta \in S_Q$ and $m : P \to Q$ in $\mathbb{P}$, satisfying $\beta \in t_P(\alpha)(m)$ will be called a transition and denoted by $\alpha \overset{m}{\to} \beta$.

Turning back to the category $\mathbb{M}$, each of its objects determines an $F_{\mathbb{P}}$-coalgebra by taking as states all its computations and as transitions all commuting triangles induced by a pair of computations. Formally, for X in $\mathbb{M}$, the corresponding $F_{\mathbb{P}}$-coalgebra has a carrier $\{Hom_{\mathbb{M}}(P,X)\}_{P \in |\mathbb{P}|}$ and transitions

$$\alpha \overset{m}{\to} \beta \iff \begin{tikzcd} P \arrow[r, "\alpha"] \arrow[d, "m"'] & X \\ Q \arrow[ur, "\beta"'] & \end{tikzcd} \text{ commutes in } \mathbb{M}. \tag{4.2}$$

To extend this mapping to morphisms, we should relax the requirement on coalgebra morphisms. Following [CGH98], a $|\mathbb{P}|$-sorted function $f : S \to S'$ between carriers of coalgebras $C = \langle S, t\rangle$ and $C' = \langle S', t'\rangle$ is called a *lax cohomomorphism* from C to C' if for each $m : P \to Q$ in $\mathbb{P}$ and $s \in S_P$,

$$\{f_Q(r) : r \in t_P(s)(m)\} \subseteq t'_P(f_P(s))(m).$$

Coalgebras together with their lax cohomomorphisms form a category $\mathcal{C}oalg_{\mathbb{P}}^{lax}$, having $\mathcal{C}oalg_{\mathbb{P}}$ as a proper subcategory ($\mathcal{C}oalg_{\mathbb{P}}$ contains those lax cohomomorphisms for which the inclusion above can be replaced by equality). Lax cohomomorphisms are like usual morphisms of transition systems, i.e., transition preserving $|\mathbb{P}|$-sorted functions:

Proposition 4.2.1 *For any coalgebras $C = \langle S, t\rangle$ and $C' = \langle S', t'\rangle$, a $\mathbb{P}$-sorted function $f : S \to S'$ is a lax cohomomorphism $C \to C'$ iff for each transition $\alpha \overset{m}{\to} \beta$ in C, $f_P(\alpha) \overset{m}{\to} f_Q(\beta)$ is a transition in C'.*

Moreover, according to the terminology from [CGH98], the lax cohomomorphisms are induced by $F_{\mathbb{P}}$ endowed by a family of pre-orders

$$\sqsubseteq_{X,Y} \subseteq Hom_{Set^{|\mathbb{P}|}}(X, F_{\mathbb{P}}(Y)) \times Hom_{Set^{|\mathbb{P}|}}(X, F_{\mathbb{P}}(Y))$$

defined by

$$f \sqsubseteq g \iff f_P(x)(m) \subseteq g_P(x)(m), \text{for each } m : P \to Q \text{ in } \mathbb{P} \text{ and } x \in X_P.$$

Now we are ready to introduce a functor $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}} : \mathbb{M} \to \mathcal{C}oalg_{\mathbb{P}}^{lax}$, acting on an object as explained in (4.2) and on a morphism by postcomposing it with computations: for $f : X \to Y$ in $\mathbb{M}$,

$$\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(f)_P : \alpha \longmapsto (\alpha; f) : Hom_{\mathbb{M}}(P, X) \to Hom_{\mathbb{M}}(P, Y).$$

A lax cohomomorphism yielded by $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}$ is not a coalgebra morphism in general. As the name $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}$ suggests, an object of $\mathbb{M}$ is mapped to a coalgebra representing its whole observable behaviour, expressed in terms of computations and path morphisms. This functor will allow us to relate $\mathbb{P}$-bisimilarity of objects of $\mathbb{M}$ with coalgebraic bisimilarity in $\mathcal{C}oalg_{\mathbb{P}}$. While spans of ($\mathbb{P}$-)open maps represent abstractly bisimulations in the category $\mathbb{M}$, for coalgebras the same role is played by spans of coalgebra morphisms. It turns out that open maps in $\mathbb{M}$ correspond via $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}$ precisely to coalgebra morphisms:

Proposition 4.2.2 *A morphism $f : X \to Y$ in $\mathbb{M}$ is $\mathbb{P}$-open iff $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(f)$ is a coalgebra morphism.*

Proof: First, notice that coalgebra morphisms $f : \langle S, t\rangle \to \langle S', t'\rangle$ are those lax cohomomorphisms satisfying the following zig-zag condition (analogously to the zig-zag morphisms of transition systems): for each $m : P \to Q$ in $\mathbb{P}$ and $\alpha \in S_P$, whenever $f_P(\alpha) \xrightarrow{m} \beta$ in $\langle S', t'\rangle$ for some $\beta \in S'_Q$, there is some $\gamma \in S_Q$ such that $\alpha \xrightarrow{m} \gamma$ in $\langle S, t\rangle$ and $f_Q(\gamma) = \beta$. Now, recalling the definition of $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}$ on objects (see (4.2)) and morphisms, the zig-zag condition for $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(f)$ can be spelled out equivalently: for each $m : P \to Q$ in $\mathbb{P}$ and $\alpha : P \to X$ in $\mathbb{M}$, whenever the following diagram commutes

$$\begin{tikzcd} P \arrow[r, "\alpha"] \arrow[d, "m"'] & X \arrow[r, "f"] & Y \\ Q \arrow[urr, "\beta"'] & & \end{tikzcd}$$

for some $\beta : Q \to Y$ in $\mathbb{M}$, there exists some $\gamma : Q \to X$ making the following triangles commute:

$$\begin{tikzcd} P \arrow[r, "\alpha"] \arrow[d, "m"'] & X \\ Q \arrow[ur, "\gamma"'] & \end{tikzcd} \qquad\qquad \begin{tikzcd} & X \arrow[d, "f"] \\ Q \arrow[ur, "\gamma"] \arrow[r, "\beta"'] & Y \end{tikzcd}$$

This is obviously equivalent to $\mathbb{P}$-openness of $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(f)$. $\square$

A very general and intuitive conclusion from the above fact is that spans of open maps, intended to represent very abstractly and uniformly bisimulations for different models of computation, are essentially spans of coalgebra morphisms. This justifies the opinion that coalgebraic bisimulation captures not only the essence of this notion, but is also widely applicable to many different models. We will further expand on this while studying examples in the following sections.

Not all coalgebras are in the image of the functor. Those which do, are well-behaved in a sense that they are *consistent* with composition of path morphisms:

Definition 4.2.3 *A $F_{\mathbb{P}}$-coalgebra $\langle S, t\rangle$ is* consistent *if*

1. $\alpha \stackrel{\mathrm{id}_P}{\to} \alpha$, *for each* $P \in |\mathbb{P}|$ *and* $\alpha \in S_P$,

2. *whenever* $\alpha \stackrel{m}{\to} \beta$ *and* $\beta \stackrel{n}{\to} \gamma$, *there is a compose transition* $\alpha \stackrel{m;n}{\to} \gamma$ *in* $\langle S, t\rangle$,

3. *for each* $\beta \in S_Q$ *and* $m : P \to Q$ *in* $\mathbb{P}$, *there is exactly one* $\alpha \in S_P$ *with* $\alpha \stackrel{m}{\to} \beta$.

A coalgebra satisfying only (3) will be called reachable *in the sequel.*

One easily recognizes some categorical aspects among requirements for consistent coalgebras, like the existence of identities or definedness of composition. It turns out that the consistent coalgebra $\mathcal{B}eh^{\mathbb{M}}_{\mathbb{P}}(X)$, for an object X in $\mathbb{M}$, is another way of looking at the small category of $\mathbb{P}$-computations $\int^{\mathbb{M}}_{\mathbb{P}} X$ (cf. Section 3.7). States of the coalgebra $\mathcal{B}eh^{\mathbb{M}}_{\mathbb{P}}(X)$ are precisely objects of $\int^{\mathbb{M}}_{\mathbb{P}} X$, while its transitions correspond to morphisms.

For the constructions presented in this section to work, $\mathbb{P}$ must be small. However, in the following sections we allow $\mathbb{P}$ to be only essentially small, and implicitly assume, when convenient, working with some its skeleton.

4.3 Examples

As an example illustrating definitions from the previous section, let us consider the category TS_L, introduced in Section 1.4; its objects are transition systems (with initial states) labelled by a fixed set L of labels; its morphisms are functions which preserve transitions and initial state. Recall that a path subcategory appropriate for strong bisimilarity is the full subcategory $Bran_L$ of finite sequences of transitions. This means that a set of states of a F_{Bran_L}-coalgebra is essentially indexed by the set L^* of all finite sequences of actions, and transitions are labelled by all prefix-embeddings of a shorter sequence into a longer one. Moreover, the functor $\mathcal{B}eh^{TS_L}_{Bran_L}$ takes a transition system T to a coalgebra representing its unfolding, whose states correspond to finite sequences of actions in T starting in the initial state.

What about the category TS^0_L of transition systems without a distinguished initial state? This category (whose morphisms are all transition-preserving mappings from states to states) is isomorphic to the category of the coalgebras for a powerset functor $\mathcal{P}(\_)^L$ together with lax cohomomorphisms (for the canonical order endowment of the powerset endofunctor cf. [CGH98]). To find out how TS^0_L fits into our framework, by Proposition 4.2.2 we should find a path subcategory inducing as open maps precisely the $\mathcal{P}(\_)^L$-coalgebra homomorphisms, i.e., all functions $f : S \to S'$ satisfying the following zig-zag property[1]: for any $s \in S$, whenever $f(s) \stackrel{a}{\to} s'$, then $s \stackrel{a}{\to} r$, for some $r \in S$ satisfying $f(r) = s'$. As such a path subcategory let us take Single_L, the category having as objects a (unique up to isomorphism) one-state transition system with no transitions and all single-transition systems (i.e., those with two states related by one transition); the only morphisms of Single_L, besides identities, are functions mapping a one-state system to the starting state of a single

[1]This is slightly different from the analogous property characterizing $Bran_L$-open morphisms between transition systems with initial states, where a zig-zag condition is only required for reachable states s.

transition:

objects : morphisms :

$$
\begin{tikzcd}
 & \bullet & & & \bullet \\
\bullet & \bullet \arrow[u, "a"] & (a \in L) & \bullet \arrow[r, dotted, "f_a"] & \bullet \arrow[u, "a"']
\end{tikzcd}
$$

Now, consider category of coalgebras for the endofunctor F_{Single_L} on $Set^{L \cup \{*\}}$ (the indexing set $L \cup \{*\}$ corresponds to objects of Single_L, with $*$ representing the one-state system). The carrier set of an F_{Single_L}-coalgebra $\langle S, t \rangle$ corresponding to a transition system T contains the set of its states (S_*) and for each $a \in L$, the set of a-transitions (S_a) in T. Moreover, $S_* \ni s \xrightarrow{f_a} \langle s, a, s' \rangle \in S_a$ iff $\langle s, a, s' \rangle$ is a transition in T. Observe that reachability amounts in this case to the requirement that each transition has precisely one starting state. On the other hand, F_{Single_L}-coalgebras have no explicit information about a final state of a transition – this would require introducing into Single_L extra morphisms, mapping the one-state system to the final state of a single transition. But this would change the class of induced open maps and we would loose the property that $\mathcal{P}(\_)^L$-coalgebra maps coincide with Single_L-open morphisms, and thus correspond (via $\mathcal{B}eh^{TS^0_L}_{\mathrm{Single}_L}$) precisely to F_{Single_L}-coalgebra maps.

4.4 Presheaves as coalgebras

As we argued already in Section 3.6, it is reasonable to consider presheaves over $\mathbb{P}^{op}$, for a subcategory $\mathbb{P}$ of paths – for example, category $\widehat{Bran_L}$ corresponds to synchronization forests (cf. [JNW93]). Recall that for any category $\mathbb{M}$ and a path subcategory $\mathbb{P}$, we can consider a canonical functor $\mathcal{Y}^{\mathbb{M}}_{\mathbb{P}} : \mathbb{M} \to \widehat{\mathbb{P}}$, mapping an object X of $\mathbb{M}$ to the restriction of contravariant hom-functor $Hom_{\mathbb{M}}(\_, X) : \mathbb{P}^{op} \to Set$. As we already mentioned in Section 3.6, the subcategory of synchronization trees is equivalent via such a canonical functor to subcategory of *rooted* presheaves, i.e., those assigning to the initial object in $Bran_L$ (or generally in $\mathbb{P}$) a singleton set.

From now on we consider presheaves over an arbitrary fixed small category $\mathbb{P}$. The behavior functor $\mathcal{B}eh^{\widehat{\mathbb{P}}}_{\mathbb{P}}$ maps presheaves to consistent coalgebras; moreover, we prove that every consistent coalgebra corresponds to some presheaf. This could be expected, since the consistency condition in Definition 4.2.3 guarantees in fact a kind of "functorial" properties of transitions, e.g., preservation of composition (2) and identities (1).

Proposition 4.4.1 *Category $\widehat{\mathbb{P}}$ is isomorphic via $\mathcal{B}eh^{\widehat{\mathbb{P}}}_{\mathbb{P}}$ to the full subcategory of consistent $F_{\mathbb{P}}$-coalgebras with lax cohomomorphisms, with the inverse functor $\mathcal{I}nv$ given by:*

$$
\begin{gathered}
\mathcal{I}nv(\langle S, t \rangle)(P) = S_P, \\
\mathcal{I}nv(\langle S, t \rangle)(m : P \to Q)(\beta \in S_Q) = \alpha \in S_P, \quad \textit{when } \beta \in t_P(\alpha)(m), \\
\mathcal{I}nv(f : \langle S, t \rangle \to \langle S', t' \rangle) = f : \mathcal{I}nv(\langle S, t \rangle) \to \mathcal{I}nv(\langle S', t' \rangle).
\end{gathered}
$$

Proof: Note that in the definition above, the action of $\mathcal{I}nv(\langle S, t \rangle)$ on morphisms is determined uniquely due to the condition (3) in Definition 4.2.3. Moreover, $\mathcal{I}nv$ always yields a functor, i.e., $\mathcal{I}nv(\langle S,t\rangle)$ preserves composition and identities, which is guaranteed by conditions (2) and (1), respectively. Both compositions of $\mathcal{B}eh_{\mathbb{P}}^{\widehat{\mathbb{P}}}$ and $\mathcal{I}nv$ result in identities, since they change neither carriers of objects nor morphisms and their only relevant action is to recode an internal structure (a behavior) of an object in one of two ways:

- for a presheaf F, a functional restriction of a "longer" computation $Q \to F$ (which is in $F(Q)$ by Yoneda Lemma) via $m : P \to Q$ to a "shorter" computation $P \to F$, represented by $F(m) : F(Q) \to F(P)$,
- for a coalgebra $\langle S, t\rangle$, the set of all extensions of a shorter computation to a longer one via $m : P \to Q$, represented by a coalgebra structure t.

Concerning morphisms, by Proposition 4.2.1 it follows that the naturality condition for a $|\mathbb{P}|$-sorted function between two presheaves is equivalent to the lax cohomomorphism condition w.r.t. the corresponding coalgebra structures. □

This proposition indicates that consistent $F_{\mathbb{P}}$-coalgebras with lax cohomomorphisms are just another formulation of the category of presheaves over $\mathbb{P}$. Moreover, by Proposition 4.2.2, condition on a $\mathbb{P}$-sorted function to be a coalgebra morphism is equivalent to $\mathbb{P}$-openness of this function (treated as a natural transformation) in $\widehat{\mathbb{P}}$. Presheaves seem to be better suited since we don't need to impose any consistency requirements on them. On the other hand, the notion of bisimulation (and bisimilarity) for coalgebras arises more naturally with coalgebra morphisms, and we avoid formulating any explicit condition on morphisms to represent a bisimulation, like the property of openness. Common for both approaches is that we start by choosing a path category $\mathbb{P}$, which stands for all possible observations to be performed on objects.

The isomorphism agrees with functors $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}$ and $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}$, i.e., $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}};\mathcal{B}eh_{\mathbb{P}}^{\widehat{\mathbb{P}}} = \mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}$. From this it follows that for any morphism f in $\mathbb{M}$,

$$f \text{ is } \mathbb{P}\text{-open in } \mathbb{M} \iff \mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(f) \text{ is open in } \widehat{\mathbb{P}}. \tag{4.3}$$

This coincidence was observed already in [JNW93] in the case when $\mathbb{P}$ is a full and dense subcategory of $\mathbb{M}$ – as we see here, this assumption is not necessary.

Our coalgebraic characterization of presheaves with open morphisms is in contrast with a standard result (cf. for instance [MM94]), stating that the category of presheaves over $\mathbb{P}$ (with all natural transformations) is equivalent to the category of coalgebras of the comonad in $Set^{|\mathbb{P}|}$ defined by the rule:

$$\{X_P\}_{P\in|\mathbb{P}|} \longmapsto \{\prod_{Q\in|\mathbb{P}|} X_Q^{Hom_{\mathbb{P}}(Q,P)}\}_{P\in|\mathbb{P}|} \;\cong\; \{\prod_{m\in\biguplus_{Q\in|\mathbb{P}|} Hom_{\mathbb{P}}(Q,P)} X_{\text{codomain}(m)}\}_{P\in|\mathbb{P}|}.$$

(cf. this with (4.1) in Section 4.2; notice that $Hom_{\mathbb{P}}(P,Q)$ is replaced by $Hom_{\mathbb{P}}(Q,P)$). Equivalently, presheaves can be presented as algebras of the monad defined by a dual construction:

$$\{X_P\}_{P\in|\mathbb{P}|} \longmapsto \{\coprod_{Q\in|\mathbb{P}|} X_Q^{Hom_{\mathbb{P}}(P,Q)}\}_{P\in|\mathbb{P}|} \;\cong\; \{\coprod_{m\in\biguplus_{Q\in|\mathbb{P}|} Hom_{\mathbb{P}}(P,Q)} X_{\text{codomain}(m)}\}_{P\in|\mathbb{P}|}.$$

This time the original direction of arrows is restored ($Hom_{\mathbb{P}}(P,Q)$), but the product is replaced by coproduct.

These two (co)algebraic presentations of presheaves seem to be more elegant, since they do not require imposing any additional conditions, like our consistency condition. On the other hand, their serious disadvantage is that (co)algebra morphisms correspond to all morphisms of presheaves, not only to those which are open. Our approach is better suited for behavioural semantics, since it relates directly open maps and coalgebra morphisms.

4.5 Many-sorted algebras as coalgebras

By Proposition 4.2.2, spans of $\mathbb{P}$-open maps in $\mathbb{M}$ are mapped to spans of coalgebra morphisms, in the image subcategory $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(\mathbb{M})$, being usually a proper subcategory of $\mathcal{C}oalg_{\mathbb{P}}^{lax}$. To be able to relate both bisimilarities, it is an important issue to characterize this image subcategory, or equivalently to find a necessary and sufficient condition on a consistent coalgebra (or a presheaf) to correspond to some object from $\mathbb{M}$. More formally, we say that a coalgebra (presheaf) C *corresponds* to an object X from $\mathbb{M}$ via $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}$ ($\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}$) if $C \cong \mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(X)$ ($C \cong \mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(X)$).

In this section we inspect closer a relationship between category of many-sorted algebras and induced category of presheaves. We study separately standard and partial algebras. In both cases we derive characterizations of presheaves corresponding to an algebra. By the results from previous sections, this immediately leads to a kind of behavioural presentation of algebras as a subclass of consistent coalgebras. Moreover, in Section 4.5.2 we recast the results more abstractly, for an arbitrary deterministic bisimilarity.

For the rest of this section we fix an algebraic many-sorted signature Σ, with a set S of sorts, and a subset $OBS \subseteq S$ of *observable sorts.* The characterization of observational equivalence given by Theorems 1.3.4 and 1.3.5 in Section 1.3 will be helpful in the sequel.

4.5.1 Standard algebras

A crucial fact for our further considerations is that observational equivalence is (re)definable by means of open maps. According to Corollary 2.1.7 in Section 2.1, there are many possibilities for a path subcategory. The best suited for our purposes is the full subcategory $\mathbb{T}_{OBS}$ of free algebras over finite OBS-sorted sets, (i.e., over those X with disjoint union $\biguplus_{s\in S} X_s$ finite). The carrier of each such free algebra $T_\Sigma(X)$ is the set of all terms with variables in X, and computations from $T_\Sigma(X)$ correspond to valuations of variables in X; morphisms in $\mathbb{T}_{OBS}$ are given by substitutions.

We know already that Σ-algebras are mapped to their behaviours in $\widehat{\mathbb{T}_{OBS}}$ via $\mathcal{Y}_{\mathbb{T}_{OBS}}^{Alg_\Sigma}$. Observe that an algebra is always mapped to a functor which preserves finite products, i.e., transports a finite coproduct (diagram) in $\mathbb{T}_{OBS}$ to product (diagram) in Set – for instance, computations from $T_\Sigma(X_1 \uplus X_2)$ (i.e., valuations from $X_1 \uplus X_2$), the coproduct of $T_\Sigma(X_1)$ and $T_\Sigma(X_2)$ (in $\mathbb{T}_{OBS}$ and in Alg_Σ as well), bijectively correspond to pairs of valuations from X_1 and X_2. It turns out that each presheaf preserving finite products corresponds to some Σ-algebra. (In particular, note that each such presheaf is automatically rooted, i.e., maps initial algebra (empty coproduct) in $\mathbb{T}_{OBS}$ to a singleton.)

Lemma 4.5.1 *A presheaf over $\mathbb{T}_{OBS}$ corresponds to a Σ-algebra iff it preserves finite products.*

This fact is closely related to the standard Lawvere's representation ([Law63]) of models of algebraic equational theories as finite product preserving functors. However, our situation is different in two respects. First, we consider *no* equations, i.e., this is the special case when the theory is empty. Second, our syntactic category $\mathbb{T}_{OBS}$ contains term algebras over only *observable* variables – only when $OBS = S$ our result (together with Lemma 4.5.2) cuts down to the standard Lawvere's result, since $\mathbb{T}_S$ is dual to the classifying category of the empty equational theory (cf. [Pit95]).

Instead of proving Lemma 4.5.1 directly, we will prove Lemma 4.5.2, from which the former one follows immediately. Our argument is based on the following method of recovering an algebra from a behaviour. For an arbitrary cocomplete $\mathbb{M}$ and any small path subcategory $\mathbb{P}$, consider a category $\int_{\mathbb{P}}^{\widehat{\mathbb{P}}} F$ of computations (cf. Section 3.7) of a presheaf F (it is also called *category of elements* of F). F induces an object in $\mathbb{M}$, as a colimit of the diagram given by the projection functor Δ_F, taking (P, α) to P:

$$\mathrm{R}_{\mathbb{P}}^{\mathbb{M}}(F) = \mathrm{colim}\Big(\big(\int_{\mathbb{P}}^{\widehat{\mathbb{P}}} F\big) \xrightarrow{\Delta_F} \mathbb{P} \hookrightarrow \mathbb{M}\Big). \tag{4.4}$$

$\mathrm{R}_{\mathbb{P}}^{\mathbb{M}}(F)$ can be intuitively understood as a reconstruction of an object from a "behaviour" F. By universal properties of colimits, this construction can be extended to a functor $\mathrm{R}_{\mathbb{P}}^{\mathbb{M}} : \widehat{\mathbb{P}} \to \mathbb{M}$ (cf. [MM94], I.5).

Following an intuition that a presheaf corresponding to an algebra represents its "behaviour", an algebra given by (4.4) can be seen as a canonical realization of behaviour F. In Lemma 4.5.2 we focus on those algebras which, roughly, are canonical realizations of their own behaviours. Formally, this is the case when $A \cong \mathrm{R}_{\mathbb{T}_{OBS}}^{Alg_\Sigma}(\mathcal{Y}_{\mathbb{T}_{OBS}}^{Alg_\Sigma}(A))$, or equivalently:

$$A = \mathrm{colim}\Delta_A. \tag{4.5}$$

More specifically, given A, the algebra $\mathrm{colim}\Delta_A$ is isomorphic to a quotient of the free algebra $T_\Sigma(|A|_{OBS})$ over carrier sets of observable sorts by all observable equalities of A, i.e., by the smallest congruence containing all pairs (a, t) such that $t_{A[\mathrm{id}]} = a$.

Let us call an algebra A *initially abstract* if it satisfies (4.5). In general not all algebras are initially abstract, and the universal morphism $\mathrm{colim}\Delta_A \xrightarrow{!} A$ in not necessarily an isomorphism. In fact, an algebra is initially abstract iff it is isomorphic to $\mathrm{colim}\Delta_A$, for some algebra A. The following fact (implying directly Lemma 4.5.1) gives a coalgebraic presentation of initially abstract algebras:

Lemma 4.5.2 *The following categories are equivalent (via $\mathcal{Y}_{\mathbb{T}_{OBS}}^{Alg_\Sigma}$ and $\mathrm{R}_{\mathbb{T}_{OBS}}^{Alg_\Sigma}$):*

- *the full subcategory of initially abstract Σ-algebras,*
- *the full subcategory of presheaves over $\mathbb{T}_{OBS}$ preserving finite products.*

(When $OBS = S$, the subcategory of finite product preserving presheaves is equivalent to the whole category Alg_Σ, since all algebras are initially abstract then.)

Proof (sketch): We define a many-sorted signature Σ^* and then we use Lawvere's equivalence of the category Alg_{Σ^*} of algebras over Σ^* and the category $\mathrm{FP}(\widehat{\mathbb{T}})$ of finite product preserving presheaves over $\mathbb{T}$; by $\mathbb{T}$ we denote a subcategory of free Σ^*-algebras over a finite set.

Σ^* is defined as follows. Its sorts are all observable sorts from Σ. Its operations are induced by Σ-terms t with observable variables s.t.

- the result sort s of t is observable,
- t has no proper subterms of observable sorts, besides variable occurrences (as in the case of OBS-atomic terms from Section 2.4.2),
- no variable occurs twice in t.

More precisely, operations of Σ^* are induced by representatives of abstraction classes of an equivalence on such terms: the equivalence relates a pair (t_1, t_2) if there exists a bijection between variables of t_1 and variables of t_2 s.t. t_1 renamed according to that bijection is equal t_2 (i.e., variable names are irrelevant).

For each representative t, with variables $x_1, \ldots, x_n$ (ordered by a fixed enumeration of the set of variables) of sorts $s_1, \ldots, s_n$, we have a new operation in Σ^*:

$$\bar{t} : s_1, \ldots, s_n \to s.$$

The subcategory of initially abstract Σ-algebras is equivalent to Alg_{Σ^*}. The equivalence G takes an initially abstract algebra A to a Σ^*-algebra $\bar{A}$:

$$\begin{aligned} |\bar{A}| &:= |A|_{OBS}, \\ \bar{t}^{\bar{A}}(a_1, \ldots, a_n) &:= t_{A[x_i \mapsto a_i]}, \end{aligned}$$

and a Σ-homomorphism h to $|h|_{OBS}$. Observe that each Σ^*-algebra induces uniquely an initially abstract Σ-algebra, up to isomorphism. Similarly, each Σ^*-homomorphism extends uniquely a Σ-homomorphism between the appropriate Σ-algebras. Moreover, G cuts down to an equivalence $\mathrm{G}_{\mathbb{T}}$ between $\mathbb{T}_{OBS}$ and $\mathbb{T}$.

The equivalence between initially abstract Σ-algebras and $\mathrm{FP}(\widehat{\mathbb{T}_{OBS}})$ follows now since the following diagram commutes up to isomorphism:

$$\begin{tikzcd} \text{initially abstract } \Sigma\text{-algebras} \arrow[r, "\mathcal{Y}^{Alg_\Sigma}_{\mathbb{T}_{OBS}}"] \arrow[d, "G"'] & \mathrm{FP}(\widehat{\mathbb{T}_{OBS}}) \\ Alg_{\Sigma^*} \arrow[r, "\mathcal{Y}^{Alg_{\Sigma^*}}_{\mathbb{T}}"] & \mathrm{FP}(\widehat{\mathbb{T}}) \arrow[u, "\mathrm{G}_{\mathbb{T}};\_"'] \end{tikzcd}$$

Left, right and bottom sides of the square are equivalences of categories, hence the upper side is an equivalence too. □

If A is initially abstract, a colimit cocone can be chosen to consist of maps $\Delta_A(P,p) \xrightarrow{p} A$. This means that $\mathbb{T}_{OBS}$ is a *dense* subcategory of the full subcategory of initially abstract algebras. Hence we cannot do better than in Lemma 4.5.2, since $\mathcal{Y}^{Alg_\Sigma}_{\mathbb{T}_{OBS}}$ is full and faithful on a subcategory of Σ-algebras containing $\mathbb{T}_{OBS}$ as a subcategory, if and only if $\mathbb{T}_{OBS}$ is dense (for details see [Mac71], section X.6). On the other hand, initially abstract algebras are not very rare, and each equivalence class of observational equivalence has an initially abstract representative. The representatives are unique (up to isomorphism), hence observationally equivalent pairs of algebras are those mapped to isomorphic presheaves (their behaviours).[2]

Proposition 4.5.3 *Σ-algebras A and B are observationally equivalent if and only if*

$$\mathcal{Y}^{Alg_\Sigma}_{\mathbb{T}_{OBS}}(A) \cong \mathcal{Y}^{Alg_\Sigma}_{\mathbb{T}_{OBS}}(B).$$

[2] This is obviously not the case in general, for arbitrary $\mathbb{M}$ and $\mathbb{P}$.

This is in accordance with an idea of a correspondence of [Sch87], which can be thought of as an algebraic counterpart of a bisimulation: a correspondence can be seen as a proof of an isomorphism between the representatives. Proposition 4.5.3 is generalized in Section 4.5.2 for an arbitrary deterministic $\mathbb{P}$-bisimilarity.

The initially abstract algebras correspond, roughly, to initial algebras within abstraction classes of observational equivalence (although, more precisely, they are only weakly initial, i.e., the universal morphism need not be determined uniquely). In this sense they are dual to the fully-abstract representatives of abstraction classes, which are weakly terminal objects in the abstraction classes. It would be worthwhile to inspect perspectives offered by initially abstract algebras for the behavioural semantics of algebraic specifications. Note that the category of initially abstract algebras (and the corresponding subcategory of presheaves) enjoys the following full-abstraction property: its two objects are observationally equivalent iff they are isomorphic.

Lemma 4.5.2 can be contrasted with other coalgebraic views of many-sorted algebras [Mal96] or hidden-sorted algebras [Cir97]. An advantage of our approach is that we need not to impose any specific requirements on Σ, as it was done in [Mal96] and [Cir97], where only signatures *monadic* in observable sorts were allowed.

4.5.2 Deterministic bisimilarity

Notation from the previous sections supports the intuition about a behaviour-realization relationship. In the case of a path subcategory $\mathbb{P}$ of a cocomplete category $\mathbb{M}$, the functors $\mathrm{R}_{\mathbb{P}}^{\mathbb{M}}$ and $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}$ form an adjunction (this is a special case of Theorem 2, p. 41 in [MM94]):

$$\begin{tikzcd} \widehat{\mathbb{P}} \arrow[r, shift left, "\mathrm{R}_{\mathbb{P}}^{\mathbb{M}}"] & \mathbb{M} \arrow[l, shift left, "\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}"] \end{tikzcd}$$

with the right adjoint giving the behaviour of an object from $\mathbb{M}$ while the left adjoint returning the canonical realization of a behaviour. It is desirable to closer investigate some further examples of this general situation.

We recast here results of the previous section in the context of an arbitrary *deterministic* bisimilarity (cf. Section 2.2.1). This is intended to clarify the general picture and to shorten tedious verification of representation theorems analogous to Lemma 4.5.2 in each separate case. Roughly, when $\mathbb{P}$-bisimilarity is deterministic, the subcategory of presheaves corresponding to objects from $\mathbb{M}$ is equivalent to the full subcategory of initially abstract objects. Moreover, those presheaves are common representatives of the whole abstraction classes of $\mathbb{P}$-bisimilarity.

In Section 4.5.3 we investigate partial algebras in detail.

Assume here that $\mathbb{P}$-bisimilarity in $\mathbb{M}$ is deterministic – in such a case (4.3) from Section 4.4 can be reformulated to

$$f \text{ is } \mathbb{P}\text{-open} \iff \mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(f) \text{ is etale in } \widehat{\mathbb{P}}, \tag{4.6}$$

where *etale* morphisms in $\widehat{\mathbb{P}}$, according to [JM94], are those natural transformations g from F to G whose naturality squares

$$\begin{tikzcd} FQ \arrow[r, "g_Q"] \arrow[d, "F\alpha"'] & GQ \arrow[d, "G\alpha"] \\ FP \arrow[r, "g_P"] & GP \end{tikzcd}$$

are pullbacks in Set, for each morphism $\alpha : P \to Q$ in $\mathbb{P}$ (cf. this with the weak-pullback characterization of open maps in Section 3.6). This can be proved analogously as Proposition 4.2.2. Assume moreover in this section that $\mathbb{P}$ and $\mathbb{M}$ have a common initial object. This guarantees that each presheaf corresponding to an object of $\mathbb{M}$ is rooted. In addition this allows us to prove the following fact:

$$f \text{ is } \mathbb{P}\text{-open} \iff \mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(f) \text{ is an isomorphism in } \widehat{\mathbb{P}}. \tag{4.7}$$

To show (4.7), observe that for an etale morphism g between rooted presheaves F and G, the following diagram is a pullback in Set:

$$\begin{tikzcd} FP \arrow[r, "g_P"] \arrow[d, "!"'] & GP \arrow[d, "!"] \\ \{*\} \arrow[r, "\mathrm{id}"] & \{*\} \end{tikzcd}$$

for any object P in $\mathbb{P}$, where $\{*\}$ stands for the value of F and G on an initial object in $\mathbb{M}$ (and in $\mathbb{P}$). Hence each etale morphism between rooted presheaves is iso, which together with (4.6) proves (4.7).

Under some additional assumption, fulfilled by algebraic observational equivalences, we derive Proposition 4.5.4, generalizing Proposition 4.5.3 from the previous section. From now on we assume, similarly as in Assumption 3.7.1 in Section 3.7, that the colimit of diagram Δ_A exists, for each object A in $\mathbb{M}$.

Proposition 4.5.4 *Assume that for each object A in $\mathbb{M}$, the induced morphism*

$$!_A : \mathrm{colim}\Delta_A \to A$$

is open. Then, for any objects A, B in $\mathbb{M}$,

$$A \sim_{\mathbb{P}} B \iff \mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(A) \cong \mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(B).$$

Proof: Whenever A and B are related by a span of open maps, by (4.7) $\mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(A)$ and $\mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(B)$ are related by the span of isomorphisms, hence are isomorphic themselves.

For the opposite direction, assume $\mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(A) \cong \mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(B)$. Recall from the previous section that

$$\mathrm{colim}\Delta_A = \mathrm{R}^{\mathbb{M}}_{\mathbb{P}}(\mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}(A)), \tag{4.8}$$

hence we conclude $\mathrm{colim}\Delta_A \cong \mathrm{colim}\Delta_B$. Now, by openness of mediating morphisms $!_A$ and $!_B$, we conclude $A \sim_{\mathbb{P}} B$. □

Assumption needed in Proposition 4.5.4 justifies the choice of path subcategory $\mathbb{T}_{OBS}$ for standard algebras – recall that the canonical homomorphism from the initially abstract $\mathrm{colim}\Delta_A$ to A is always observational. Furthermore, we restate here more generally an equivalence of categories from Lemma 4.5.2:

Proposition 4.5.5 *Assume, together with assumption in Proposition 4.5.4, that $\mathbb{P}$ is a full subcategory of $\mathbb{M}$. The following categories are equivalent (via $\mathcal{Y}^{\mathbb{M}}_{\mathbb{P}}$ and $\mathrm{R}^{\mathbb{M}}_{\mathbb{P}}$):*

- *the full subcategory of initially abstract objects of $\mathbb{M}$, i.e., those satisfying $A = \mathrm{colim}\Delta_A$, with a colimit cocone consisting of maps $\Delta_A(P, p) \xrightarrow{p} A$;*

- *the full subcategory of presheaves over* $\mathbb{T}$ *corresponding to some object of* $\mathbb{M}$.

Proof: From the assumption that $!_A : \mathrm{colim}\Delta_A \to A$ is open it follows that $\mathrm{colim}\Delta_A$ is initially abstract, for each object A. In fact, by (4.7) we conclude that $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(A)$ and $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(\mathrm{colim}\Delta_A)$ are isomorphic, and now by (4.8) we obtain

$$\mathrm{colim}\Delta_A \cong \mathrm{R}_{\mathbb{P}}^{\mathbb{M}}(\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(\mathrm{colim}\Delta_A)) \cong \mathrm{colim}\Delta_{\mathrm{colim}\Delta_A}.$$

Since $\mathbb{P}$ is full, all its objects are automatically initially abstract. Hence, by the very definition, $\mathbb{P}$ is a dense subcategory of the full subcategory of initially abstract objects, hence $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}$ restricted to initially abstract objects is a full and faithful embedding. Moreover, each presheaf corresponding to an object of A corresponds to some initially abstract object, namely to $\mathrm{colim}\Delta_A$ – this is implied by (4.7). From all this the equivalence of categories under consideration follows. □

4.5.3 Partial algebras

In Lemma 2.4.17 in Section 2.4.2 we proved that observational equivalence of partial algebras is induced by a path subcategory consisting of only finite pre-free algebras. More precisely, the path subcategory contains those pre-free algebras whose carrier set is given by closure of a finite set of terms $\{t_1, \ldots, t_n\}$ with variables of observable sorts, under subterms, and whose operations are defined as in a term algebra. We noticed that the path subcategory can be restricted to only those pre-free algebras, for which terms $t_1, \ldots, t_n$ above are OBS-atomic, i.e., have no proper subterms of observable sorts besides variable occurrences (this means that the only subterms of observable sort of each term t_i are variable occurrences and possibly t_i itself). Let $\mathbb{T}_{OBS}$ denote in this subsection the full subcategory of such algebras in $PAlg_{\Sigma}^{\mathrm{w}}$.

In Lemma 4.5.6 below we characterize presheaves over $\mathbb{T}_{OBS}$ corresponding to partial algebras. Unlike in the case of standard algebras, when preservation of finite products is sufficient, we will require here preservation of certain kind of finite limits.

A homomorphism in $\mathbb{T}_{OBS}$ is called *var-to-var* if it maps, roughly, variables to variables. Formally, a morphism in $\mathbb{T}_{OBS}$ is var-to-var if it is in the image of some mapping in Set^{OBS} under the left pre-adjoint from Section 2.4.2. Consequently, a diagram in $\mathbb{T}_{OBS}$ is called var-to-var if all morphisms in this diagram are so. Moreover, a diagram in ${\mathbb{T}_{OBS}}^{op}$ is called var-to-var if this diagram, with direction of arrows reversed, is var-to-var in $\mathbb{T}_{OBS}$.

Lemma 4.5.6 *A presheaf* $F : {\mathbb{T}_{OBS}}^{op} \to Set$ *corresponds to a partial algebra (via* $\mathcal{B}eh_{\mathbb{T}_{OBS}}^{PAlg_{\Sigma}^{\mathrm{w}}}$*) iff it preserves finite var-to-var limits, i.e., if* F *transports each colimit cocone over a finite var-to-var diagram in* $\mathbb{T}_{OBS}$ *to a limit cone in* Set.

Below we give the proof of the above lemma. Note that equivalently, it would be sufficient to require F to preserve only terminal object (i.e., to be rooted) and var-to-var pullbacks.

Lemma 4.5.6 resembles the standard representation theorem stating that the category of partial algebras is equivalent to the category of finite limit preserving presheaves over the full subcategory of finitely presentable partial algebras (cf. for instance [AR94]). However, our situation is different at least in two respects. First, our result is parametric in OBS: objects in $\mathbb{T}_{OBS}$ are generated by its observable elements (in other words, $\mathbb{T}_{OBS}$ seen as a syntactic category contains term algebras over only observable variables). Second, all objects in our path category are pre-free, hence $\mathbb{T}_{OBS}$ is not closed under colimits; in particular, path objects satisfy no nontrivial equations, even in observable sorts.

Proof of Lemma 4.5.6: The *only if* part is immediate, since for each partial algebra, restriction of its contravariant hom-functor to a category P of finitely presentable algebras preserves finite limits (cf. [AR94], Sections 1.C and 3.D) and $\mathbb{T}_{OBS}$ is closed, as a subcategory of P, under var-to-var colimits.

For the *if* direction, assume that a presheaf F preserves finite var-to-var limits. A pre-free algebra induced by terms $t_1, \ldots, t_n$ in the way described above we denote by $[t_1, \ldots, t_n]$. Assume in the sequel that $\mathbb{T}_{OBS}$ is skeletal. All terms considered in the proof are OBS-atomic, hence we often do not mention it explicitly.

We construct a partial algebra A as a colimit (4.4) considered in Section 4.5.1; an explicit construction can be given as follows. For each observable sort $s \in OBS$, $|A|_s := F([x_s])$, where x_s is the single variable term of sort s. For technical reasons assume, without loss of generality, that sets $F([x_s])$ are pairwise disjoint. On non-observable sorts, the carrier set of A is a subset of a set of OBS-atomic terms over the carriers of A of observable sorts; that is, we consider terms whose variables are values of observable sorts of A. Namely, let $t \in |T_\Sigma(|A|_{OBS})|_s$ be such an OBS-atomic term of sort $s \notin OBS$ and let $a_1, \ldots, a_n$ denote variables of t. Let $[t']$ be an isomorphic representative of $[t]$ in $\mathbb{T}_{OBS}$, with variables $x_1, \ldots, x_n$, respectively. By $[x_i] \hookrightarrow [t']$ we mean the unique morphism $[x_s] \to [t']$ in $\mathbb{T}_{OBS}$ mapping $x_s \mapsto x_i$, where s is the sort of x_i. Now, t is in $|A|_s$ if there exists $\alpha \in F([t'])$ s.t. $F([x_i] \hookrightarrow [t'])(\alpha) = a_i$, for $i = 1, \ldots, n$.

Remark: Let $\alpha \in F([t'])$, for some $[t']$ in $\mathbb{T}_{OBS}$, where t' is a term of sort $s \notin OBS$ with variables $x_1, \ldots, x_n$; let $a_i := F([x_i] \hookrightarrow [t'])(\alpha)$. Whenever a_i are pair-wise distinct, α corresponds to some term $t \in |T_\Sigma(|A|_{OBS})|_s$ as above (recall that sets $F([x_s])$ are assumed to be pairwise disjoint).

An operation f of non-observable sort s is defined in A on $t_1, \ldots, t_k$ if the term $f(t_1, \ldots, t_k)$ is in $|A|$ (and $f^A(t_1, \ldots, t_k) = f(t_1, \ldots, t_k)$ then). When the result sort s of f is observable, its value on $t_1, \ldots, t_k$ should be in $|A|_s = F([x_s])$. Consider term $t = f(t_1, \ldots, t_k)$ and let $a_1, \ldots, a_n$ denote variables of t. Similarly as above, let $[t']$ be an isomorphic representative of $[t]$ in $\mathbb{T}_{OBS}$, with variables $x_1, \ldots, x_n$, respectively. Now, value of f on $t_1, \ldots, t_k$ is defined if there exists $\alpha \in F([t'])$ s.t. $F([x_i] \hookrightarrow [t'])(\alpha) = a_i$, for $i = 1, \ldots, n$. If it is so, we put $f^A(t_1, \ldots, t_k) := F(r : [x_s] \to [t'])(\alpha)$, where $r : x_s \mapsto t'$.

Remark above applies to observable sorts s as well. Furthermore, since F preserves finite var-to-var limits, we have:

Fact 4.5.7 *For a tuple of terms $t_1, \ldots, t_k$ with variables $a_1, \ldots, a_n \in |A|_{OBS}$, let $[t'_1, \ldots, t'_k]$ be an isomorphic representative of $[t_1, \ldots, t_k]$ in $\mathbb{T}_{OBS}$, with variables $x_1, \ldots, x_n$, respectively. We have that values of all t_i are defined in A if and only if there exists $\alpha \in F([t'_1, \ldots, t'_k])$ s.t. $F([x_i] \hookrightarrow [t'])(\alpha) = a_i$, for $i = 1, \ldots, n$.*

This follows easily form the observation that $[t'_1, \ldots, t'_k]$ is a colimit (in $\mathbb{T}_{OBS}$) of the following diagram (maps c_j form the colimit cocone):

$$\begin{tikzcd}
 & & & & {[t'_1, \ldots, t'_k]} & & & & \\
{[t''_1]} \arrow[urrrr, dotted, "c_1"] & & {[t''_2]} \arrow[urr, dotted, "c_2"] & & & & {[t''_3]} \arrow[ull, dotted, "c_3"'] & & {[t''_k]} \arrow[ullll, dotted, "c_k"'] \\
 & & & {[x, \ldots]} \arrow[ulll, "m_{1,3}"'] \arrow[urrr, "m_{3,1}"] & & & & \cdots & \\
 & {[x, \ldots]} \arrow[uul, "m_{1,2}"] \arrow[uur, "m_{2,1}"'] & & & & {[x, \ldots]} \arrow[uulll, "m_{2,3}"] \arrow[uur, "m_{3,2}"'] & & &
\end{tikzcd}$$

where $[t''_j]$ is an isomorphic representative of $[t_j]$ in $\mathbb{T}_{OBS}$ with variables x^j_i, say, instead of x_i. The diagram consists of homomorphisms $m_{j,k}$, $m_{k,j}$, for $k \neq j$, having as domains algebras induced by tuples of single variable terms. The role of $m_{j,k}$ and $m_{k,j}$ is to identify x^j_i with x^k_i, whenever both occur in t_j and t_k, respectively, i.e., whenever t_j and t_k essentially share variable x_i. Now, since F preserves finite var-to-var limits, there exists precisely one α in $F([t'_1, \ldots, t'_n])$ with $F(c_j)(\alpha) = \alpha_j$, where $\alpha_j \in F(t''_j)$ corresponds to t_j; hence α satisfies Fact 4.5.7.

Each $\alpha \in F([t'_1, \ldots, t'_k])$ induced by a tuple of terms $t_1, \ldots, t_k$ as in Fact 4.5.7 above is called *final*. For a given tuple $t_1, \ldots, t_k$, α is determined uniquely: since F necessarily transports var-to-var epis in $\mathbb{T}_{OBS}$ to monos, $F([x_1, \ldots, x_n] \hookrightarrow [t'_1, \ldots, t'_k])$ is mono; now, since $[x_1, \ldots, x_n]$ is a coproduct, hence transported by F to product, α is determined uniquely by a_i.

Remark: Observe that $\alpha \in F([t'_1, \ldots, t'_k])$, for any $[t'_1, \ldots, t'_k]$ in $\mathbb{T}_{OBS}$, is final precisely when a_i, $i = 1, \ldots, n$, defined by $a_i := F([x_i] \hookrightarrow [t'])(\alpha)$ as in Fact 4.5.7, are pair-wise distinct. This follows by the very definition of the carrier set of A and by the construction of α above. This fact will be crucial in the sequel.

Bijection: We need to show that there exists a bijection

$$b_{[t_1,\ldots,t_n]} : Hom_{PAlg^{\mathrm{w}}_{\Sigma}}([t_1, \ldots, t_n], A) \to F([t_1, \ldots, t_n]),$$

for each algebra $[t_1, \ldots, t_n]$ in $\mathbb{T}_{OBS}$, and that this bijection is natural w.r.t. $[t_1, \ldots, t_n]$. In the sequel we omit subscripts of b. The bijection b is defined as follows, for a given $v : [t_1, \ldots, t_n] \to A$. The homomorphism v factorizes uniquely through:

$$\begin{tikzcd} {[t_1, \ldots, t_n]} \arrow[r, "w"] \arrow[rr, bend right, "v"'] & {[u_1, \ldots, u_k]} \arrow[r, "s"] & A \end{tikzcd}\ ; \tag{4.9}$$

w is surjective and var-to-var, $u_1, \ldots, u_k$ are terms over $|A|_{OBS}$ and s stands for the evident valuation of terms u_i. Note that $[u_1, \ldots, u_k]$ is a homomorphic image of v only when the result sorts of all t_i are non-observable; otherwise, s identifies u_i with $v(t_j)$ whenever $w(t_j) = u_i$. The algebra $[u_1, \ldots, u_k]$ will be called an *atomic image* of v.

Let $\alpha \in F([u'_1, \ldots, u'_k])$ be a value corresponding to a tuple $u_1, \ldots, u_k$, as in Fact 4.5.7. Now, we put

$$b(v) := F(w')(\alpha),$$

where w' equals w composed with the isomorphism between $[u_1, \ldots, u_k]$ and $[u'_1, \ldots, u'_k]$.

For b to be a bijection, it is enough to show that for each $\beta \in F([t_1, \ldots, t_n])$ there exists precisely one final $\alpha \in F([u_1, \ldots, u_k])$ with $F(w)(\alpha) = \beta$, for a surjective and var-to-var $w : [t_1, \ldots, t_n] \to [u_1, \ldots, u_k]$.

Injectivity: Assume the contrary, i.e., that there are two final elements, $\alpha \in F([u_1, \ldots, u_k])$ and $\widetilde{\alpha} \in F([\widetilde{u_1}, \ldots, \widetilde{u_{\tilde{k}}}])$, s.t. $F(w)(\alpha) = F(\widetilde{w})(\widetilde{\alpha}) = \beta$, for some surjective var-to-var w and $\widetilde{w}$.

Since F transports epis to monos, $w \neq \widetilde{w}$. If both w and $\widetilde{w}$ were injective, we would have $w = w' = \mathrm{id}$. Thus at least one of w, $\widetilde{w}$ is not injective, say w. Taking a pushout (in

$\mathbb{T}_{OBS}$) of w and $\widetilde{w}$ we get v and $\widetilde{v}$, s.t. v is not injective:

$$
\begin{tikzcd}
& {} & \\
{} \arrow[ur, "\widetilde{v}"] & & {} \arrow[ul, "v"'] \\
& {} \arrow[ul, "w"] \arrow[ur, "\widetilde{w}"'] &
\end{tikzcd}
$$

Since F preserves var-to-var pullbacks, there exists some γ s.t. $F(v)(\gamma) = \widetilde{\alpha}$. But this is impossible, for non-injective v and final $\widetilde{\alpha}$ by the very definition, since this forces a collapse

$$a_i = F([x_i] \hookrightarrow [\widetilde{u_1}, \ldots, \widetilde{u_{\widetilde{k}}}]; v)(\gamma) = F([x_j] \hookrightarrow [\widetilde{u_1}, \ldots, \widetilde{u_{\widetilde{k}}}]; v)(\gamma) = a_j,$$

whenever $v(x_i) = v(x_j)$ (recall the remark below Fact 4.5.7).

Surjectivity: We should show that for each $\beta \in F([t_1, \ldots, t_n])$ there exists some final $\alpha \in F([u_1, \ldots, u_k])$ with $F(w)(\alpha) = \beta$, for a surjective var-to-var homomorphism $w : [t_1, \ldots, t_n] \to [u_1, \ldots, u_k]$. Let $x_1, \ldots, x_m$ denote variables occurring in $t_1, \ldots, t_n$ and let $a_i = F([x_i] \hookrightarrow [t_1, \ldots, t_n])(\beta)$, for $i = 1, \ldots, m$. We construct $[u_1, \ldots, u_k]$ as a coequalizer of p and q in $\mathbb{T}_{OBS}$:

$$
\begin{tikzcd}
{[x, \ldots]} \arrow[r, shift left, "p"] \arrow[r, shift right, "q"'] & {[t_1, \ldots, t_n]} \arrow[r, "w"] & {[u_1, \ldots, u_k]}
\end{tikzcd}
$$

where p, q are var-to-var and identify precisely those x_i, x_j with $a_i = a_j$. Since F preserves equalizers, there exists a final α with $F(w)(\alpha) = \beta$.

Naturality: We need to show that the following diagram commutes:

$$
\begin{tikzcd}
Hom_{PAlg^{\mathrm{w}}_{\Sigma}}([t_1, \ldots, t_n], A) \arrow[r, "b"] \arrow[d, "m;\_"'] & F([t_1, \ldots, t_n]) \arrow[d, "F(m)"] \\
Hom_{PAlg^{\mathrm{w}}_{\Sigma}}([\widetilde{t_1}, \ldots, \widetilde{t_{\widetilde{n}}}], A) \arrow[r, "b"] & F([\widetilde{t_1}, \ldots, \widetilde{t_{\widetilde{n}}}])
\end{tikzcd}
\tag{4.10}
$$

for any homomorphism $m : [\widetilde{t_1}, \ldots, \widetilde{t_{\widetilde{n}}}] \to [t_1, \ldots, t_n]$ in $\mathbb{T}_{OBS}$. Consider any homomorphism $v : [t_1, \ldots, t_n] \to A$ in $PAlg^{\mathrm{W}}_{\Sigma}$. Similarly as in (4.9), v and $(m; v)$ factor uniquely through their atomic images. Moreover, a unique homomorphism $\widetilde{m}$ between the atomic images is induced, making the following diagram commute:

$$
\begin{tikzcd}
{[t_1, \ldots, t_n]} \arrow[r, "w"] \arrow[rr, bend left, "v"] & {[u_1, \ldots, u_k]} \arrow[r] & A \arrow[d, equal] \\
{[\widetilde{t_1}, \ldots, \widetilde{t_{\widetilde{n}}}]} \arrow[u, "m"] \arrow[r, "\widetilde{w}"] \arrow[rr, bend right, "m;v"'] & {[\widetilde{u_1}, \ldots, \widetilde{u_{\widetilde{k}}}]} \arrow[u, hook, dotted, "\widetilde{m}"] \arrow[r] & A
\end{tikzcd}
$$

Transporting the inner left square (up to the unique isomorphic representatives in $\mathbb{T}_{OBS}$) via F we get a commuting square:

$$\begin{tikzcd} F([t_1,\ldots,t_n]) \arrow[d, "F(m)"'] & F([u'_1,\ldots,u'_k]) \arrow[l, "F(w')"'] \arrow[d, dotted, "F(\widetilde{m}')"] \\ F([\widetilde{t_1},\ldots,\widetilde{t_{\widetilde{n}}}]) & F([\widetilde{u_1}',\ldots,\widetilde{u_{\widetilde{k}}}']) \arrow[l, "F(\widetilde{w}')"'] \end{tikzcd} \tag{4.11}$$

To show that diagram (4.10) commutes, consider any final $\alpha \in F([u'_1,\ldots,u'_k])$ corresponding to v. Since $b(v) = F(w')(\alpha)$, we have

$$(F(m) \circ b)(v) = (F(m) \circ F(w'))(\alpha). \tag{4.12}$$

Further, we have $b(m;v) = F(\widetilde{w}')(\widetilde{\alpha})$, where a final $\widetilde{\alpha} \in F([\widetilde{u_1}',\ldots,\widetilde{u_{\widetilde{k}}}'])$ is determined by $(m;v)$. Since $(m;v)$ induces a final value uniquely, we conclude that

$$F(\widetilde{m}')(\alpha) = \widetilde{\alpha}.$$

This implies:

$$b(m;v) = (F(\widetilde{w}') \circ F(\widetilde{m}')(\alpha). \tag{4.13}$$

Since diagram (4.11) commutes, the right sides of (4.12) and (4.13) are equal. Hence the left sides are equal too, which proves that diagram (4.10) commutes. This completes the proof. □

4.6 Bisimulations

As usual in theory of coalgebras, by a *bisimulation* we mean a relation represented by a span of coalgebra morphisms [Rut96]. For $F_{\mathbb{P}}$, bisimulations have an explicit formulation, in analogy to strong bisimulations of transition systems:

Definition 4.6.1 *A* bisimulation *between two coalgebras $C = \langle S,t\rangle$ and $C' = \langle S',t'\rangle$ is any $\mathbb{P}$-sorted relation $R = \{R_P\}_{P\in|\mathbb{P}|} \subseteq S \times S'$ such that, if $(\alpha,\beta) \in R_P$ and $m : P \to Q$ in $\mathbb{P}$, then*

- *whenever $\alpha \overset{m}{\to} \alpha'$ in C, there exists some $\beta' \in S'$ with $\beta \overset{m}{\to} \beta'$ in C' and $(\alpha',\beta') \in R_Q$,*
- *whenever $\beta \overset{m}{\to} \beta'$ in C', there exists some $\alpha' \in S$ with $\alpha \overset{m}{\to} \alpha'$ in C and $(\alpha',\beta') \in R_Q$,*

One can easily see that each bisimulation, being a $|\mathbb{P}|$-sorted set, has a coalgebra structure (transitions are all $(\alpha,\beta) \overset{m}{\to} (\alpha',\beta')$ with $m : P \to Q$, when $(\alpha,\beta) \in R_P$, $(\alpha',\beta') \in R_Q$, $\alpha \overset{m}{\to} \alpha'$ and $\beta \overset{m}{\to} \beta'$) and together with two projections forms a span of coalgebra maps. We call a bisimulation R *consistent* whenever this coalgebra is so.

In [JNW93] a generalization of bisimulation was studied for an arbitrary category $\mathbb{M}$ (with a path subcategory $\mathbb{P}$), called there a *path bisimulation.* Similarly as there, we assume in this section that $\mathbb{P}$ and $\mathbb{M}$ have a common initial object.

Definition 4.6.2 ([JNW93]) *A* path bisimulation *wrt. $\mathbb{P}$ between two objects X,Y in $\mathbb{M}$ is a set R of pairs of computations in X and Y, respectively, with a common domain, such that:*

- *R contains the pair of unique computations ι_X, ι_Y from the initial object,*
- *for $(\alpha, \beta) \in R$ and a morphism m in $\mathbb{P}$,*
  - *if $\alpha = m; \alpha'$ then there exists some β' satisfying $\beta = m; \beta'$ and $(\alpha', \beta') \in R$.*
  - *if $\beta = m; \beta'$ then there exists some α' satisfying $\alpha = m; \alpha'$ and $(\alpha', \beta') \in R$.*

Intending to clarify a relationship between a path bisimulation and its coalgebraic counterpart, we derive the following equivalence:

Proposition 4.6.3 *For any two objects X, Y in $\mathbb{M}$, a ($|\mathbb{P}|$-sorted) relation R is a path bisimulation between X and Y iff it is a coalgebraic bisimulation between $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(X)$ and $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(Y)$ containing the pair (ι_X, ι_Y).*

Since a path bisimulation is in general too weak to force the existence of a span of open maps, in [JNW93] some strengthening of path bisimulation was proposed – it is called *strong* if it further satisfies:

- if $(\alpha, \beta) \in R$, with $\alpha : Q \to X$, $\beta : Q \to Y$ and $m : P \to Q$ in $\mathbb{P}$, then $(m; \alpha, m; \beta) \in R$.

Proposition 4.6.4 *A path bisimulation R between two objects X and Y in $\mathbb{M}$ is strong iff it is consistent.*

In other words, strong path bisimulations are precisely those path bisimulations which, equipped with a coalgebra structure and seen as a coalgebra, correspond to a presheaf.

Equivalently, one could only require reachability in Proposition 4.6.4, since the other two condition from Definition 4.2.3 always hold.

As a corollary, the following statements are equivalent:

- X and Y in $\mathbb{M}$ are strong path bisimilar,
- $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(X)$ and $\mathcal{B}eh_{\mathbb{P}}^{\mathbb{M}}(Y)$ are related by a nonempty consistent bisimulation,
- $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(X)$ and $\mathcal{Y}_{\mathbb{P}}^{\mathbb{M}}(X)$ are related by a span of open maps in the category of non-empty (or rooted) presheaves.

For $\mathbb{M}$ being the category of non-empty (or rooted) presheaves we conclude by Propositions 4.6.4 and 4.4.1 that strong path bisimilarity and open-map bisimilarity coincide – a fact proved already in [JNW93]. It is also the case in many concrete models, like event structures or transition systems. However, path bisimilarity and open-map bisimilarity (w.r.t. $\mathbb{T}_{OBS}$) *do not* coincide in the category of standard algebras – the rest of this section is devoted to a counterexample.

Example 4.6.5 *For a signature Σ of stacks, with usual operations push(_, _), pop(_), empty and top(_), let STACK_A and STACK_B denote two Σ-algebras of stacks with elements from A and B, respectively, for some two sets A and B of different infinite cardinality. The latter assumption guarantees that STACK_A and STACK_B are not observationally equivalent, w.r.t. to the single observable sort of stack elements. However, we will sketch a construction of a strong path bisimulation relating them. For the sake of completness assume that in both STACK_A and STACK_B it holds pop(empty) = empty. Moreover, let $a \in A$ and $b \in B$ denote values yielded by top(empty) is STACK_A and STACK_B, respectively.*

Recall from Section 4.5 that $\mathbb{T}_{OBS}$*-computations in an algebra are simply finite valuations. Hence a path bisimulation w.r.t.* $\mathbb{T}_{OBS}$ *contains pairs of valuations, which can be presented equivalently as pairs of finite sequences of elements of carrier sets of algebras under consideration (ordered by an arbitrary fixed enumeration of the set of variables). Now, we define a path bisimulation* $\sim$ *to contain all such pairs of sequences* $(a_1a_2\ldots a_n,\ b_1b_2\ldots b_n)$ *of the same length, satisfying the following condition:*

$$(\forall i,j.\ a_i = a_j \Leftrightarrow b_i = b_j) \quad \wedge \quad (\forall i. a_i = a \Leftrightarrow b_i = b) \tag{4.14}$$

(i.e., precisely the same equalities between elements hold in the two sequences and the same elements are the value of top(empty)).

The path bisimulation property instantiates in the case of algebras as follows. If $\sim$ *contains a pair* $(a_1a_2\ldots a_n,\ b_1b_2\ldots b_n)$*, then for each* n*-tuple of terms* $t_1,\ldots,t_n$ *from* $T_\Sigma(\{x_1,\ldots,x_k\})$*, for some* k*, and for each valuation* $v_A : \{x_1,\ldots,x_k\} \to |\mathrm{STACK}_A|$ *satisfying* $(t_i)_{\mathrm{STACK}_A[v_A]} = a_i$*, for* $1 \le i \le n$*, there exists some valuation* $v_B : \{x_1,\ldots,x_k\} \to |\mathrm{STACK}_B|$ *such that* $(t_i)_{\mathrm{STACK}_B[v_B]} = b_i$*, for* $1 \le i \le n$*, and moreover the pair*

$$(v_A(x_1)v_A(x_2)\ldots v_A(x_k),\ v_B(x_1)v_B(x_2)\ldots v_B(x_k))$$

is in $\sim$*. It can be easily checked that an appropriate valuation* v_B *can be chosen according to the following rules, for* $1 \le i \le k$*:*

- *if* $v_A(x_i) = a$*, put* $v_B(x_i) := b$*,*
- *if* $v_A(x_i) = a_j$*, for any* $1 \le j \le n$*, put* $v_B(x_i) := b_j$*,*
- *otherwise, choose* $v_B(x_i)$ *arbitrarily, but respecting the first part of condition (4.14).*

Moreover, $\sim$ *is strong, which is equivalent to the following condition. If* $\sim$ *contains a pair* $(a_1a_2\ldots a_n,\ b_1b_2\ldots b_n)$*, then for each* k*-tuple of terms* $t_1,\ldots,t_k \in T_\Sigma(\{x_1,\ldots,x_n\})$*, for some* k*,* $\sim$ *contains also the pair*

$$((t_1)_{A[v_A]}(t_2)_{A[v_A]}\cdots(t_k)_{A[v_A]},\ (t_1)_{B[v_B]}(t_2)_{B[v_B]}\cdots(t_k)_{B[v_B]});$$

valuations $v_A : \{x_1,\ldots,x_n\} \to |\mathrm{STACK}_A|$ *and* $v_B : \{x_1,\ldots,x_n\} \to |\mathrm{STACK}_B|$ *are defined by* $v_A(x_i) = a_i, v_B(x_i) = b_i$*, for* $1 \le i \le n$*.*

Roughly, from the above counterexample we can conclude that observational equivalence differs from the usual concept of bisimulation in one important respect: it requires a bijection between carrier sets of observable sorts.

Chapter 5

Regular algebras

In this chapter we study regular algebras, introduced in [Tiu78], and then intensively studied in early eighties, see e.g. [Tiu79, GP83] and literature mentioned there. Our starting point here is a more recent paper [BT95], studying observational equivalence of regular algebras and induced behavioural semantics of specification.

The aim of this chapter is two-fold. The most important topic is to analyze applicability of regular algebras as the models of behavioural specifications in the process of step-wise development of software systems. In addition, we study some mathematical properties of regular algebras, especially those related to observational equivalence, following the kind of the basic developments in the previous chapters.

Recall from Section 1.3 that regular algebras were defined by completeness and continuity conditions. In Section 5.1 we show that the continuity is equivalent to an apparently stronger condition, used in [BT95]. As the second step in definitional systematization of the field of regular algebras, we relate in Section 5.2 definition of regular algebras from Section 1.3 with another one, used in [Tiu78], and show them to be equivalent.

The latter result leads to quite a simple proof of an interesting fact, formulated in Section 5.3: observational indistinguishability in a regular algebra is induced by only finitary observable contexts. This does not extend, in general, to correspondences of regular algebras – we prove by a counterexample that an infinitary correspondence is indispensable to witness observational equivalence of two regular algebras.

The main subject of the other four sections are behavioural specifications in the setting of regular algebras. In Section 5.4 we explain how to combine, roughly speaking, regular algebras with standard algebras. More precisely, we extend the setting of regular algebras with a possibility to have carriers of some sorts "unordered" (these sorts are called non-regular or *algebraic*), which is necessary for practical examples. Technically, this is obtained by imposing an additional requirement on models of a specification: carrier sets of algebraic sorts are essentially flat. Moreover, we notice that when all observable sorts are algebraic, finitary correspondences are sufficient to prove observational equivalence of regular algebras.

Next, in Sections 5.5 and 5.6 we attack the problem of adaptation of the constructor behavioural implementation methodology [ST88, ST97] to the new framework. Intuitively, our main motivation was to enable finitary implementations of infinitary regular data structures. To this aim we introduce a method, which we call μ-induction, for defining fixed-points of recursors. At an intermediate step in constructor implementation we use pre-regular algebras, defined as algebras of a syntactical monad, which corresponds to an abstract (i.e., considered up to α-conversion) syntax. In Theorem 5.6.2 in Section 5.6 a condition sufficient for correctness of such implementation is given.

The general considerations are illustrated by few examples in Sections 5.5 and 5.7.

5.1 Continuity

Recall that regular Σ-algebras were defined in Section 1.3 as those ordered algebras which satisfy the completeness and continuity conditions. However, in [BT95] and [BT96] a stronger continuity condition was required in definition of regular algebras:

> (*strong continuity*): for all terms $t \in T^{\mu}_{\Sigma}(X \cup \{y : s\})_{s'}$, $q \in T^{\mu}_{\Sigma}(X \cup \{z : s\})_s$ and valuations $v : X \to |A|$, $t_{A[v_i]} \leq t_{A[v_{i+1}]}$, for $i \geq 0$, and $t_{A[v']} = \bigsqcup_{i\geq 0} t_{A[v_i]}$, where the valuation $v' : X \cup \{y : s\} \to |A|$ extends v by $v'(y) = (\mu z.q)_{A[v]}$ and for $i \geq 0$, $v_i : X \cup \{y : s\} \to |A|$ extends v by $v_i(z) = q^i_{A[v]}(\perp)$.

The only difference is that terms t are arbitrary regular terms in the strong continuity condition, while in Section 1.3 continuity was only required for terms t which are algebraic, i.e., do not contain symbol μ. To distinguish both definitions, continuity formulated in Section 1.3 will be called *weak* in the sequel.

As the main result of this section we show that both definitions are equivalent.

Theorem 5.1.1 *Weak continuity implies strong continuity (in an ordered algebra satisfying completeness condition).*

First we will need some some auxiliary lemmas; moreover, we will introduce a convenient notation to deal with finite approximations of regular terms (we need it also in the next section).

In the sequel, let A be an arbitrary fixed ordered Σ-algebra and let $s, s', s_1, s_2, \ldots$ denote arbitrary sorts from Σ.

Σ-algebraic and Σ-regular functions A mapping $f : |A|_{s_1} \times \ldots \times |A|_{s_n} \to |A|_s$ is called Σ-*regular* if it is defined by some regular term $t \in T^{\mu}_{\Sigma}(X \cup \{x_1 : s_1, \ldots, x_n : s_n\})_s$ and a valuation $v : X \to |A|$ in the following way:

$$f(a_1, \ldots, a_n) = t_{A[v']}, \text{ for } \quad a_1 \in |A|_{s_1}, \ldots, a_n \in |A|_{s_n}, \tag{5.1}$$

where v' extends v by $v'(x_1) = a_1, \ldots, v'(x_n) = a_n$, assumed that $t_{A[v']}$ is defined for all such v'. Function f defined in this way will be denoted by $\llcorner t_v(x_1, \ldots, x_n) \lrcorner$. When t is *algebraic*, i.e., $t \in T_{\Sigma}(X \cup \{x_1 : s_1, \ldots, x_n : s_n\})_s$, the function f is called Σ-*algebraic*. For any unary $f : |A|_s \to |A|_s$, by $f^{\omega}(\perp_s)$ we denote the least upper bound $\bigsqcup_{i\geq 0} f^i(\perp_s)$, if it exists; f^i denotes here a self-composition of f: $f^0 = \mathrm{id}_{A_s}$, $f^{i+1} = f \circ f^i$. The following lemma describes how Σ-regular functions are related to values of regular terms in A:

Lemma 5.1.2 *Assume that A satisfies completeness condition. For any Σ-regular term $t \in T^{\mu}_{\Sigma}(X \cup \{z : s\})_s$ and valuation $v : X \to |A|$, let $h : A_s \to A_s = \llcorner t_v(z) \lrcorner$; then*

- $h^i(\perp_s) = t^i_{A[v]}(\perp)$, *for $i \geq 0$,*
- $h^{\omega}(\perp_s)$ *exists and is equal to* $(\mu z.t)_{A[v]}$.

For any mapping $f : |A|_{s_1} \times \ldots \times |A|_{s_n} \to |A|_s$, by $f[a_1, \ldots, a_k]$ (for some $k < n$ and $a_1 \in |A|_{s_1}, \ldots, a_k \in |A|_{s_k}$) we mean a function $|A|_{s_{k+1}} \times \ldots \times |A|_{s_n} \to |A|_s$ defined by

$$f[a_1, \ldots, a_k](a_{k+1}, \ldots, a_n) = f(a_1, \ldots, a_n).$$

Whenever f is Σ-regular, $f[a_1, \ldots, a_k]$ is Σ-regular as well, by an appropriate extension of a valuation.

Monotonicity and continuity In the sequel, unless stated otherwise, $i \geq 0$ is understood as $\omega > i \geq 0$. A crucial observation is that in a regular algebra, a partial order $\leq$ relates some "inessential" pairs, i.e., not relevant for completeness or continuity condition. We rule out some of them, by restricting to the greatest sub-order $\sqsubseteq$ of $\leq$ preserved by all Σ-algebraic mappings. For $a_1, a_2 \in |A|_s$, let $\sqsubseteq_s$ be a partial order defined by:

$$a_1 \sqsubseteq_s a_2 \text{ iff for all } \Sigma\text{-algebraic mappings } f : A_s \to A_{s'},\ f(a_1) \leq_{s'} f(a_2).$$

When A is regular, $a_1 \sqsubseteq_s a_2 \Rightarrow a_1 \leq_s a_2$; it also holds then $\perp_s \sqsubseteq_s a$, for any $a \in |A|_s$. In the sequel we use a concise notation for least upper bounds preserved by algebraic mappings:

Definition 5.1.3 *For a sequence $\{a_i\}_{i \geq 0}, a_i \in |A|_s$, by $\{a_i\}_{i\geq 0} \Rrightarrow a$ we mean that*

- *for each Σ-algebraic $f : |A|_s \to |A|_{s'}$, $\{f(a_i)\}_{i \geq 0}$ is a chain w.r.t. $\leq$ with the l.u.b. $f(a)$.*

Note that $\{a_i\}_{i\geq 0} \Rrightarrow a$ implies necessarily that $\{a_i\}_{i \geq 0}$ is a chain w.r.t. $\leq$ with the l.u.b. a; moreover $a_i \sqsubseteq_s a$, for all $i \geq 0$ and a is necessarily the l.u.b. of $\{a_i\}_{i\geq 0}$ w.r.t. $\sqsubseteq$ as well. If A is regular, by weak continuity $\{t^i_{A[v]}(\perp)\}_{i\geq 0} \Rrightarrow (\mu z.t)_{A[v]}$; or equivalently $\{h^i(\perp_s)\}_{i \geq 0} \Rrightarrow h^\omega(\perp_s)$, for any $t \in T^\mu_\Sigma(X \cup \{z : s\})_s$ and $h = \llcorner t_v(z : s) \lrcorner$.

The following fact enables us to calculate "two-dimensional" least upper bounds in an ordered Σ-algebra A:

Lemma 5.1.4 *Let $\{a_{i,j} \in |A|_s\}_{i,j\geq 0}$ satisfy $\{a_{i,j}\}_{i\geq 0} \Rrightarrow a_{\omega,j}$, for each $j \geq 0$ and symmetrically $\{a_{i,j}\}_{j\geq 0} \Rrightarrow a_{i,\omega}$, for each $i \geq 0$. Then*

(1) *the following equivalence holds, for each $a \in |A|_s$:*

$$\{a_{i,\omega}\}_{i\geq 0} \Rrightarrow a \quad \textit{iff} \quad \{a_{\omega,j}\}_{j \geq 0} \Rrightarrow a;$$

(2) *whenever any of two equivalent conditions in* (1) *holds, then also $\{a_{i,i}\}_{i\geq 0} \Rrightarrow a$.*

Proof: Routine. □

Definition 5.1.5 (Algebraic continuity) *We say that $f : |A|_{s_1} \times \ldots \times |A|_{s_n} \to |A|_s$ is* algebraically continuous *if it preserves all those least upper bounds which are preserved by all algebraic mappings. Formally,*

$$\{a^1_i\}_{i\geq 0} \Rrightarrow a^1_\omega, \ldots, \{a^n_i\}_{i \geq 0} \Rrightarrow a^n_\omega \quad \textit{implies} \quad \{f(a^1_i, \ldots, a^n_i)\}_{i\geq 0} \Rrightarrow f(a^1_\omega, \ldots, a^n_\omega). \tag{5.2}$$

The following lemma, crucial for the proof of Theorem 5.1.1, implies that in a regular algebra all Σ-regular mappings are algebraically continuous:

Lemma 5.1.6 (Algebraic continuity) *Let A be a regular Σ-algebra.*

(1) Given algebraically continuous Σ-regular mappings $f_1, \dots, f_k$, where for $i = 1, \dots, k$, $f_i : |A|_{s_1} \times \dots \times |A|_{s_n} \to |A|_{s'_i}$, and algebraic k-ary $g : |A|_{s'_1} \times \dots \times |A|_{s'_k} \to |A|_s$, the function f defined by a composition

$$f(a_1, \dots, a_n) = g(f_1(a_1, \dots, a_n), \dots, f_k(a_1, \dots, a_n))$$

is also algebraically continuous.

(2) For any algebraically continuous Σ-regular $g : |A|_{s_1} \times \dots \times |A|_{s_n} \times |A|_s \to |A|_s$, the "fix-point" function f given by $f(a_1, \dots, a_n) = g[a_1, \dots, a_n]^\omega(\perp_s)$, if defined, is also algebraically continuous.

Proof: *(1)* By induction on k we will show that

$$\{a_i^1\}_{i\geq 0} \Rrightarrow a_\omega^1, \dots, \{a_i^k\}_{i\geq 0} \Rrightarrow a_\omega^k \quad \text{implies} \quad \{g(a_i^1, \dots, a_i^k)\}_{i\geq 0} \Rrightarrow g(a_\omega^1, \dots, a_\omega^k),$$

which is sufficient for *(1)*. The case $k = 1$ is straightforward by the very definition of $\Rrightarrow$. For the induction step, put

$$b_{i,j} := g[a_i^1](a_j^2, \dots, a_j^k), \quad \text{for } \omega \geq i, j \geq 0.$$

Observe that for each $\omega \geq i \geq 0$, by the induction assumption,

$$\{g[a_i^1](a_j^2, \dots, a_j^k)\}_{j\geq 0} \Rrightarrow g[a_i^1](a_\omega^2, \dots, a_\omega^k),$$

that is $\{b_{i,j}\}_{j\geq 0} \Rrightarrow b_{i,\omega}$. Moreover, also $\{b_{i,j}\}_{i\geq 0} \Rrightarrow b_{\omega,j}$, for each $\omega \geq j \geq 0$ (this is the case $k = 1$ applied for the mapping $a \mapsto g[a](a_j^2, \dots, a_j^k)$). Hence, by Lemma 5.1.4(2) we obtain $\{b_{i,i}\}_{i\geq 0} \Rrightarrow b_{\omega,\omega}$, which completes the proof.

(2) We will show (5.2) using Lemma 5.1.4. Let

$$b_{i,j} := g[a_j^1, \dots, a_j^n]^i(\perp_s), \quad \text{for } \omega \geq i, j \geq 0.$$

Since A is regular and g is Σ-regular, we conclude by Lemma 5.1.2 that $\{b_{i,j}\}_{i\geq 0} \Rrightarrow b_{\omega,j}$, for each $\omega \geq j \geq 0$. Moreover, by the assumption that g is algebraically continuous, we conclude that $\{b_{i,j}\}_{j\geq 0} \Rrightarrow b_{i,\omega}$, for each $\omega > i \geq 0$. Now, applying Lemma 5.1.4(1) we obtain $\{b_{\omega,j}\}_{j\geq 0} \Rrightarrow b_{\omega,\omega}$, which implies (5.2). □

Proof of Theorem 5.1.1: Using Lemma 5.1.6 we will show inductively that all Σ-regular functions in a regular algebra A are algebraically continuous. This, due to a correspondence given by Lemma 5.1.2, implies strong continuity.

Formally, we show that for each term $t \in |T_\Sigma^\mu(X \cup \{x'_1 : s'_1, \dots, x'_k : s'_k\})|_s$ and valuation $v : X \to |A|$, for *any sequences* $\{q_j^1\}_{j\geq 0}, \dots, \{q_j^k\}_{j\geq 0}$ satisfying

$$\{q_j^i\}_{j\geq 0} \Rrightarrow q_\omega^i, \text{ for } 1 \leq i \leq k, \tag{5.3}$$

it holds

$$\{t_{A[v_j]}\}_{j\geq 0} \Rrightarrow t_{A[v_\omega]}, \tag{5.4}$$

where $v_\omega : X \cup \{x'_1 : s'_1, \ldots, x'_k : s'_k\} \to |A|$ extends v by $v_\omega(x'_i) = q^i_\omega$, for $1 \le i \le k$; and for $j \ge 0$, $v_j : X \cup \{x'_1 : s'_1, \ldots, x'_k : s'_k\} \to |A|$ extends v by $v_j(x'_i) = q^i_j$, for $1 \le i \le k$. The proof is by structural induction on t (accompanied with an arbitrary valuation v).

This is more general than strong continuity in two respects. First, we consider here a finite number of sequences $\{q^1_j\}_{j\ge 0}, \ldots, \{q^k_j\}_{j\ge 0}$ instead of only one $\{q_j\}_{j\ge 0}$. Second, these sequences need not be approximation chains of Σ-regular μ-terms – the only assumption needed is (5.3).

As the base of induction, i.e., for t being a constant or a single variable occurrence, (5.4) is obvious. For the induction step, we fix $t \in |T^\mu_\Sigma(X)|_s$ and valuation $v : X \to |A|$ and assume that

$$\textit{(5.4) holds for all proper subterms of } t \textit{ and all appropriate valuations.} \tag{5.5}$$

There are two cases, and Lemma 5.1.6 can be applied to both of them. When $t = f(t_1, \ldots, t_n)$, for some operation f in Σ, (5.4) follows by Lemma 5.1.6(1). Otherwise, when $t = \mu x.t'$ is a μ-term, (5.4) follows by Lemma 5.1.6(2). □

5.2 Relating definitions

The category of regular algebras was for the first time studied in [Tiu78]. Afterwards, much work was done on regular algebras, e.g. recently they were studied in the contexts of observational algebraic specifications [BT95]. Definitions of a regular algebra considered in these both paper, although intuitively similar, are technically different. As a main result of this section, we show in Theorem 5.2.2 that they coincide. To distinguish both definition, regular algebras from [Tiu78] are called *T-regular* in the sequel.

Throughout this section we fix a one-sorted signature Σ, hence we omit sort names. Restricting to single sorted algebras makes the presentation easier and extending the result to many-sorted case is routine.

An ordered Σ-algebra A with the least element $\bot \in |A|$ is *strict* (w.r.t. $\le$) if $\bot \sqsubseteq a$ (i.e., $f(\bot) \le f(a)$, for every Σ-algebraic f), for every $a \in |A|$. Let $f_1, \ldots, f_n : |A|^n \to |A|$ be n-ary Σ-algebraic mappings. In the sequel we extend some notation from the previous section to n-ary functions $\langle f_1, \ldots, f_n\rangle : |A|^n \to |A|^n$. Namely, we consider sequences $\{\langle f_1, \ldots, f_n\rangle^k(\bot, \ldots, \bot) \in |A|^n\}_{k\ge 0}$, obtained by iteration of $\langle f_1, \ldots, f_n\rangle$, and the least upper bounds $\langle f_1, \ldots, f_n\rangle^\omega(\bot, \ldots, \bot)$ w.r.t. the point-wise order, if existing.

Definition 5.2.1 (T-regular algebras, [Tiu78]) *An ordered Σ-algebra A is* T-regular *if*

1. *it is strict,*

2. *for all Σ-algebraic mappings $f_1, \ldots, f_n : |A|^n \to |A|$, $\{\langle f_1, \ldots, f_n\rangle^k(\bot, \ldots, \bot)\}_{k\ge 0}$ has a l.u.b. $\langle f_1, \ldots, f_n\rangle^\omega(\bot, \ldots, \bot)$, which is moreover a fixed point of $\langle f_1, \ldots, f_n\rangle$.*

Definition of T-regular algebras emphasizes fixed points, while definition of regular algebras concentrates on continuity. To relate these essentially different definitions, we will intensively use the notation $\_ \Rrightarrow \_$, introduced in Definition 5.1.3. By now, we only mention that in a regular algebra a least upper bound of an approximation chain is necessarily a fixed point of the associated Σ-regular mapping, analogously to the T-regular algebras. Moreover, all regular algebras are necessarily strict. On the other hand, in a T-regular algebra an analogue of continuity holds.

The rest of this section is devoted to the proof of the fact that definition 5.2.1 is equivalent to the definition of regular algebras from Section 1.3 by the completeness and continuity conditions:

Theorem 5.2.2 *Regular Σ-algebras coincide with T-regular Σ-algebras.*

Syntactical restriction By a μ-term (μ-subterm) we mean a term (subterm) of the form $\mu x.t'$. For any regular term t, $[t]_\mu$ denotes the term obtained from t by replacing each μ-subterm of the form $\mu x'.t'$ by the variable x'. Without loss of generality, we consider in this section only regular Σ-terms t satisfying the following restriction (such terms are called *separable*): if the same variable x is bound in two different subterms $\mu x.t_1$ and $\mu x.t_2$ of t, then these subterms are disjoint and terms $[t_1]_\mu$ and $[t_2]_\mu$ are identical.

Let A be an arbitrary T-regular algebra in the following. We will also make some simplifying assumptions concerning the algebraic mappings $\langle f_1, \ldots, f_n \rangle$ in Definition 5.2.1. Functions $f_1, \ldots, f_n$ are defined by $f_i = \llcorner (t_i)_v(x_1, \ldots, x_n) \lrcorner$, for some valuation $v : X \to |A|$ and some algebraic terms $t_1, \ldots, t_n \in T_\Sigma(X \cup \{x_1, \ldots, x_n\})$. Consider a system of equations $\mathcal{U}$:

$$\begin{array}{rcl} x_1 & = & t_1, \\ \ldots & = & \ldots \\ x_n & = & t_n. \end{array} \tag{5.6}$$

$\langle f_1, \ldots, f_n \rangle^\omega(\bot, \ldots, \bot)$ is the least solution of $\mathcal{U}$. Let a quasi-order $\succeq$ be a reflexive-transitive closure of a binary relation $\succ^1$ on $\mathbb{I} = \{1, \ldots, n\}$ defined by $i \succ^1 j$ if x_j appears in term t_i. (Hence $i \succeq j$ means that x_i "depends" on x_j.) We call a system $\mathcal{U}$ *decomposable* if for each $i \in \mathbb{I}$, $\mathbb{I}_i = \{j : i \succeq j\} \neq \mathbb{I}$ (i.e., no element depends on all others). Clearly, $i \succeq j \Leftrightarrow \mathbb{I}_j \subseteq \mathbb{I}_i$.

Observation 5.2.3 *Assume that $\mathcal{U}$ is decomposable and let i be maximal w.r.t. $\succeq$, $1 \leq i \leq n$. Let $\mathcal{U}_1$ and $\mathcal{U}_2$ denote two subsystems of $\mathcal{U}$, containing equations indexed by $\mathbb{I}_i$ and $\cup_{j \notin \mathbb{I}_i} \mathbb{I}_j$, respectively. Then the least solution of $\mathcal{U}$ in a T-regular algebra is obtained by appropriately merging the least solutions of $\mathcal{U}_1$ and $\mathcal{U}_2$.*

Restriction in definition of T-regular algebras Due to the above observation, we assume that an algebraic mapping $\langle f_1, \ldots, f_n \rangle$ in Definition 5.2.1 is defined by some non-decomposable system of equations of the form (5.6).

In the proof of Theorem 5.2.2 we will rely on the following correspondence between non-decomposable systems of equations and separable regular μ-terms.

From equations to a term For an arbitrary non-decomposable system $\mathcal{U}$ of equations (5.6) we define a regular μ-term t. Start by taking $t := x_i$, for some i satisfying $\mathbb{I}_i = \mathbb{I}$. Then repeat, until no change of t is possible: if an unbound variable x_k occurs in t, then replace this occurrence of x_k by a subterm $\mu x_k.t_k$.

From a term to equations Let $t = \mu x_1.t_1$ be a regular separable μ-term. We define a system of equations by induction on the number of different variables bound by μ in t, denoted by $|t|_\mu$. If t_1 contains no μ, then put $\mathcal{U} := \{x_1 = t_1\}$. Otherwise, let $\mu x_2.t_2, \ldots, \mu x_k.t_k$ be all maximal μ-subterms of t', and let $\mathcal{U}_i$ be a system of equations for $\mu x_i.t_i$, $i = 2, \ldots, k$. We define

$$\mathcal{U} \; := \; (\bigcup_{2 \leq i \leq k} \mathcal{U}_i) \;\; \cup \;\; \{x_1 = [t_1]_\mu\}.$$

Two transformations described above take non-decomposable systems of equations $\mathcal{U}$ to separable regular μ-terms t and vice versa. A term t determines system $\mathcal{U}$ uniquely, while $\mathcal{U}$ induces a term t ambiguously, depending on the choice of $i \in \mathbb{I}$ satisfying $\mathbb{I}_i = \mathbb{I}$. The number of equations in $\mathcal{U}$ is equal to $|t|_\mu$.

To prove Theorem 5.2.2, we will show, roughly speaking, that in a strict ordered algebra the following are equivalent:

- definedness of a separable μ-term t under valuation v,
- existence of the l.u.b. of $\{\langle f_1, \ldots, f_n\rangle^k(\bot, \ldots, \bot)\}_{k\geq 0}$ for $\langle f_1, \ldots, f_n\rangle$ defined by the non-decomposable system $\mathcal{U}$ of equations (5.6) corresponding to t and valuation v.

Moreover, we will show that the values of least upper bounds in both cases coincide.

An appropriate formalization of this equivalence is spelled out in Definition 5.2.4. It introduces, together with Definition 5.2.5, some notation, useful in the proof of Lemma 5.2.7 and Theorem 5.2.2. Lemma 5.2.7 is of crucial importance here, as it plays a central role in the proof of Theorem 5.2.2.

Let A denote an arbitrary strict ordered Σ-algebra in definitions below. We reserve names $x_1, x_2, \ldots$ for variables bound by μ, like in (5.6).

The notation $\_ \Rrightarrow \_$ was introduced by now only for sequences of elements of $|A|$. We extend below this notation for $|A|^k$, for $k \geq 1$; i.e., in Definition 5.1.3, $\leq$ should be understood as the point-wise order in $|A|^k$ and f is a Σ-algebraic function $|A|^k \to |A|$. Note that Lemma 5.1.4 still holds for such a generalized meaning of $\_ \Rrightarrow \_$.

Definition 5.2.4 (k-equivalence) *Let $t = \mu x_1.t_1 \in T^\mu_\Sigma(X)$ denote an arbitrary separable μ-term such that $|t|_\mu = k$ and let $v : X \to |A|$ be a valuation. Let $f_i : |A|^k \to |A|$, $1 \leq i \leq k$, denote Σ-algebraic functions defined by v and the system of equations corresponding to t. Let $\mu x_i.t_i$, $1 \leq i \leq k$, denote all different μ-subterms of t.*

We say that A satisfies k-equivalence if for each t and v as above, the following conditions are equivalent, for any $g = \langle g_1, \ldots, g_k\rangle \in |A|^k$:

1. $\{\langle f_1, \ldots, f_k\rangle^i(\bot, \ldots, \bot)\}_{i\geq 0} \Rrightarrow g$,
2. *$(\mu x_i.t_i)_{A[v_i]}$ is defined and $\{(t_i)^j_{A[v]}(\bot)\}_{j\geq 0} \Rrightarrow g_i = (\mu x_i.t_i)_{A[v_i]}$, for $i = 1, \ldots, k$, where v_i extends v by $v_i(x_l) = g_l$, for all $l \preceq i$ such that x_l appears freely in t_i.*

Observe that 2 implies that $(\mu x_1.t_1)_{A[v]}$ is defined and equal g_1, since no variable x_l appears freely in t_1.

Definition 5.2.5 (k-definedness) *We say that A satisfies k-definedness if for each t and v as in Definition 5.2.4, one of conditions 1, 2 is satisfied, for some $g \in |A|^k$.*

Lemma 5.2.6 (Monotonicity) *In any strict ordered Σ-algebra A, for any Σ-algebraic mapping $f : |A|_{s_1} \times \ldots \times |A|_{s_n} \to |A|_s$, $n \geq 1$,*

$$a_1 \sqsubseteq_{s_1} a'_1, \ldots, a_n \sqsubseteq_{s_n} a'_n \Longrightarrow f(a_1, \ldots, a_n) \sqsubseteq_s f(a'_1, \ldots, a'_n),$$

Proof: Routine induction on n. $\square$

Lemma 5.2.7 *In any strict ordered Σ-algebra A, for each $k \geq 2$,*

$$l\text{-equivalence and } l\text{-definedness (for all } l < k) \quad \text{implies} \quad k\text{-equivalence.}$$

Proof: Fix k. Let $t = \mu x_1.t_1 \in T^{\mu}_{\Sigma}(X)$ be a separable μ-term such that $|t|_\mu = k$ and let $v : X \to |A|$ be some valuation. Let $\{f_i : |A| \to |A|\}_{1 \leq i \leq k}$ be Σ-algebraic functions defined by v and the system of equations corresponding to t. Let $\mu x_2.t_2, \ldots, \mu x_n.t_n$ be maximal μ-subterms of t_1. For simplicity of presentation, assume that there exists precisely one such term, i.e., $n = 2$; for $n > 2$ the proof proceeds similarly, due to Observation 5.2.3. The number $l := |\mu x_2.t_2|_\mu$ of variables bound in $\mu x_2.t_2$ is less than k.

We are going to apply Lemma 5.1.4. To this aim we define $b^j_i \in |A|$, $a^j_i \in |A|^{k-1}$, for $0 \leq i < \omega, 0 \leq j \leq \omega$, as follows. For each $j < \omega$:

- $b^j_0 := \bot$, $a^j_0 := \bot$,
- $a^j_{i+1} := \langle f_2, \ldots, f_k\rangle[b^j_i]^j(\bot, \ldots, \bot)$, and $b^j_{i+1} := f_1(b^j_i, a^j_{i+1})$, where

$$\langle f_2, \ldots, f_k\rangle[b^j_i] \text{ is a function } |A|^{k-1} \to |A|^{k-1}.$$

Moreover, for $j = \omega$ we put analogously

- $b^\omega_0 := \bot$, $a^\omega_0 := \bot$,
- $a^\omega_{i+1} := \langle f_2, \ldots, f_k\rangle[b^\omega_i]^\omega(\bot, \ldots, \bot)$, and $b^\omega_{i+1} := f_1(b^\omega_i, a^\omega_i)$.

In the sequel $\pi^r_i : |A|^r \to |A|$, $1 \leq i \leq r$, denotes the projection on i-th coordinate. By the assumption on l-equivalence and l-definedness for all $l < k$ (although when assumed $n = 2$, we only need $(k-1)$-definedness and $(k-1)$-equivalence), we conclude that a^ω_i, $i \geq 0$ is well-defined and that $\pi^{k-1}_1(a^\omega_{i+1}) = (\mu x_2.t_2)_{A[v']}$ under a valuation v' extending v by $v'(x_1) = b^\omega_i$. Hence, clearly $b^\omega_i = (\mu x_1.t_1)^i_{A[v]}(\bot)$, for $i < \omega$, while b^j_i $(j < \omega)$ represent some kind of finitary approximations of $(\mu x_1.t_1)^i_{A[v]}(\bot)$.

Denote by $c_i := \langle f_1, \ldots, f_k\rangle^i(\bot, \ldots, \bot) \in |A|^k$, for $0 \leq i < \omega$. It is sufficient to prove:

$$\{c_i\}_{i\geq 0} \Rrightarrow g \quad \Leftrightarrow \quad \{(b^\omega_i, a^\omega_i)\}_{i \geq 0} \Rrightarrow g \quad (\text{for each } g \in |A|^k). \tag{5.7}$$

This is sufficient due to the $(k-1)$-equivalence for $\mu x_2.t_2$ together with a valuation v' extending v by $v'(x_1) = g_1$ (and the induced Σ-algebraic functions $f_i[g_1] : |A|^{k-1} \to |A|$, $i = 2, \ldots, k$).

A point-wise order on $|A|^k$ induced by $\sqsubseteq$ will be also denoted by $\sqsubseteq$ below – this is justified by Lemma 5.2.6. Observe that the following inequalities hold by Lemma 5.2.6, since A is strict and satisfies $(k-1)$-definedness:

$$c_i \sqsubseteq c_{i+1}, \quad (b^j_i, a^j_i) \sqsubseteq (b^j_{i+1}, a^j_{i+1}), \quad (b^j_i, a^j_i) \sqsubseteq (b^{j+1}_i, a^{j+1}_i) \sqsubseteq (b^\omega_i, a^\omega_i),$$

for each $i, j \geq 0$ – they can be proved by easy inductions on i or j.

Before attacking (5.7), we need some auxiliary facts. First, for each $0 \leq j < \omega$, $\{c_i\}_{i\geq 0}$ bounds $\{(b^j_i, a^j_i) \in |A|^k\}_{i \geq 0}$ in the following sense:

$$\forall i \geq 0 \;\; \exists\, i' \geq i \;\; \text{such that} \;\; (b^j_i, a^j_i) \sqsubseteq c_{i'}. \tag{5.8}$$

This can be proved easily (for a fixed j) by induction on i. Obviously $(b_0^j, a_0^j) \sqsubseteq c_0$. Assumed $(b_i^j, a_i^j) \sqsubseteq c_{i'}$ (for some i'), we have the following inequalities, by multiple application of Lemma 5.2.6:

$$a_{i+1}^j = \langle f_2, \ldots, f_k \rangle [b_i^j]^j (\perp, \ldots, \perp) \sqsubseteq \langle f_2, \ldots, f_k \rangle [\pi_1^k(c_{i'})]^j (\perp, \ldots, \perp) \sqsubseteq \pi_{2\ldots k}^k(c_{i'+j}),$$

$$b_{i+1}^j = f_1(b_i^j, a_{i+1}^j) \sqsubseteq f_1(\pi_1^k(c_i), \pi_{2\ldots k}^k(c_{i'+j})) \sqsubseteq f_1(\pi_1^k(c_{i'+j}), \pi_{2\ldots k}^k(c_{i'+j})) = \pi_1^k(c_{i'+j+1}),$$

hence $(b_{i+1}^j, a_{i+1}^j) \sqsubseteq c_{i'+j+1}$, which completes the induction step for (5.8).

On the other hand, $\{c_i\}_{i\geq 0}$ is bounded from another side by $\{(b_i^\omega, a_i^\omega)\}_{i\geq 0}$:

$$\forall i \geq 0 \;\; c_i \sqsubseteq (b_i^\omega, a_i^\omega). \tag{5.9}$$

Similarly, induction on i gives an argument. Obviously $c_0 \sqsubseteq (b_0^\omega, a_0^\omega)$. Assume that $c_i \sqsubseteq (b_i^\omega, a_i^\omega)$ (for a fixed i). We have the following inequalities, following similarly as before from Lemma 5.2.6:

$$\pi_{2\ldots k}^k(c_{i+1}) \sqsubseteq \langle f_2, \ldots, f_k \rangle [b_i^\omega]^{i+1} (\perp, \ldots, \perp) \sqsubseteq \langle f_2, \ldots, f_k \rangle [b_i^\omega]^\omega (\perp, \ldots, \perp) = a_{i+1}^\omega,$$

$$\pi_1^k(c_{i+1}) = f_1(c_i) \sqsubseteq f_1(b_i^\omega, \pi_{2\ldots k}^k(c_{i+1})) \sqsubseteq f_1(b_i^\omega, a_{i+1}^\omega) = b_{i+1}^\omega,$$

hence $c_{i+1} \sqsubseteq (b_{i+1}^\omega, a_{i+1}^\omega)$, which completes the induction step for (5.9).

Second, aiming at applying Lemma 5.1.4, we need to prove the following:

(a) $\{(b_i^j, a_i^j)\}_{i\geq 0} \Rrightarrow (b_\omega^j, a_\omega^j)$, for some $b_\omega^j \in |A|$, $a_\omega^j \in |A|^{k-1}$ (for each $0 \leq j < \omega$),

(b) $\{(b_i^j, a_i^j)\}_{j\geq 0} \Rrightarrow (b_i^\omega, a_i^\omega)$, for each $0 \leq i < \omega$.

We start by showing *(a)*. Observe that for a fixed j,

$$\{b_i^j\}_{i\geq 0} \Rrightarrow b_\omega^j, \quad \text{for some } b_\omega^j, \tag{5.10}$$

follows immediately from 1-definedness, since $b_{i+1}^j = f_1(b_i^j, \langle f_2, \ldots, f_k \rangle [b_i^j]^j (\perp, \ldots, \perp))$, and $x \mapsto \langle f_2, \ldots, f_k \rangle [x]^j (\perp, \ldots, \perp)$ determines a Σ-algebraic mapping $g : |A| \to |A|$. Moreover, exploiting further this mapping, we obtain

$$\{a_i^j\}_{i\geq 0} \Rrightarrow a_\omega^j := g(b_\omega^j) \tag{5.11}$$

as an easy corollary from (5.10), since $a_{i+1}^j = g(b_i^j)$. One can easily check that (5.10) and (5.11) imply *(a)*.

Now we are going to show *(b)*, by induction on i.

For $i = 0$ *(b)* holds; assume *(b)* for some $i \geq 0$. Put $d_j^0 := \perp$, for $0 \leq j \leq \omega$, and

$$d_j^m := \langle f_2, \ldots, f_k \rangle [b_i^j]^m (\perp, \ldots, \perp) \in |A|^{k-1}, \quad \text{for all } 0 \leq j \leq \omega,\ 0 < m < \omega.$$

By $(k-1)$-definedness we have, for each $j \leq \omega$, $\{d_j^m\}_{m\geq 0} \Rrightarrow d_j^\omega$, for some $d_j^\omega \in |A|$. On the other hand we have (analogously as in the proof of *(a)*), for each $m < \omega$, $\{d_j^m\}_{j\geq 0} \Rrightarrow d_\omega^m$ Hence we can apply Lemma 5.1.4(2) to obtain $\{d_j^j\}_{j\geq 0} \Rrightarrow a_{i+1}^\omega$ that is

$$\{a_{i+1}^j\}_{j\geq 0} \Rrightarrow a_{i+1}^\omega.$$

Now $\{b_{i+1}^j\}_{j\geq 0} \Rrightarrow b_{i+1}^\omega$ follows easily and we finally conclude

$$\{(b_{i+1}^j, a_{i+1}^j)\}_{j\geq 0} \Rrightarrow (b_{i+1}^\omega, a_{i+1}^\omega),$$

similarly like in the proof of *(a)*.

By bounds (5.8) and (5.9) and by *(a)* we conclude that for each Σ-algebraic function $h : |A|^k \to |A|$,

$$\bigsqcup_{i\geq 0} h(c_i) = g \quad\Leftrightarrow\quad \bigsqcup_{j\geq 0} h(b^j_\omega, a^j_\omega) = g \quad \text{(for any } g \in |A|).$$

Moreover, by *(a)* and *(b)* Lemma 5.1.4(1) applies and implies, together with the above fact, the following equivalence: for each Σ-algebraic function $h : |A|^k \to |A|$,

$$\bigsqcup_{i\geq 0} h(c_i) = g \quad\Leftrightarrow\quad \bigsqcup_{i\geq 0} h(b^\omega_i, a^\omega_i) = g \quad \text{(for any } g \in |A|).$$

Now (5.7) follows immediately and this completes the proof. □

Proof of Theorem 5.2.2: For both directions we use Lemma 5.2.7. The idea of the proof relies on the observation that a strict ordered Σ-algebra is T-regular iff for all k, for all t and v as in Definition 5.2.4, it satisfies condition 1 there, for some $g \in |A|^k$. Symmetrically, a strict ordered Σ-algebra is regular iff for all k, t and v, it satisfies condition 2 in Definition 5.2.4, for some $g \in |A|^k$.

Both regular and T-regular algebras are strict and satisfy 1-equivalence and 1-definedness. Due to symmetry of k-equivalence, both implications

$$\text{T-regular} \;\Rightarrow\; \text{regular} \quad\text{and}\quad \text{T-regular} \;\Leftarrow\; \text{regular}$$

are proved similarly.

"$\Rightarrow$": Assumed A to be T-regular , we prove that each regular μ-term $t = \mu x_1.t_1$ has a value and moreover $\{(t_1)^i_{A[v]}(\perp)\}_{i\geq 0} \Rrightarrow t_{A[v]}$, by induction on $k = |t|_\mu \geq 1$. Induction step follows immediately from Lemma 5.2.7.

"$\Leftarrow$": Symmetrically, assumed A to be regular, we show (by induction on $k \geq 1$) that for each Σ-algebraic functions $f_1, \ldots, f_k : |A| \to |A|$, there exists a l.u.b. of the chain $\{\langle f_1, \ldots, f_k\rangle^i(\perp, \ldots, \perp)\}_{i\geq 0}$, which is a fixed point of $\langle f_1, \ldots, f_k\rangle$. Similarly, induction step is immediate by Lemma 5.2.7. □

Having Lemma 5.2.7, it is routine to verify that regular homomorphisms coincide with morphisms of T-regular algebras (cf. [Tiu78]).

5.3 Finitary observations

By Theorem 5.2.2 we conclude directly (in Theorem 5.3.1 below) that finite observational contexts are powerful enough for an observational equivalence in regular algebras as introduced in Section 1.3.4. More precisely, the only "infinite" regular terms we need are the least elements $\perp$ of non-observable sorts, given for instance by terms of the form $\mu x.x$.

Let OBS be a set of observable sorts, for a fixed signature Σ. Let $\mathcal{F}^{OBS}$ denote the set of all observable contexts $\gamma \in |T^\mu_\Sigma(X \cup \{z_s : s\})|_o$, $o \in OBS$, such that the only μ-subterms of γ (if any) are $\mu x.x$, for a variable x of a non-observable sort.

Our main result of this section states that indistinguishability $\sim^{OBS}_A$, induced in a regular algebra by the set of all regular observational contexts, coincides with the indistinguishability induced by contexts from $\mathcal{F}^{OBS}$; the latter is denoted by $\sim^{\mathcal{F}^{OBS}}_A$.

Theorem 5.3.1 *In a regular algebra A, $(\sim_A^{OBS}) = (\sim_A^{\mathcal{F}^{OBS}})$.*

Proof: Assume that two values $a, a' \in |A|_s$ of sort s are distinguished by some regular context γ and a valuation v, i.e., $\gamma_{A[v]}(a) \neq \gamma_{A[v]}(a')$. Without loss of generality we can assume γ to be separable.

For Σ-algebraic n-ary mappings $f_1, \ldots, f_n$, the sequence of projections on k-th coordinate: $\{\pi_k^n(\langle f_1, \ldots, f_n\rangle^i(\perp, \ldots, \perp))\}_{i \geq 0}$, for some $k \leq n$, is called *expansion* [Tiu78]. From the proof of Theorem 5.2.2 it follows that the value of each regular term is the least upper bound of some expansion. In particular, $\gamma_{A[v]}(a)$ and $\gamma_{A[v]}(a')$ are least upper bounds of expansions, say $\{a_i\}_{i\geq 0}$ and $\{a_i'\}_{i \geq 0}$, respectively. Therefore, there necessarily exists i such that $a_i \neq a_i'$. Moreover, from the proof of Theorem 5.2.2 it follows that for each element a_i (and a_i') of the expansion there exists a context u whose the only μ-subterms are of the form $\mu x.x$ (corresponding to $\perp$), satisfying $u_{A[v]}(a) = a_i$ and $u_{A[v]}(a') = a_i'$. We can easily eliminate each term $\mu x.x$ of observable sort by replacing it with a fresh variable z and extending the valuation by $z \mapsto \perp$. This way we obtain another context $\hat{u}$, being necessarily in $\mathcal{F}^{OBS}$, and a valuation $\hat{v}$, satisfying

$$\hat{u}_{A[\hat{v}]}(a) = a_i \neq a_i' = \hat{u}_{A[\hat{v}]}(a'),$$

which completes the proof. $\square$

As an important conclusion from that result, we obtain:

Corollary 5.3.2 *Ordinary context induction [Hen89] can be applied for proving behavioural properties of a regular algebra.*

From Theorem 5.3.1 we conclude moreover that observational indistinguishability in a regular algebra A is precisely the indistinguishability in $|A|_{\Sigma(\perp)_{OBS}}$, the standard $\Sigma(\perp)_{OBS}$-algebra, returned by the obvious forgetful functor $|\_|_{\Sigma(\perp)_{OBS}} : RAlg_\Sigma \to Alg_{\Sigma(\perp)_{OBS}}$; signature $\Sigma(\perp)_{OBS}$ denotes here Σ enriched by a constant symbol $\perp_s$ for each non-observable sort s. Unfortunately, from this observation we cannot conclude that observational equivalence of regular algebras is reducible to observational equivalence of standard $\Sigma(\perp)_{OBS}$-algebras. The following *does not* hold:

$$A \equiv_{OBS} B \quad \Leftrightarrow \quad |A|_{\Sigma(\perp)_{OBS}} \equiv_{OBS} |B|_{\Sigma(\perp)_{OBS}}. \tag{5.12}$$

Let us find out where the difficulties are. By Theorem 5.3.1 the forgetful functor $|\_|_{\Sigma(\perp)_{OBS}}$ commutes with observational quotient:

$$|A|_{\Sigma(\perp)_{OBS}}/_{\sim^{OBS}} \cong |A/_{\sim^{OBS}}|_{\Sigma(\perp)_{OBS}}, \tag{5.13}$$

hence implication from left to right holds in (5.12). (Note that the quotients $\_/_{\sim^{OBS}}$ on both sides of the above formula are performed in different categories.) On the other hand, from $|A|_{\Sigma(\perp)_{OBS}} \equiv_{OBS} |B|_{\Sigma(\perp)_{OBS}}$ we can only conclude that $|A/_{\sim^{OBS}}|_{\Sigma(\perp)_{OBS}} \cong |B/_{\sim^{OBS}}|_{\Sigma(\perp)_{OBS}}$, which does not guarantee $A/_{\sim^{OBS}} \cong B/_{\sim^{OBS}}$ in general.

From these considerations we conclude also that the concept of observational correspondence (cf. Section 1.3.4) cannot be spelled out by only finite contexts in the setting of regular algebras. To avoid ambiguity, in the following we will use explicit prefixes, like standard (regular) correspondence, when dealing with standard (regular) algebras.

Example 5.3.3 *Now we present a counterexample, which also demonstrates that (5.12) does not hold in general. More precisely, we define two regular algebras, A and B, such that $|A|_{\Sigma(\perp)_{OBS}}$ and $|B|_{\Sigma(\perp)_{OBS}}$ are related by a standard correspondence (hence are observationally equivalent), while A and B are not observationally equivalent themselves. These algebras are over a signature Σ containing two sorts, $\{o, n\}$, and two operations: $f : o \to o$, $g : n \to o$; the only observable sort is o.*

$$\begin{array}{ll}
A_o = \{a_0 = \perp_o, a_\infty, a'_\infty\} \cup \{a_i\}_{i=1,2,\ldots} & B_o = \{b_0 = \perp_o, b_\infty, b'_\infty\} \cup \{b_i\}_{i=1,2,\ldots} \\
A_n = \{a = \perp_n\} & B_n = \{b = \perp_n\} \\
\perp_o \le a_1 \le \ldots,\ a_\infty = \bigsqcup_{i=0,1,\ldots} a_i & \perp_o \le b_1 \le \ldots,\ b_\infty = \bigsqcup_{i=0,1,\ldots} b_i \\
a_\infty \le a'_\infty & b_\infty \le b'_\infty \\
f^A(a_i) = a_{i+1},\ f^A(a_\infty) = a_\infty,\ f^A(a'_\infty) = a'_\infty & f^B(b_i) = b_{i+1},\ f^B(b_\infty) = b_\infty,\ f^B(b'_\infty) = b'_\infty \\
g^A(a) = a_\infty & g^B(b) = b'_\infty
\end{array}$$

Both A and B are fully abstract, i.e., their indistinguishabilities are identities. This implies that they are not observationally equivalent, since they are non-isomorphic, e.g. because the equation:

$$\forall x.\ g(x) = \mu y.f(y)$$

is satisfied only in one of them. On the other hand, evidently, $|A|_{\Sigma(\perp)_{OBS}} \cong |B|_{\Sigma(\perp)_{OBS}}$; an isomorphisms is given by a standard correspondence $\sim$ which relates these two standard algebras: $\sim_o = \{(a_i, b_i)\}_{i=0,1,\ldots} \cup \{(a_\infty, b'_\infty), (a'_\infty, b_\infty)\}$, $\sim_n = \{(a, b)\}$.

In Section 5.4.1 we come back to the issue of finitary observations – it turns out that a property similar to (5.12) holds, assumed that carriers of observable sorts are *algebraic*, i.e., essentially unordered. Consequently, to show observational equivalence in such a case it is sufficient to find an appropriate standard correspondence.

The rest of this section is devoted to elaboration of an adaptation of the notion of correspondence for regular algebras.

Definition 5.3.4 *A* regular observational correspondence *(w.r.t. a set OBS of observable sorts) between two regular Σ-algebras A and B is a many-sorted relation $\sim \subseteq |A| \times |B|$ such that*

- *$\sim$ is a bijection on each observable sort,*
- *for each $n \ge 0$, a regular term $t \in T^\mu_\Sigma(\{x_1, \ldots, x_n\})$, elements $a_1, \ldots, a_n$ in $|A|$ and elements $b_1, \ldots, b_n$ in $|B|$ of appropriate sorts, whenever $a_1 \sim b_1, \ldots, a_n \sim b_n$, then*

  $$t_{A[v_a]} \sim t_{B[v_b]},$$

  where $v_a : \{x_1, \ldots, x_n\} \to |A|$ and $v_b : \{x_1, \ldots, x_n\} \to |B|$ are valuations defined by:

  $$v_a(x_1) = a_1, v_b(x_1) = b_1, \ldots, v_a(x_n) = a_n, v_b(x_n) = b_n.$$

(Note that by replacing in this definition $T^\mu_\Sigma(\{x_1, \ldots, x_n\})$ with the set of algebraic terms $T_\Sigma(\{x_1, \ldots, x_n\})$ we get a standard correspondence.)

For regular algebras, similarly as for standard and partial algebras, correspondences can be used to prove observational equivalence.

Proposition 5.3.5 *Two regular algebras are observationally equivalent iff they are related by a regular observational correspondence.*

Proof: The proof can be derived directly from the proof of more general Theorem 2.3.3 in Section 2.3. It is routine to check that a slight mismatch between regular correspondence and the free regular algebra used in the proof of Theorem 2.3.3 is inessential. □

5.4 Regular and algebraic sorts

In many practical situations we need only some sorts of a regular algebra A to have upper bounds of ω-chains $\{t^i_{A[v]}(\bot)\}_{i<\omega}$. More precisely, we would like to partition the set S of sorts into two subsets, $S = RS \cup AS$, a set RS of *regular* sorts and a set AS of non-regular ones, called *algebraic* in the sequel. The aim is to simplify the work with regular algebras and not to be bothered with considering limit values of algebraic sorts. For instance, we would not like to have an "infinite" natural number as the value of $\mu x.x + 1$.

In the sequel we will assume that subsets of regular and algebraic sorts are given in a signature: by a regular signature we will mean a triple $\langle RS, AS, F\rangle$, with disjoint sets RS and AS of regular and algebraic sorts, respectively, and a set of operation names F, as usual.

The most straightforward approach here is probably to combine in some way ordered carriers (of regular sorts) with unordered ones (of algebraic sorts). We sketch such a solution and discuss some drawbacks it creates. Then, we present a slightly different attempt to solve the problem, amounting in simulating unordered sorts within the framework of regular algebras. In the sequel it will be argued that despite some difficulties, this turns out a proper choice for dealing with non-regular sorts.

In the following sections, whenever studying observational equivalence w.r.t. a subset OBS of sorts, we will assume that $OBS \subseteq AS$, that is, only algebraic sorts can be observable. In most of examples we have $OBS = AS$, and observable sorts together with their operations are given by some basic datatypes, like integers or booleans.

Unordered carriers

A solution suggested by examples is to require algebraic sorts to be actually *unordered*. To this aim we have to modify the definition of regular algebra to deal with unordered sorts, by restricting the completeness and continuity conditions only to those iterator terms t of a regular sort $s \in RS$ which, roughly speaking, have subterms of only regular sorts on each path leading to an occurrence of the iterator variable, (intuitively speaking, iteration doesn't pass though non-regular sorts).

This idea can be easily formalized in the following way. For a given regular signature Σ, we introduce its *iterator subsignature* Σ', containing the same sorts but only those operations, whose result sort is regular. Then, we modify the completeness and continuity conditions in the definition of regular algebras by considering there only terms from $T^{\mu}_{\Sigma'}(X)$, i.e., terms over the iterator signature. As a consequence of this modification, both completeness and continuity takes place only in regular sorts, hence carrier sets of only those sorts need to be ordered. We do not even require $\bot$'s in algebraic sorts.

To illustrate these general considerations, let us have a look at a small example specification of sets of integers. We declare which sorts are intended to be regular, and assume that carriers sets of all other (i.e., algebraic) sorts are unordered in all models of the specification. Here, as well as in all example specifications in this chapter, we are actually interested only in those elements in a model, which are reachable from observable sorts (*int*, *bool* in the following specification), hence we define operations inductively and do not take care about their values on other arguments.

specification `SET1` **extends** `INT`, `BOOL` by
regular sorts
    set;

operations

$\emptyset : set;$
$insert(\_,\_) : int \times set \to set;$
$del(\_,\_) : int \times set \to set;$
$\_ \in \_ : int \times set \to bool;$

axioms

$insert(n, insert(m, s)) = insert(m, insert(n, s)),$
$insert(n, s) = s \ \Leftarrow \ n \in s,$

$n \in insert(n, s),$
$n \in insert(m, s) = n \in s \Leftarrow n \neq m,$
$n \in \emptyset = false,$

$del(n, insert(n, s)) = del(n, s),$
$del(n, s) = s \ \Leftarrow \ \neg(n \in s).$

A notational shorthand is used for terms of sort *bool*: we omit comparing with $true$, e.g. $n \in s$ stands for $n \in s = true$. Obviously, `SET1` doesn't allow for infinite sets (at least in the subalgebra generated from $\emptyset$), but it can be easily extended by adding an operation

$$inc(\_,\_) : int \times set \to set,$$

defined by

$$\begin{gathered} inc(m, \emptyset) = \emptyset, \\ inc(m, insert(n, s)) = insert(n + m, inc(m, s)), \\ inc(m, \bot_{set}) = \bot_{set}, \end{gathered}$$

which opens possibility of defining various infinite sets, like $\mu x.insert(0, inc(1, x))$ (the set of all natural numbers) or $inc(1, \mu x.insert(0, inc(2, x)))$ (the set of odd natural numbers).

The approach based on unordered carriers of algebraic sorts, although seems to be simple and safe, possesses one serious drawback. Namely, with no continuity on all sorts, the specification `SET1` doesn't define some operations unambiguously. For instance, what is the value of

$$3 \in \mu x.insert(0, inc(3, x))?$$

It is continuity which would guarantee the value $true$, since

$$3 \in insert(0, inc(3, insert(0, inc(3, \bot_{set})))) = true.$$

Similarly, $del(6, \mu x.insert(0, inc(3, x)))$ is also defined by continuity (on a regular sort). More generally, it is not sufficient to define values of *destructor* operations, like $\_ \in \_$ or $del(\_,\_)$, on sets generated by *constructor* operations, i.e., $insert(\_,\_)$ and $\emptyset$ – we must also deal with sets defined by the μ-operator.

By the way, observe that due to continuity on sort *set* we have for instance:

$$insert(3, \mu x.insert(0, inc(3, x))) = \mu x.insert(0, inc(3, x)),$$

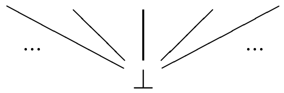
without referring to the second axiom for $insert(\_,\_)$, which could be replaced by:

$$insert(n, insert(n, s)) = insert(n, s).$$

AS-algebraicity

As argued by now, restricting definedness and continuity to regular sorts only is not a proper choice. As an alternative, we propose another solution, which simulates, roughly speaking, unordered carriers in the framework of regular algebras.

Definition 5.4.1 *We say that a regular Σ-algebra is AS-*algebraic *if it is exactly isomorphic to a regular algebra A', such that for each algebraic sort $s \in AS$, the ordering $\leq_s^{A'}$ of $|A'|_s$ is flat:*



(5.14)

(Recall that by an exact isomorphism we mean a regular isomorphism i such that its underlying function $|i|$ is an identity.) We say then that the partial order $\leq_s^A$ is essentially flat, for each algebraic sort s.

Observe that in an AS-algebraic algebra, all approximation chains of an algebraic sort collapse, in the sense that any Σ-regular mapping f induced by an iterator term is idempotent on $\perp$, that is, $f(f(\perp)) = \perp$; moreover, all those chains which are taken into account in the continuity condition collapse too.

Unless stated otherwise, in the sequel we assume that models of a specification over a regular signature $\Sigma = \langle RS, AS, F\rangle$ are all regular Σ-algebras with satisfy all axioms and which moreover are AS-algebraic.

This attempt to simulate unordered sorts in the framework of regular algebras looks quite promising, but it causes some technical problems. For example, imagine that we would like to have an operation

$$max(\_) : set \to int,$$

yielding the greatest element for bounded sets and any element for unbounded ones, i.e.,

$$\begin{array}{rcl} n \in s \ \wedge \ (\exists m.\forall n.n \in s \Rightarrow n \leq m) & \Rightarrow & n \leq max(s), \\ (\exists n.n \in s) & \Rightarrow & max(s) \in s. \end{array}$$

But now the value of $max(\mu x.insert(0, inc(1, x)))$ should be the l.u.b. of the chain of all natural numbers, by the continuity condition! Moreover, continuity condition requires a non-flat ordering of sort int, for instance:

$$0 \leq max(insert(0, inc(1, \perp_{set}))) \leq max(insert(0, inc(1, insert(0, inc(1, \perp_{set}))))) \leq 1.$$

The reason is that while specifying $max(\_)$ we do not distinguish "usual" finite sets like $insert(0, inc(1, \emptyset))$ from "partially undefined" finite sets, like $insert(0, inc(1, \perp_{set}))$. The possible way out is to redefine $max(\_)$ in such a way that its value on all partially undefined sets is $\perp_{int}$:

$$\begin{array}{rcl} n > max(s) & \Rightarrow & max(insert(n, s)) = n, \\ \neg(n > max(s)) & \Rightarrow & max(insert(n, s)) = max(s). \end{array}$$

Observe that we should define explicitly value of $max(\_)$ on $\perp_{set}$, e.g.:

$$max(\perp_{set}) = \perp_{int}. \tag{5.15}$$

Our considerations by now can be summarized as follows:

> For practical purposes, continuity condition in all sorts seems to be too restrictive. On the other hand, when one defines operations by structural induction w.r.t. a chosen subset of constructor operations, which is common in practical examples, continuity seems to be indispensable.

AS-strictness

Recall that the symbol $\perp_s$ representing the smallest element of sort s, although not present explicitly in the language, can be easily defined, for example by the term $\mu x.x$ of an appropriate sort.

As we have just observed, $\perp$'s must be treated in regular specifications to some extent as constants, and the values of operations on $\perp$'s should be explicitly given – they are not determined by continuity, as it is the case for other μ-terms. In order to not to be bothered with the $\perp$ arguments, we consider now a possibility of an implicit axiomatization of all the relevant situations as in (5.15).

First, observe that we are able to enforce all approximation chains $\{t^i_{A[v]}(\perp)\}_{i<\omega}$ to collapse (i.e., to make all upper bounds equal to $\perp$) by adding to our specifications, for each symbol $f : s_1, \ldots, s_n \to s$, and for each $i = 1, \ldots, n$, an axiom

$$\forall x_1 \in s_1, \ldots, \forall x_n \in s_n : f(x_1, \ldots, x_{i-1}, \perp_{s_i}, x_{i+1}, \ldots, x_n) = \perp_s. \tag{5.16}$$

In the case of natural numbers, equation (5.16) instantiates for instance to $\perp_{nat} + n = \perp_{nat}$. This implies $(x+1)^i_{A[v]}(\perp) = \perp_{nat}$, for $i \geq 0$, and in consequence $\mu x.x + 1 = \perp_{nat}$. An algebra satisfying all equalities of the form (5.16) can be seen as a partial algebra, with $\perp_s$ corresponding to "undefined" in sort s, since all operations are *strict*, i.e., preserve undefinedness.[1] The partial orders on all sorts are essentially flat.

Now, we are going to appropriately generalize the above idea for a subset AS of algebraic sorts. By *strictness axioms* for AS we mean the set of all formulas of the form (5.16), where either s_i or s is an algebraic sort, i.e., $\{s_i, s\} \not\subseteq RS$. In other words, we do *not* impose "strictness" only in the case when $\{s_i, s\} \subseteq RS$, i.e., both s_i and s are regular. We call a regular algebra AS-*strict* if it satisfies the strictness axioms.

In our example specification `SET1`, by strictness axioms we get "for free" many useful properties, like $insert(\perp_{int}, s) = \perp_{set}$ (inserting an "undefined" value to a set results in "undefined") or $\perp_{int} \in s = \perp_{bool}$, while still having possibility to obtain approximation chains of sort set. On the other hand, note that *not all* reasonable equalities of the form (5.16) are given automatically by strictness axioms, for example

$$del(n, \perp_{set}) = \perp_{set}.$$

When we impose strictness axioms for all sorts, all relevant chains appearing in completeness and continuity condition collapse, i.e., they are constantly $\perp$. This does not extend to the case when AS is a proper subset of sorts.

> When AS is a proper subset of sorts, the strictness axioms for AS do not even imply that partial orders on algebraic sorts are essentially flat, i.e., do not imply that all models are AS-algebraic. This is why AS-strictness alone is *not* sufficient for our purposes, and we must stick to AS-algebraic models.

The strictness axioms lead to further complications. For instance: what is the value of

$$-1 \in \mu x.insert(0, inc(1, x))? \tag{5.17}$$

We would expect $false$, but it is actually $\perp_{bool}$! Why? First, observe that AS-strictness (as well as AS-algebraicity) implies that

$$-1 \in \perp_{set} = \perp_{bool}, \quad -2 \in \perp_{set} = \perp_{bool}, \quad \ldots; \tag{5.18}$$

[1] A difference arises on morphisms: regular homomorphisms reflect definedness of operations (due to preservation of $\perp$), but not necessarily preserve definedness, as partial homomorphisms do.

otherwise, for example if we put $-1 \in \bot_{set} = false$, we would get $false \leq true$ by continuity. Second, the value $\bot_{bool}$ propagates through iterations of

$$s \mapsto insert(0, inc(1, s)),$$

since $-1 \in insert(0, inc(1, s)) = -1 \in inc(1, s) = -2 \in s$. At first sight a hopeful idea is to force $false = \bot_{bool}$. This would turn us back to two-valued logic and moreover would imply empty set $\emptyset$ to be observationally equivalent (w.r.t. observable sorts $\{int, bool\}$) to $\bot_{set}$; for instance $-1 \in \bot_{set} = false$. But on the other hand, replacing $\bot_{bool}$ with $false$ in strictness axioms would lead for instance to:

$$false \text{ or } true = false,$$

which is not acceptable, since forces a collapse $true = false$. Hence we must agree with the fact that $\mu x.insert(0, inc(1, x))$ represents, say, a partially undefined set; intuitively, one can justify the value $\bot_{bool}$ of (5.17) as follows: it is not possible to decide computationally, that is in a finite number of steps, whether -1 belongs to $\mu x.insert(0, inc(1, x))$, since it involves, roughly speaking, searching through all the elements of the infinite set.

Finally, observe that by strictness axioms it follows that $del(\bot_{int}, s) = \bot_{set}$, but on the other hand by the last axiom for $del(\_, \_)$ we have, for any s,

$$del(\bot_{int}, s) = s,$$

since $\bot_{int} \in s = \bot_{bool} \neq true$; hence we obtain $s = \bot_{set}$, surely an unexpected result! This can be easily omitted by additionally preconditioning this axiom with $n \neq \bot_{int}$, or even better by replacing the premise $n \in s \neq true$ with $n \in s = false$:

$$del(n, s) = s \ \Leftarrow \ n \in s = false;$$

it is remarkable, however, that we must deal with this kind of troubles. Moreover, we should also add an axiom $del(n, \bot_{set}) = \bot_{set}$, which is no more implied by axioms for $del(\_, \_)$. In a similar way, some other axioms must be appropriately preconditioned, for instance:

$$\begin{aligned} & insert(n, insert(m, s)) = insert(m, insert(n, s)) \Leftarrow n \neq \bot \ \wedge \ m \neq \bot, \\ & n \in insert(n, s) \Leftarrow n \neq \bot, \\ & n \in \emptyset = false \Leftarrow n \neq \bot. \end{aligned}$$

In Section 5.5 we propose a general scheme of preconditions, that exclude the $\bot$ values. From the above considerations we conclude:

> The "undefined" values $\bot$ in algebraic sorts cause technical troubles and require much care when they appear as arguments or values of operations.

In the following sections we assume that models of specifications are AS-algebraic, w.r.t. a set AS of algebraic sorts given in a signature. We assume also that models of specifications are AS-strict, i.e., that the strictness axioms are implicitly imposed. The latter is only for convenience; AS-strictness is not indispensable for our further considerations.

5.4.1 Finitary observations

When all observable sorts are algebraic, $OBS \subseteq AS$, finite observational contexts (and consequently standard correspondences) are powerful enough for observational equivalence of regular algebras, in contrast to the negative statement (5.12) in Section 5.3. The only difference is that we need here all finitary contexts, together with all bottoms $\mu x.x$, including those of observable sorts. Hence, instead of $\Sigma(\bot)_{OBS}$ we will use a slightly richer signature, $\Sigma(\bot)$, including constants $\bot_s$ of all sorts. Similarly as in Section 5.3, we use here an evident forgetful functor $|\_|_{\Sigma(\bot)} : RAlg_\Sigma \to Alg_{\Sigma(\bot)}$.

Theorem 5.4.2 *Assume that $OBS \subseteq AS$. For regular AS-algebraic Σ-algebras A and B,*

$$A \equiv_{OBS} B \quad \Leftrightarrow \quad |A|_{\Sigma(\bot)} \equiv_{OBS} |B|_{\Sigma(\bot)}.$$

Proof: Recall from Section 5.3 that the implication from left to right holds for any regular A and B (a slight modification of the forgetful functor does not influence the argument). For the opposite direction, assume that $|A|_{\Sigma(\bot)} \equiv_{OBS} |B|_{\Sigma(\bot)}$, i.e., $|A|_{\Sigma(\bot)}/_{\sim^{OBS}}$ and $|B|_{\Sigma(\bot)}/_{\sim^{OBS}}$ are related by an isomorphism of standard $\Sigma(\bot)$-algebras. Analogously as in (5.13) in Section 5.3 we conclude that

$$|A/_{\sim^{OBS}}|_{\Sigma(\bot)} \cong |B/_{\sim^{OBS}}|_{\Sigma(\bot)}.$$

Let $i : |A/_{\sim^{OBS}}|_{\Sigma(\bot)} \to |B/_{\sim^{OBS}}|_{\Sigma(\bot)}$ denote an isomorphism.

We should show that $A/_{\sim^{OBS}}$ and $B/_{\sim^{OBS}}$ are isomorphic too – to this aim it is enough to prove that i is a regular homomorphism.

Let $\mu z.t$ be a μ-term and let v be a valuation into $|A/_{\sim^{OBS}}|$. Let $a_j := t^j_{(A/\sim^{OBS})[v]}(\bot)$, $b_j := t^j_{(B/\sim^{OBS})[v;i]}(\bot)$, for $j \geq 0$, and let $a := \mu z.t_{(A/\sim^{OBS})[v]}$, $b := \mu z.t_{(B/\sim^{OBS})[v;i]}$. To show that i is a regular homomorphism, it is sufficient to show the following: for any t and v as above, whenever $i(a_j) = b_j$, for $j \geq 0$, it holds $i(a) = b$. (Then a routine inductive argument on the structure of a term proves that i is a regular homomorphism.)

Assume the contrary, i.e., $i(a) = b' \neq b$. Consider any observable $\Sigma(\bot)$-context γ and a valuation w into $|A/_{\sim^{OBS}}|$. Observe that since A and B are AS-algebraic, it follows that $A/_{\sim^{OBS}}$ and $B/_{\sim^{OBS}}$ are AS-algebraic too. Moreover $OBS \subseteq AS$, hence all relevant chains in observable sorts are collapsed in $A/_{\sim^{OBS}}$ and $B/_{\sim^{OBS}}$, i.e., are constant starting from some position.

By continuity, the value of $\gamma_{(B/\sim^{OBS})[w;i]}(b)$ in $B/_{\sim^{OBS}}$ is equal to b'' iff starting from some position, say j_0, values $\gamma_{(B/\sim^{OBS})[w;i]}(b_j)$ of γ on elements of the approximation chain $\{b_j\}$ are all equal to b'', for all $j \geq j_0$.

In a similar way the value of $\gamma_{(B/\sim^{OBS})[w;i]}(b')$ is determined by $\gamma_{(B/\sim^{OBS})[w;i]}(b_j)$. Namely, since i is a homomorphism of standard algebras and only finitary γ are considered, we have

$$\gamma_{(B/\sim^{OBS})[w;i]}(b') = i(\gamma_{(A/\sim^{OBS})[w]}(a)).$$

Moreover, by continuity in $A/_{\sim^{OBS}}$, the value of $\gamma_{(A/\sim^{OBS})[w]}(a)$ is equal to a'' iff starting from some position, values $\gamma_{(A/\sim^{OBS})[w]}(a_j)$ are all equal to a''.

From this we conclude that $\gamma_{(B/\sim^{OBS})[w;i]}(b) = \gamma_{(B/\sim^{OBS})[w;i]}(b')$, for all γ and w. This implies $b \sim^{OBS}_{B/_{\sim OBS}} b'$. But now recall that $B/_{\sim^{OBS}}$ is fully abstract, i.e., $\sim^{OBS}_{B/_{\sim OBS}}$ is identity, hence $b = b'$.

Thus we conclude that i is a regular homomorphism, which completes the proof. $\square$

We should point out where the richer signature $\Sigma(\perp)$ is actually needed in the proof. Namely, it guarantees that $\perp$'s in all observable sorts A and B are necessarily related by i. Notice that if working with $\Sigma(\perp)_{OBS}$, this would hold, by strictness axioms, only for those observable sorts, which are "reachable" by a sequence of operations in Σ from a non-observable regular sort.

As a conclusion, we obtain an effective and complete proof technique for observational equivalence of regular algebras:

Corollary 5.4.3 *If $OBS \subseteq AS$, then AS-algebraic Σ-algebras A and B are observationally equivalent if and only if they are related by a standard $\Sigma(\perp)$-correspondence.*

5.5 Behavioural implementation

A starting point now is the following specification of sets, being a slight modification of `SET1`:

specification `SET` **extends** `INT, BOOL` by
regular sorts
Set;
operations
$\emptyset : Set$;
$insert(\_,\_) : int \times Set \to Set$;
$del(\_,\_) : int \times Set \to Set$;
$\_ \in \_ : int \times Set \to bool$;
$inc(\_,\_) : int \times Set \to Set$;
axioms
$insert(n, insert(m, s)) = insert(m, insert(n, s))$,
$insert(n, insert(n, s)) = insert(n, s)$,

$n \in insert(n, s)$,
$n \in insert(m, s) = n \in s \Leftarrow n \neq m$,
$n \in \emptyset = false$,

$del(n, \emptyset) = \emptyset$,
$del(n, insert(n, s)) = del(n, s)$,
$del(n, insert(m, s)) = insert(m, del(n, s)) \ \Leftarrow\ n \neq m$,
$del(n, \perp_{Set}) = \perp_{Set}$,

$inc(m, \emptyset) = \emptyset$,
$inc(m, insert(n, s)) = insert(n + m, inc(m, s))$,
$inc(m, \perp_{Set}) = \perp_{Set}$.

According to the remarks from the previous section, some axioms should be preconditioned to exclude the $\perp$ value. For the sake of readability, we assume implicitly that all axioms are equipped with a precondition, excluding $\perp$ values of each variable of an algebraic sort. For instance, second axiom for $del(\_,\_)$ is actually:

$$del(n, insert(n, s)) = del(n, s) \Leftarrow n \neq \perp_{int}.$$

In fact, we can go one step further:

> In all specifications which follow, we impose implicit preconditions of the form $x \neq \perp$, on all variables x appearing as an argument of an operation at a position captured by strictness axioms.

In general this leads to preconditions on variables of regular sorts too.

Consider a classical implementation of sets by lists.

specification LIST **extends** INT, BOOL by
regular sorts
   $list$;
operations
   $nil : list$;
   $cons(\_,\_) : int \times list \to list$;
   $head(\_) : list \to int$;
   $tail(\_) : list \to list$;
axioms
   $head(cons(n,l)) = n$,
   $tail(cons(n,l)) = l$,
   $tail(nil) = nil$.

We fix observable sorts $OBS = \{int, bool\}$ both in SET and LIST. A constructor $\kappa_{\mathtt{LIST}}$, implementing behaviourally SET by means of LIST, transforms a model $L \in BehMod(\mathtt{LIST})$ to an algebra $S = \kappa_{\mathtt{LIST}}(L)$ over $Sig(\mathtt{SET})$ by the following rule:

$S_{Set} := L_{list}$,

$\emptyset^S := nil^L$,

$insert^S(x,l) :=$ **if** $x \in^S l$ **then** l **else** $cons^L(x,l)$,

$del^S(x, nil^L) := nil^L$,
$del^S(x, cons^L(y,l)) :=$ **if** $x = y$ **then** l **else** $cons^L(y, del^S(x,l))$,
$del^S(x, \perp_{list}) = \perp_{list}$,

$(x \in^S nil^L) := false$,
$(x \in^S cons^L(y,l)) :=$ **if** $x = y$ **then** $true$ **else** $x \in^S l$,

We use here an ad hoc, but hopefully self-explanatory notation. This is a classical example illustrating the concept of behavioural realization: sets are implemented by lists carrying different values. Operations $del(\_,\_)$ and $\_ \in \_$ are defined inductively, hence are unambiguously determined only in the subalgebra generated by nil^L and $cons^L(\_,\_)$.

Defining constructor above, we omitted operation $inc(\_,\_)$ in SET – it can be easily implemented if we extend the specification of lists with an additional operation $incr(\_,\_) : int \times list \to list$, defined by:

$incr(x, nil) := nil$,
$incr(x, cons(y,l)) := cons(x+y, incr(x,l))$,
$incr(x, \perp_{list}) := \perp_{list}$.

Note that even without operation $incr(\_,\_)$, specification LIST admits some simple infinite lists, like $\mu l.cons(0, cons(1,l))$, representing a stream $0,1,0,1,\ldots$.

We would like to stress here on some novelty which appears in the case of regular algebras, in contrast with standard or partial algebras. Namely, we are obliged to define a partial order in $\kappa_{\mathtt{LIST}}(L)$ to have it fully defined! In this case we could simply re-use the partial order from L, since the carriers are the same – but then the proof of completeness and continuity is needed for the implementations of SET operations. This seems to be quite involved, especially because a partial order in L is not determined uniquely.

In contrast to the above implementation, the partial order of the implementing model can not be re-used in general. This problems arises especially in the case when algebraic sorts implement regular ones, which seems to be common in practical examples. We will demonstrate this situation by an implementation of lists in the way this is done for example in `Pascal`, that is by pointers linking dynamically allocated memory cells. The implementing specification could be for example the following one (an operation $alloc(\_)$ with product codomain is used here, as a useful shorthand for a pair of separate operations):

specification `POINTER-LIST` **extends** `INT, BOOL` by
sorts
$memory, address;$
operations
$initmem : memory;$
$null : address;$
$alloc(\_) : memory \to memory \times address;$
$avail(\_,\_) : memory \times address \to bool;$
$\_[\_].val : memory \times address \to int;$
$\_[\_].nxt : memory \times address \to address;$
$(\_[\_].val := \_) : memory \times address \times int \to memory;$
$(\_[\_].nxt := \_) : memory \times address \times address \to memory;$
$\_[\_] \leftarrow \langle \_,\_ \rangle : memory \times address \times int \times address \to memory;$
$copy(\_,\_,\_) : memory \times address \times address \to memory;$
axioms

$$a \neq null \Rightarrow avail(initmem, a) = true,$$

$$alloc(m) = \langle m', a \rangle \Rightarrow \\ a \neq null \ \wedge \ avail(m,a) = true \ \wedge \ avail(m',a) = false \wedge \ identical(m, m', a),$$

$$a \neq null \ \wedge \ m' = (m[a].val := x) \Rightarrow \\ avail(m',a) = avail(m,a) \wedge \ m'[a].val = x \ \wedge \ m'[a].nxt = m[a].nxt \wedge \ identical(m, m', a),$$

$$a \neq null \ \wedge \ m' = (m[a].nxt := a'') \Rightarrow \\ avail(m',a) = avail(m,a) \wedge \ m'[a].nxt = x \ \wedge \ m'[a].val = m[a].val \wedge \ identical(m, m', a),$$

$$m[a] \leftarrow \langle x, a' \rangle = ((m[a].val := x)[a].nxt := a'), \\ copy(m, a, a') = ((m[a'].val := m[a].val)[a'].nxt := m[a].nxt).$$

where $identical(m, m', a)$ means that m and m' do not differ in other addresses than a:

$$identical(m, m', a) \iff (\forall a' \in address.\ a \neq a' \Rightarrow \\ avail(m,a') = avail(m',a') \wedge \ m[a'].val = m'[a'].val \ \wedge \ m[a'].nxt = m'[a'].nxt).$$

For `POINTER-LIST`, a set of observable sorts is again $OBS = \{bool, int\}$. Note that there are no regular sorts in this specification. A situation like this seems to be very common in practice, as it corresponds to a finite realization of finitely definable but infinite objects.

As an implementing constructor $\kappa_{\texttt{POINTER-LIST}}$ we choose a natural one, transforming an algebra P satisfying behaviourally `POINTER-LIST` to the $Sig(\texttt{LIST})$-algebra $L = \kappa_{\texttt{POINTER-LIST}}(P)$ as follows:

$$L_{list} := P_{memory} \times P_{address},$$

$$nil^L := \langle initmem, null \rangle,$$

$$head^L(\langle m, a \rangle) := \ \mathbf{if} \ \ a = null \vee a = \bot_{address} \ \ \mathbf{then} \ \ \bot_{int} \ \mathbf{else} \ \ m[a].val, \\ tail^L(\langle m, a \rangle) := \ \mathbf{if} \ \ a = null \vee a = \bot_{address} \ \ \mathbf{then} \ \ \langle m, a \rangle \ \mathbf{else} \ \ \langle m, m[a].nxt \rangle, \\ cons^L(x, \langle m, a \rangle) := \ \mathbf{let} \ \ \langle m', a' \rangle = alloc(m) \ \ \mathbf{in} \ \ \langle m'[a'] \leftarrow \langle x, a \rangle, a' \rangle.$$

This functor implements in a straightforward manner all finite lists, but what about infinite ones? Are they realizable in (behavioural) models of `POINTER-LIST`? The idea is to exploit cyclic lists, for example a stream $\mu x.cons(0, cons(1, x))$ would be implemented by

$$\mathbf{let} \ \ \langle m', a' \rangle = alloc(initmem), \ \ \langle m'', a'' \rangle = alloc(m') \ \ \mathbf{in} \\ \langle (m''[a'] \leftarrow \langle 1, a'' \rangle)[a''] \leftarrow \langle 0, a' \rangle, a'' \rangle. \tag{5.19}$$

We should explicitly define the realization of all lists definable by the least upper bounds. Fortunately in this case we can restrict ourselves to only terms of the form $\mu l.t$, for constructor terms $t = cons(x_1, cons(x_2, \ldots, cons(x_n, l)))$, for which a straightforward generalization of (5.19) is sufficient. However, a more flexible method is needed in general.

Roughly, our idea to define values of all μ-terms in $\kappa_{\text{POINTER-LIST}}(P)$ is as follows: first, we choose appropriate values for $\perp$'s; then, we define a semantical counterpart of μ-operator, that is for each function $f : L_{list} \to L_{list}$, a value

$$\mathsf{fix}(f) \in L_{list}.$$

This is enough to define values of all regular terms, since we can take $\mathsf{fix}(f)$ as the value of a μ-term $\mu x.t$ under a valuation v, for a function f induced by term t and this valuation. Formally, using $\mathsf{fix}(\_)$ we can define inductively value $(\mu x.t)_{A[v]}$ by:

$$(\mu x.t)_{A[v]} := \mathsf{fix}(\llcorner t_v(x) \lrcorner);$$

recall from Section 5.1 that $\llcorner t_v(x) \lrcorner = \lambda a.t_{A[v_a]}$, where v_a extends v by $v_a(x) = a$. In our example this could look as follows:

$$\perp_{list} := \langle initmem, \perp_{address} \rangle$$

$$\begin{aligned} \mathsf{fix}(f) := \ & \mathbf{let} \ \ \langle m', a' \rangle = alloc(initmem), \ \ \langle m, a \rangle = f(\langle m', a' \rangle) \ \ \mathbf{in} \\ & \mathbf{if} \ \ a \neq a' \ \ \mathbf{then} \ \ \langle copy(m, a, a'), a' \rangle \ \ \mathbf{else} \ \ \perp_{list}. \end{aligned}$$

This method of defining values of all μ-terms will be called *μ-induction* in the sequel. In general, we should define $\perp$ in each algebraic sort and a $\mathsf{fix}(\_)$ operator for each regular sort, hence μ-induction in an algebra A needs a family $\{\perp_s\}_{s \in AS}$ and a family of functions,

$$\mathsf{fix}_s : {|A|_s}^{|A|_s} \to |A|_s, \quad \text{for } s \in RS.$$

Note that for regular sorts s, we derive $\perp_s$ by $\perp_s = \mathsf{fix}_s(\mathrm{id}_{|A|_s})$. For algebraic sorts s, one can assume that $\mathsf{fix}_s(f) = f(\perp_s)$.

According to the terminology of the following section, we are able to define in this way a *pre-regular algebra*, differing from a regular algebra in at least one respect: its carrier sets are not ordered and consequently, it does not satisfy continuity; in particular, values of μ-terms given by μ-induction do not even have to be fixed points. In the following section we introduce pre-regular algebras and use them to develop a framework for behavioural implementation of regular algebras, in analogy to a constructor implementation step.

5.6 Pre-regular algebras

Let S denote the set of sorts of a fixed signature Σ. The set $|T^{\mu}_{\Sigma}(X)|$ of regular terms over X contains instances of μ-terms $\mu x.t$ different only in a variable bound by μ. Since this syntactical difference is inessential, it is reasonable to consider an abstract syntax given by the set $\mathfrak{T}^{\mu}_{\Sigma}X$ of regular terms up to α-conversion, equal to the quotient of $|T^{\mu}_{\Sigma}(X)|$ by all equalities of the form:

$$\mu x.t(x) = \mu y.t(y), \text{ for all terms } t \text{ and variables } x, y.$$

Note that $\mathfrak{T}^{\mu}_{\Sigma}(X)$ does *not* coincide with the carrier of a free regular algebra (see Section 2.1.2).

The mapping $X \mapsto \mathfrak{T}^{\mu}_{\Sigma}(X)$ can be extended to an endofunctor $\mathfrak{T}^{\mu}_{\Sigma}$ in Set^S, similarly to an endofunctor $X \mapsto |T_{\Sigma}(X)|$ yielding the set of algebraic terms; moreover, in a similar way, $\mathfrak{T}^{\mu}_{\Sigma}$ can be extended to a monad, i.e., can be equipped with a unit and a multiplication. An observation crucial for further considerations is that using μ-induction we define in fact an algebra of this monad - those algebras are called pre-regular algebras in the sequel. Morphisms of pre-regular algebras are simply morphisms of algebras of monad $\mathfrak{T}^{\mu}_{\Sigma}$; this category is denoted *Pre-RAlg*$_{\Sigma}$ in the sequel.

In a pre-regular algebra we have enough structure to define, in a canonical way, a value of a regular Σ-term. Morphisms of pre-regular algebras are precisely those functions which preserve values of all regular terms, similarly as regular homomorphisms: h is a morphism from A to B in *Pre-RAlg*$_{\Sigma}$ iff $h(t_{A[v]}) = t_{B[v;h]}$, for any valuation v in A. However, pre-regular algebras are different from regular algebras is one essential respect: μ-terms do not necessarily represent fixed points.

Each regular algebra is (via the evident forgetful functor) a pre-regular algebra; in fact, category $RAlg_{\Sigma}$ is equivalent to a full subcategory of pre-regular algebras. Moreover, those two categories are related by an adjunction:

Proposition 5.6.1 *Category of regular algebras is a reflexive subcategory of the category of pre-regular algebras.*

Proof: Observe that the notion of pre-congruence from Section 1.3 is also meaningful for pre-regular algebras. By the quotient of a pre-regular algebra A by a pre-congruence $\sqsubseteq$, denoted by $A/\sqsubseteq$, we mean the quotient by a congruence $\sqsubseteq \cap \sqsubseteq^{-1}$ induced by $\sqsubseteq$; $A/\sqsubseteq$ is defined canonically, like in the case of regular algebras. A crucial observation is that the quotient by a pre-congruence is actually a regular algebra, i.e., can be appropriately equipped with a partial order, similarly as it was done for regular algebras.

Exploiting these idea, the left adjoint to the forgetful functor is given by the quotient of a pre-regular algebra A by the smallest pre-congruence $\preccurlyeq_A$, $A \mapsto A/\preccurlyeq_A$. This mapping is functorial. To see this, notice that every morphism $f : A \to B$ of pre-regular algebras together with a pre-congruence $\sim$ on B induces a pre-congruence on A as follows:

$$\{(a, a') : f(a) \sim f(a')\}.$$

Now, since $\preccurlyeq_A$ is the smallest pre-congruence, there exists a unique morphism $A/_{\preccurlyeq_A} \to B/_{\preccurlyeq_B}$ making the following square commute (horizontal maps are quotient projections in category *Pre-RAlg*$_{\Sigma}$):

$$\begin{tikzcd} A \arrow[r, "\pi"] \arrow[d, "f"'] & A/_{\preccurlyeq_A} \arrow[d, "!"] \\ B \arrow[r, "\pi"'] & B/_{\preccurlyeq_B} \end{tikzcd}$$

Moreover, by the very definition, this induced morphism is a regular homomorphism.

Now, when B is regular, its smallest pre-congruence is identity; thus, by instantiating the diagram above we immediately get a 1-1 correspondence between $Hom_{Pre\text{-}RAlg_{\Sigma}}(A, B)$ and $Hom_{RAlg_{\Sigma}}(A/\preccurlyeq_A, B)$. This proves that the quotient functor $\_/\preccurlyeq_{\_}$ is the left adjoint to the embedding of regular algebras into *Pre-RAlg*$_{\Sigma}$; the projections π form a unit. □

Notions of subalgebra, quotient, etc. can be straightforwardly lifted to pre-regular algebras, since *Pre-RAlg*$_{\Sigma}$ together with evident forgetful functor form a concrete category (cf. Section 2.3). The relation $\models_{OBS}$ of behavioural satisfaction can be lifted to pre-regular algebras in a natural way. By $A \models_{OBS} \phi$, for a pre-regular algebra A, we mean that $A/_{\sim_A^{OBS}} \models \phi$, where $\sim_A^{OBS}$ denotes the indistinguishability induced by the set of *all* regular observable contexts (including μ-terms) in the full pre-regular subalgebra A_{OBS} of A generated by OBS. (A_{OBS} has a property that each its element is a value of some regular term with only observable variables.) Note that it is indispensable to take all contexts into account, since values of μ-terms in a pre-regular algebra can be defined arbitrarily and we can not use continuity, as in the case of regular algebras in Theorem 5.3.1, to restrict the set of relevant contexts. Moreover, considering all regular contexts guarantees that $\sim_A^{OBS}$ is a congruence on A_{OBS} (in fact, the greatest congruence which is identity on OBS).

5.6.1 Constructor implementation

In the preceding section we found pre-regular algebras useful in the behavioural implementation of regular algebras. Following the examples, we propose a methodology, as a slight adaptation of the constructor implementation of standard algebras.

Similarly as for standard algebras, given a specification SP over a regular signature $\Sigma = \langle RS, AS, F\rangle$, by $BehMod(SP)$ we mean the class of all regular AS-algebraic Σ-algebras which satisfy behaviourally SP.

Let SP, SP' be two specifications, whose models are regular algebras over signatures Σ and Σ', respectively. In order to show that SP' implements (refines) SP, we define a constructor $\kappa : BehMod(SP') \to Pre\text{-}RAlg_\Sigma$, for instance by μ-induction. This mapping induces a function $\kappa' : BehMod(SP') \to RAlg_\Sigma$ by

$$\kappa'(A) = \kappa(A)/\preccurlyeq_{\kappa(A)}.$$

Then, we say that SP' implements behaviourally SP via κ if

$$\bar{\kappa}'(BehMod(SP')) \subseteq BehMod(SP), \tag{5.20}$$

where $\bar{\kappa}'(\_)$ stands for direct image of κ'. This requires also that $\kappa'(A)$ is AS-algebraic, for all behavioural models A of SP'.

5.6.2 Behavioural consistency

Throughout this section, let OBS denote a subset of sorts of the signature Σ. As in Section 5.4.1, we assume that Σ has a distinguished subset AS of algebraic sorts and that $OBS \subseteq AS$.

Similarly as it is done in the case of standard algebras, for the correctness of the constructor implementation step one needs to show (5.20). When $SP = \langle \Sigma, Ax\rangle$ is a basic specification given by a set of axioms, (5.20) can be proved for instance by showing that $\kappa'(A)$ satisfies behaviourally axioms of SP, for any behavioural model A of SP'. However, this is difficult in practice, since the congruence in the construction of $\kappa'(A)$ is not given explicitly – the task would be simpler, if we could consider $\kappa(A)$ instead of $\kappa'(A)$. This would be the case when $\kappa(A)$ and $\kappa'(A)$ satisfy behaviourally the same formulas:

$$\{\phi : \kappa(A) \models_{OBS} \phi\} = \{\phi : \kappa'(A) \models_{OBS} \phi\}. \tag{5.21}$$

Note that $\models_{OBS}$ denotes here behavioural satisfaction in a pre-regular algebra; however, for regular $\kappa'(A)$ it agrees with the behavioural satisfaction of regular algebras. We say that implementation step is *behaviourally consistent* whenever (5.21) is satisfied.

A sufficient condition for (5.21) is:

$$\kappa(A) \equiv_{OBS} \kappa(A)/\preccurlyeq_{\kappa(A)}, \quad \text{for each } A \in BehMod(SP'), \tag{5.22}$$

where $\equiv_{OBS}$ denotes the equivalence factorized by the indistinguishabilities $\sim_A^{OBS}$. The rest of this section is devoted mainly to formulation of a condition sufficient for (5.22) in Theorem 5.6.2 below.

In the sequel let $B := \kappa(A)$ denote the pre-regular algebra yielded by κ, for a fixed $A \in BehMod(SP')$. For behavioural consistency, i.e., for

$$B \equiv_{OBS} B/\preccurlyeq_B, \tag{5.23}$$

it is sufficient to know that $\preccurlyeq_B$ induces an observational congruence on B:

$$\preccurlyeq_B \cap \preccurlyeq_B^{-1} \subseteq \sim_B^{OBS}. \tag{5.24}$$

Roughly speaking, (5.24) states that the congruence induced by $\preccurlyeq_B$ is identity on observable sorts; in other words, imposing continuity on B does not lead to identification of observable elements. From the practical perspective this is a natural condition, since it conforms neatly to the situation when observable sorts are to support some standard datatypes, which rest unaffected during a refinement step.

Indistinguishability $\sim_B^{OBS}$ is a partial congruence in general. However, for (5.24) to hold $\sim_B^{OBS}$ should be necessarily total. This is why we need in the following that B is generated by the observable sorts OBS:

$$B = B_{OBS}. \tag{5.25}$$

Let us assume then that constructor κ is defined so that $\kappa(A)$ is generated by OBS, $\kappa(A) = \kappa(A)_{OBS}$, for each A in $BehMod(SP')$. In such a case we say that κ is *OBS-generated.*

Evidently, $\sim_B^{OBS}$ is equal to $\lesssim_B^{OBS} \cap (\lesssim_B^{OBS})^{-1}$, where the *observational pre-order* $\lesssim_B^{OBS}$ is defined analogously to $\sim_B^{OBS}$: for $b, b' \in |B|_s$, $b \lesssim_B^{OBS} b'$ if and only if for all valuations $v : X \to |B|_{OBS}$ and all regular observable contexts γ,

$$\gamma_{B[v]}(b) \leq \gamma_{B[v]}(b');$$

a partial order $\leq$ denotes here the flat order in all observable sorts (with the smallest elements $\perp$'s) – this is justified by our assumption that $OBS \subseteq AS$. Having this, we conclude that for (5.24) it is sufficient to have the following:

$$\preccurlyeq_B \subseteq \lesssim_B^{OBS}. \tag{5.26}$$

Now, recalling that $\preccurlyeq_B$ is the smallest pre-congruence on B, (5.26) holds whenever

$$\lesssim_B^{OBS} \text{ is a pre-congruence.} \tag{5.27}$$

Since κ is OBS-generated, we can suitably represent each non-observable value by a term with only observable variables. This implies that operations in B are monotonic w.r.t. $\lesssim_B^{OBS}$, hence $\sim_B^{OBS}$ is preserved by operations. We need to check the other condition from the definition of pre-congruence (cf. Section 1.3.3). For any recursor $q \in |T_\Sigma^\mu(X \cup \{z : s\})|_s$ and valuation $v : X \to |B|$, let $b_i := q^i_{B[v]}(\perp)$, for $i = 0, 1, \ldots$, $b := (\mu z.q)_{B[v]}$; then we should show that

$$\begin{gathered}\text{for each term } t \in |T_\Sigma(X \cup \{y : s\})|_{s'} \text{ and valuation } u : X \to |B|,\\ \{t_{B[u_i]}\}_{i \geq 0} \text{ is a chain w.r.t. } \lesssim_B^{OBS} \text{ with the least upper bound } t_{B[u']} \ (u_i,\\ i = 0, 1, \ldots, \text{ and } u' \text{ extend } u \text{ by } u_i(y) = b_i \text{ and } u'(y) = b, \text{ respectively}).\end{gathered} \tag{5.28}$$

The assumption that κ is OBS-generated allows us to replace each occurrence of non-observable variable in t (besides y) by a term with only observable variables. Using this observation and expanding the definition of $\lesssim_B^{OBS}$, we notice that above condition is implied by the following one: for each observable context γ, valuation w in $|B|_{OBS}$ and $b, b_0, b_1, \ldots$ as above,

$$\{\gamma_{B[w]}(b_i)\}_{i \geq 0} \text{ is a chain w.r.t. } \leq \text{ with the least upper bound } \gamma_{B[w]}(b). \tag{5.29}$$

For any $i \geq 0$, b_{i+1} is obtained from b_i by application of a regular mapping $\llcorner q_v(z) \lrcorner$. Regular mappings preserve necessarily the contextual pre-order $\lesssim_B^{OBS}$, hence (5.29) can be replaced by:

- $\gamma_{B[w]}(\perp_s) \leq \gamma_{B[w]}(b_1)$,
- $\gamma_{B[w]}(b) = \bigsqcup_{i=0,1,\ldots} \gamma_{B[w]}(b_i)$ w.r.t. the partial order $\leq$.

Since $\leq$ is flat, this can be reformulated as in Theorem 5.6.2, summing up our considerations:

Theorem 5.6.2 *Assume that $OBS \subseteq AS$ and that κ is OBS-generated. A constructor implementation step determined by κ is behaviourally consistent if for each A in $BehMod(SP')$ it holds: for all contexts $\gamma \in |T_\Sigma^\mu(X \cup \{z_s : s\})|_o$ of an observable sort o and valuations w into $|B|_{OBS}$, where $B = \kappa(A)$,*

- $\gamma_{B[w]}(\perp_s) \neq \perp_o \Rightarrow \forall b \in B_s.\ \gamma_{B[w]}(b) = \gamma_{B[w]}(\perp_s)$,
- *for any recursor $q \in |T_\Sigma^\mu(X \cup \{z : s\})|_s$ and valuation $v : X \to |B|$, let $b_i := q^i_{B[v]}(\perp)$, for $i = 0, 1, \ldots$, $b := (\mu z.q)_{B[v]}$; then*

$$\begin{array}{rcl} \forall i.(\gamma_{B[w]}(b_i) \neq \perp_o & \Rightarrow & \gamma_{B[w]}(b) = \gamma_{B[w]}(b_i)), \\ (\forall i.\gamma_{B[w]}(b_i) = \perp_o) & \Rightarrow & \gamma_{B[w]}(b) = \perp_o. \end{array}$$

For instance, it can be checked that constructor $\kappa_{\text{POINTER-LIST}}$ from the previous section is behaviourally consistent.

5.6.3 Finitary contexts

Theorem 5.6.2 concerns all regular observable contexts γ. From practical perspective, especially when considering proof methods for behavioural properties, it would be useful to be able to restrict to only finitary contexts. Assume for a while that we replaced $\lesssim_B^{OBS}$ in (5.27) with pre-order induced by only finitary contexts (e.g. contexts from $\mathcal{F}^{OBS}$, as in Theorem 5.3.1 in Section 5.3). If we prove now that this pre-order is a pre-congruence, then the induced congruence, say $\sim$, necessarily coincides with $\sim_B^{OBS}$. This follows from the fact that $\sim$ is obviously coarser than $\sim_B^{OBS}$, $\sim_B^{OBS} \subseteq \sim$, but on the other hand it is necessarily identity on OBS. Since $\sim_B^{OBS}$ is the greatest congruence being identity on OBS, we get $\sim = \sim_B^{OBS}$. From this we conclude that (5.27) holds and behavioural consistency follows.

Unfortunately, our further considerations, culminating in Theorem 5.6.2, are no longer valid when only contexts γ from $\mathcal{F}^{OBS}$ are considered. Infinitary regular terms are indispensable as they can appear in contexts in (5.29), when valuation u in (5.28) maps a variable occurring in term t to value in B not representable by a finitary term with observable variables. Consequently, we must consider fairly all regular contexts in Theorem 5.6.2 – we can only eliminate those contexts, in which a context variable occurs inside a μ-terms.

5.7 Example

We conclude this chapter with another example of behavioural implementation in the setting of regular algebras, in order to illustrate the considerations from the previous sections. The following specification defines an algebra of while-programs. The `while` loop is defined by a least upper bound; other upper bounds give, roughly speaking, a language with non-parametric mutually recursive procedures.

specification `WHILE-PROG` **extends** `INT, IDENT` **by**
sorts
$\quad expr;$
regular sorts
$\quad prog, env;$
operations
$\quad \_\langle\_\rangle : env \times ident \to int;$
$\quad \_[\_/\_] : env \times int \times ident \to env;$

$\quad [\_] : int \to expr;$
$\quad [\_] : ident \to expr;$
$\quad \_+\_, \ldots : expr \times expr \to expr;$

$\quad skip : prog;$
$\quad \_ \leftarrow \_ : ident \times expr \to prog;$
$\quad \_;\_ : prog \times prog \to prog;$
$\quad if\ \_\ then\ \_\ else\ \_ : expr \times prog \times prog \to prog;$
$\quad while\ \_\ do\ \_ : expr \times prog \to prog;$

$\quad eval(\_,\_) : env \times expr \to int;$
$\quad run(\_,\_) : env \times prog \to env;$
axioms
$\quad \forall \rho \in env, n \in int, p, p_1, p_2 \in prog, e, f \in expr, x, y \in ident :$

$\quad \rho[n/x]\langle x\rangle = n,$
$\quad x \neq y \Rightarrow \rho[n/x]\langle y\rangle = \rho\langle y\rangle,$

$\quad run(\rho, skip) = \rho,$
$\quad run(\rho, x \leftarrow e) = \rho[eval(\rho, e)/x],$
$\quad run(\rho, p_1; p_2) = run(run(\rho, p_1), p_2),$
$\quad eval(\rho, e) \neq 0 \Rightarrow run(\rho, if\ e\ then\ p_1\ else\ p_2) = run(\rho, p_1),$
$\quad eval(\rho, e) = 0 \Rightarrow run(\rho, if\ e\ then\ p_1\ else\ p_2) = run(\rho, p_2),$
$\quad while\ e\ do\ p = \mu x.if\ e\ then\ p; x\ else\ skip,$

$\quad eval(\rho, [n]) = n,$
$\quad eval(\rho, [x]) = \rho\langle x\rangle,$
$\quad eval(\rho, e{+}f) = eval(\rho, e) + eval(\rho, f),$
$\quad \ldots$

Note that the sort *env* of "environments" is regular, hence we do *not* obtain the following properties:

$$run(\bot_{env}, p) = \bot_{env},$$
$$run(\rho, \bot_{prog}) = \bot_{env},$$
$$\bot_{env}[n/x] = \bot_{env}.$$

Strictness axioms state only e.g.: $\bot_{env}\langle x\rangle = \bot_{int}$. The operation $run(\_,\_)$ enables one to define the behaviour of the `while` loop by continuity:

$$run(\rho, while\ e\ do\ p) = run(\rho, \mu x.if\ e\ then\ p; x\ else\ skip).$$

For specification `WHILE-PROG` we choose the set $OBS = \{int, ident\}$ of observable sorts. This time $OBS \subsetneq AS$, i.e., not all algebraic sorts are observable.

We will implement behaviourally while-programs using the stack machine defined by the following specification. It has no regular sorts, since it is intended to describe finitely realizable computations.

specification `STACK-MACHINE` **extends** `STACK, NAT, INT` **by**
sorts
$opcode, code, memory, address$
operations
$PUSH(\_) : address \to opcode;$
$POP(\_) : address \to opcode;$
$PUSHCONST(\_) : int \to opcode;$
$GOTO(\_), IFZERO(\_) : int \to opcode;$
$GOSUB(\_) : nat \to opcode;$
$RET, STOP, HALT, NOP : opcode;$
$ADD, SUB, \ldots : opcode;$

$initmem : memory;$
$\_\langle\_\rangle : memory \times address \to int;$
$\_[\_/\_] : memory \times int \times address \to memory;$

$\varepsilon : code;$
$\{\_\} : opcode \to code;$
$\_.\_ : code \times code \to code;$
$\_\langle\langle\_\rangle\rangle : code \times nat \to opcode;$
$len(\_) : code \to nat;$

$RUN1(\_,\_,\_,\_) : stack \times memory \times nat \times opcode \to stack \times memory \times nat;$
$RUN(\_,\_,\_,\_,\_) : nat \times stack \times memory \times nat \times code \to stack \times memory \times nat;$
$stops?(\_,\_,\_,\_) : stack \times memory \times code \times nat \to bool;$
axioms
$\forall s \in stack, m \in memory, a, a' \in address,\ c, c_1, c_2, c_3 \in code, o \in opcode,\ i, j \in nat, n \in int :$

$m[n/a]\langle a\rangle = n,$
$a \neq a' \Rightarrow m[n/a]\langle a'\rangle = m\langle a'\rangle,$

$RUN1(s, m, i, STOP) = s, m, i,$
$HALT = \bot_{opcode},$
$RUN1(s, m, i, NOP) = s, m, i + 1,$
$RUN1(s, m, i, PUSH(a)) = push(s, m\langle a\rangle), m, i + 1,$
$RUN1(s, m, i, POP(a)) = pop(s), m[top(s)/a], i + 1,$
$RUN1(s, m, i, PUSHCONST(n)) = push(s, n), m, i + 1,$
$RUN1(s, m, i, ADD) = push(pop(pop(s)), top(s) + top(pop(s))), m, i + 1,$
$RUN1(s, m, i, SUB) = \ldots,$
$RUN1(s, m, i, GOTO(n)) = s, m, i + n,$
$top(s) = 0 \Rightarrow RUN1(s, m, i, IFZERO(n)) = pop(s), m, i + n,$
$top(s) \neq 0 \Rightarrow RUN1(s, m, i, IFZERO(n)) = pop(s), m, i + 1,$
$RUN1(s, m, i, GOSUB(j)) = push(s, i), m, j,$
$top(s) \neq -1 \Rightarrow RUN1(s, m, i, RET) = pop(s), m, top(s),$
$top(s) = -1 \Rightarrow RUN1(s, m, i, RET) = pop(s), m, i + 1,$

$RUN(0, s, m, i, c) = s, m, i,$
$RUN(j + 1, s, m, i, c) = \ \mathbf{let}\ \ s', m', i' = RUN1(s, m, i, c\langle\langle i\rangle\rangle)\ \ \mathbf{in}\ \ RUN(j, s', m', i', c),$
$stops?(s, m, c, j) \iff \ \mathbf{let}\ \ s', m', i' = RUN(j, s, m, 1, c)\ \ \mathbf{in}\ \ c\langle\langle i'\rangle\rangle = STOP,$

$0 \leq n < len(c_1) \Rightarrow (c_1.c_2)\langle\langle n\rangle\rangle = c_1\langle\langle n\rangle\rangle,$
$0 \leq n < len(c_2)\ \wedge\ c_2\langle\langle n\rangle\rangle = GOSUB(i) \Rightarrow (c_1.c_2)\langle\langle len(c_1) + n\rangle\rangle = GOSUB(len(c_1) + i),$
$0 \leq n < len(c_2)\ \wedge\ \neg(\exists i.\ c_2\langle\langle n\rangle\rangle = GOSUB(i)) \Rightarrow (c_1.c_2)\langle\langle len(c_1) + n\rangle\rangle = c_2\langle\langle n\rangle\rangle,$
$c\langle\langle len(c)\rangle\rangle = STOP,$
$n > len(c) \Rightarrow c\langle\langle n\rangle\rangle = HALT,$
$len(\varepsilon) = 0,$
$len(\{o\}) := 1,$
$len(c_1.c_2) := len(c_1) + len(c_2).$

We use here operations with product codomains, as a shorthand for a couple of separate operations. A triple (s, m, i) in $stack \times memory \times nat$ can be thought of as a configuration of a machine; s is a stack, m stands for operational memory and i is the program counter. According to this intuition, $RUN1(s, m, i, o)$ executes opcode o and returns a new configuration. Furthermore, $RUN(j, s, m, i, c)$ returns a configuration after executing j instructions of a program given by a machine code c defined as a sequence of opcodes; $stops?(s, m, c, j)$ informs whether c stops after j steps, when started in its first instruction.

Jumping instructions of the stack machine are addressed in two different ways: $GOTO(\_)$ and $IFZERO(\_)$ uses relative addressing, while $GOSUB(\_)$ is addressed directly. This difference must be dealt with carefully by the operation of concatenation $\_.\_$ of sequences of opcodes, namely all $GOSUB(\_)$'s in a second argument must be re-addressed. RET is complementary to $GOSUB(\_)$, it returns to an address pushed on the stack by $GOSUB(\_)$. Opcodes in a sequence c are indexed by $0, 1, \ldots, len(c) - 1$; jumping to address $len(c)$ is to denote stopping of the machine, which justifies a requirement $c\langle\langle len(c)\rangle\rangle = STOP$.

Since $HALT = \bot_{opcode}$,

$$RUN1(s, m, i, HALT) = \bot_{stack}, \bot_{memory}, \bot_{nat}$$

follows by strictness axioms. Moreover, it follows also $\{HALT\} = \bot_{code}$.

A basic idea in implementation of `WHILE-PROG` by `STACK-MACHINE` is that we should define, for all syntactical constructions of the language of while-programs, a way of compilation to the machine code. This compilation should be structural with respect to the structure of algebra of while-programs.

As observable sorts for `STACK-MACHINE` we choose only $\{int, address\}$, which are necessary to access memory cells in order to inspect program result. Note that even with sort nat not observable, it is still possible to observe a machine code c by executing its chosen i-th opcode $c\langle\langle i\rangle\rangle$, since carrier of sort nat is presumably generated from constants.

We define a constructor mapping as follows. Given a behavioural model S of `STACK-MACHINE` w.r.t. observable sorts $\{int, address\}$, a resulting Sig(`WHILE-PROG`)-algebra W is defined as follows.

$$W_{expr} = W_{prog} := S_{code}, \quad W_{env} := S_{memory}, \quad W_{ident} := S_{address}.$$

Operations in W are determined by the following rules:

$\forall m \in memory, a \in address,\ n \in int, c, c_1, c_2 \in code$

$[n] := \{PUSHCONST(n)\}$
$[a] := \{PUSH(a)\}$
$(c_1{+}c_2) := c_1.c_2.\{ADD\}$
$\ldots$

$skip := \varepsilon$
$(x \leftarrow e) := e.\{POP(x)\}$
$(c_1; c_2) := c_1.c_2$
$(if\ c\ then\ c_1\ else\ c_2) := c.\{IFZERO(len(c_1) + 2)\}.c_1.\{GOTO(len(c_2) + 1)\}.c_2$

$run(m, c) :=$
$\quad\quad$ $\mathbf{if}\ \ \exists j.stops?(empty, m, c, j)\ \mathbf{then}\ \ \ \mathbf{let}\ \ s', m', i' = RUN(j, empty, m, 1, c)\ \ \mathbf{in}\ \ m'$
$\quad\quad$ $\mathbf{else}\ \ \bot_{memory}$
$eval(m, c) :=\ \mathbf{let}\ \ s', m', i' = RUN(len(c), empty, m, 1, c)\ \ \mathbf{in}\ \ pop(s)$

Roughly, the implementation corresponds to a compilation of while-programs into the stack machine code. Value $\bot_{memory}$ signalizes that a computation of the machine does not stop. Observe, by the way, that $run(\_, \_)$ is not realizable effectively, since it would require to solve the undecidable halting problem.

We define values of all μ-terms by μ-induction, similarly as in Section 5.5.

$\perp_{ident} := \perp_{address}$
$\perp_{expr} := \{HALT\}$

$\mathsf{fix}_{prog}(f) := \{PUSHCONST(-1)\}.\mathrm{replace}(f(\{NOP\}), NOP, GOSUB(1)).\{RET\}$
$\mathsf{fix}_{env}(f) := f(\perp_{memory})$

Function replace(c, o_1, o_2) replaces each occurrence of opcode o_1 in c with opcode o_2. The definition of fix_{prog} needs some explanation. Intuitively, μ-terms of sort *prog* represent programs with non-parametrized mutually recursive procedures. According to this intuition, $\mathsf{fix}_{prog}(f)$ is roughly equal to $f(GOSUB(1))$, as $GOSUB(1)$ realizes a recursive procedure call. Moreover, $PUSHCONST(-1)$ prevents us from returning from a procedure (by RET) when it was not called. Finally, NOP was used temporarily instead of $GOSUB(1)$ to avoid the readdressing of $GOSUB(\_)$ during possible concatenation of codes by f. In particular, we get $\perp_{prog} = \{PUSHCONST(-1)\}.\{GOSUB(1)\}.\{RET\}$, a non-terminating loop.

We did not define explicitly the `while` instruction as it is definable by other operations using a fixed point. Note that if we add to our constructor the following definition:

$$while\ c\ do\ c_1 := c.\{IFZERO(len(c_1) + 2)\}.c_1.\{GOTO(-len(c) - len(c_1) - 1)\},$$

the equality defining $while\ \_\ do\ \_$ would no longer be satisfied. However, it would be still satisfied behaviourally.

The definition of fix_{env} implies in particular $\perp_{env} = \perp_{memory}$. This definition can be read out as that all the Σ-algebraic mappings induced by iterator terms are idempotent on $\perp$'s. This corresponds to an intuition that multiple updating an environment is equivalent to a single update: $\mu\rho.\rho[n/x] = \perp_{env}[n/x]$, or running a program p a number of times, starting from the $\perp$-environment, yields the same result as running p once: $\mu\rho.run(\rho, p) = run(\perp_{env}, p)$. The latter corresponds intuitively to the fact that a program p starting in $\perp_{env}$ should not refer to an uninitialized variable; if it does then $run(\perp_{env}, p)$ yields $\perp_{env}$ by strictness axioms. Note that condition $\mathsf{fix}_{env}(f) := f(\perp_{env})$ is not far from saying that the order in sort *env* is essentially flat. It forces all approximation chains of a μ-term to collapse. However, it does not say much about chains induced by continuity.

Chapter 6

Summary

The main contribution of this thesis is to put into a common framework two different concepts of behavioural equivalence: algebraic observational equivalence and open-map bisimilarity. Both these concepts are quite general and capture behavioural equivalences of numerous specific models, hence as an outcome of our work we get a clearer view of relationships between algebraic and bisimulation-like approach.

In Chapters 2 and 3 we attempted to draw direct links between the abovementioned concepts. At a somewhat informal level, results of these two chapters go in opposite directions: in Chapter 2 we capture a range of algebraic equivalences in the setting of open maps, while in Chapter 3 we give a condition for an open-map bisimilarity to be factorizable, i.e., definable by an algebraic construction of a quotient to a fully-abstract model.

In the following Chapter 4 we investigated relationships between algebraic and coalgebraic approach to behavioural equivalence. Using open-map bisimilarity of presheaves as an intermediate model, we obtained the coalgebraic representation of abstraction classes of algebraic observational equivalences.

Finally, in the last Chapter 5 our general considerations are instantiated in the framework of regular algebras. We give some explicit comparison of various versions of their definition (stating them to be equivalent) and study the notion of behavioural implementation in this context.

In the following we briefly summarize contribution of particular chapters and sketch possible continuation of this work.

The main results of Chapter 2 are characterizations of observational equivalences for a number of classes of algebras as the bisimilarities induced by subcategories of free, multifree or pre-free objects. All the characterizations are conclusions from the more general facts concerning an arbitrary adjunction or multiadjunction. Moreover, the framework of pre-adjunction was developed, and its relations to the concept of multiadjunction investigated, in order to deal with categories of models with morphisms "weakly" preserving structure of models, like category of partial algebras with weak homomorphisms.

We identified small observation subcategories, corresponding to "finitary" observations, which are sufficient to induce the observational equivalences. This can be useful in proving observational equivalence of algebras. Moreover, we observed that the algebraic bisimilarities (and more generally, those induced by (multi)free objects) are deterministic, in contrast to bisimilarities between models for concurrent systems like transition systems or event structures. Interestingly, in the case of deterministic bisimilarity, the concept of open maps specializes precisely to the notion of *etale* maps. Some further study should be devoted to analyze relationships between deterministic bisimilarity and other formulations of determinism.

The generality of our considerations allows one to apply the results in the setting of a concrete category. We discussed the possibilities to extend the results to such a setting, but further work is needed to examine applicability of our results to other frameworks, not treated here, for example to continuous algebras.

In Chapter 3 we investigate, for a given open-map bisimilarity, the issue of its factorizability by indistinguishability structures.

We analyzed conditions sufficient for the factorization of bisimilarity by partial open congruences. This general result gives an interesting conclusion that a wide class of bisimilarities (including those for transition systems, event structures and presheaves) can be equivalently defined by co-spans of open quotient projections, similarly as algebraic behavioural equivalences. We also formulated the necessary and sufficient condition for the factorization: quotients by all maximal open congruences are isomorphic.

The crucial role in our investigations is played by the domain subobjects, which are generated by "observable parts" of objects. We gave the universal construction of these subobjects, by means of a colimit of the diagram of all computation domains.

The natural possible continuation of this work would be to check whether the results can be applied to other equivalences of processes, like testing equivalence or weak bisimilarity.

In Chapter 4 we aimed at finding relationships between bisimulations defined by open maps and by coalgebra morphisms. We constructed a functorial mapping of objects of a category of models into an appropriate category of coalgebras with the property that open maps are those mapped to coalgebra morphisms. A central role in this development is played by categories of presheaves, being at an intermediate position and linking both approaches: they are uniform models to study open maps as well as they correspond to consistent coalgebras, with open maps corresponding precisely to coalgebra morphisms.

We found characterizations of presheaves corresponding to concrete models, namely many-sorted (partial) algebras. Perhaps the most important issue for further work here is to search for characterizations of subcategories of coalgebras corresponding to other specific models, along the lines of Sections 4.5, 4.5.3. A possible continuation is to verify to what extent the results can be moved to observational equivalences in other algebraic frameworks, for instance to regular algebras. An interesting question is whether there exists any general characterization of a subcategory of presheaves $\widehat{\mathbb{P}}$ meaningful as a model of computation, for a suitably chosen path category $\mathbb{P}$.

Another direction of research is motivated by the intuition about a behaviour-realization relationship, suggested in previous sections. It is desirable to closer investigate other examples of this general situation as well as to compare it to other adjunction-based approaches to a relationship between a behaviour and its implementation, for instance in [Jac96a].

In Chapter 5 we studied regular algebras and their applicability for the specification of "infinitary" datatypes. A starting point for our considerations was definition of regular algebras in [BT95] – our first result here shows that strong continuity can be equivalently replaced by weak continuity. Moreover, we showed that this definition of regular algebras is equivalent to another one, used in [Tiu78].

We proved that for observational indistinguishability in a regular algebra it is sufficient to consider only finitary contexts. This does not extend, however, to correspondences of regular algebras, which we proved by a counterexample.

As the main subject of Chapter 5 we were studying behavioural specifications in the setting of regular algebras. By imposing a "flatness" condition on non-regular sorts in models of a specification, we extended this setting to algebras with carriers of only some sorts being ordered (regular), which is required for practical examples. Moreover, we adapted the constructor behavioural implementation methodology to this framework. For an implementation step, we introduced the concepts of μ-induction and pre-regular algebra. We proposed a condition sufficient for correctness of implementation. The general considerations are illustrated by a number of examples. Surprisingly, in the case when all observable sorts are non-regular, it turns out that finitary correspondences are sufficient to prove observational equivalence of regular algebras.

Bibliography

[AO93] S. Abramsky, C.-H. Luke Ong. Full Abstraction in the Lazy Lambda Calculus. *Information and Computation* 105(2), 159-267, 1993.

[Acz88] Aczel, P. *Non-well-founded sets.* CSLI Lecture Notes 14, Stanford, 1988.

[ANR95] Adamek, J., Nelson, E., Reiterman, J. Continuous Algebras Revisited. JCSS 51(3):460-471, 1995.

[AR94] Adamek, J., Rosicky, J. Locally Presentable and Accessible Categories. London Mathematical Society Lecture Note Series, 189. Cambridge University Press, 1994.

[AGR92] Astesiano, E., Giovini, A., Reggio, G. Observational structures and their logics. *Theoretical Computer Science*, 96(1):249-283, 1992.

[Bed91] Bednarczyk, M. Hereditary history preserving bisimulations or what is the power of future perfect in programs logics. Technical report, Polish Academy of Sciences, Gdańsk, 1991.

[BB88] Benson, D.B., Ben-Schachar, O. Bisimulation of automata. *Information and Computation*, 79, 60-83, 1988.

[BB91] Bernot, G., Bidoit, M. Proving the correctness of algebraically specified software: modularity and observability issues. Technical Report LIENS-91-8, Laboratoire d'Informatique de l'Ecole Normale Superieure, 1991.

[BH95] Bidoit, M., Hennicker, R. Proving the correctness of behavioural implementations. Proc. of 4th *AMAST*'95, LNCS 936, 152-168, Springer-Verlag, 1995.

[BHW95] Bidoit, M., Hennicker, R., Wirsing, M. Behavioural and abstractor specifications. *Science of Computer Programming* 25(2-3), 1995. An earlier version: Characterising behavioural semantics and abstractor semantics. *Proc. European Symposium on Programming*, ESOP'94, Edinburgh 1994, LNCS 788, 105-119, Springer-Verlag 1994.

[BT95] Bidoit, M., Tarlecki, A. Regular algebras: a framework for observational specifications with recursive definitions. Report LIENS-95-12, Ecole Normale Superieure, 1995.

[BT96] Bidoit, M., Tarlecki A. Behavioural satisfaction and equivalence in concrete model categories. *Manuscript.* A short version appeared in *Proc. 20th Coll. on Trees in Algebra and Computing* CAAP'96, Linköping, LNCS 1059, 241-256, Springer-Verlag, 1996.

[BL70] Birkhoff, G., Lipson, J.D. Heterogeneous algebras. *J. Combinatorial Theory* 8(1970), 115-133.

[BW82] Broy, M., Wirsing, M. Partial abstract data types. *Acta Informatica* 18(1982), 47-64.

[C85] Castellani, I. Bisimulation and abstraction homomorphisms. *Proc. of CAAP'85*, LNCS 186, 223-238, Springer-Verlag, 1985.

[CW96] Cattani, G.L., Winskel, G. Presheaf models for concurrency. *Computer Science Logic* CSL'96, LNCS 1258, 58-75, Springer-Verlag, 1996. Preliminary version appeared as BRICS Report RS-96-35.

[CFW98] Cattani, G.L., Fiore, M.P., Winskel, G. A Theory of Recursive Domains with Applications to Concurrency. *Proc. 13th Annual Symposium on Logic in Computer Science* LICS'98, 214-225.

[CSW97] Cattani, G.L., Stark, I., Winskel, G. Presheaf Models for the π-Calculus. BRICS Report RS-97-34.

[CPW98] Cattani, G.L., Power, A.J., Winskel, G. A categorical axiomatics for bisimulation. *Proc. of CONCUR'98*, LNCS 1466 (1998), 581-596, Springer Verlag.

[CFW99] Cattani, G.L., Fiore, M.P., Winskel, G. Weak bisimulation and open maps. *Proc. 14th Annual Symposium on Logic in Computer Science* LICS'99, 67-76.

[CN95] Cheng, A., Nielsen, M. Open maps (at) work. Research series RS-95-23, BRICS, Department of Computer Science, University of Aarhus, 1995.

[Cir97] Cirstea, C. Coalgebra semantics for hidden algebra: Parametrised objects and inheritance. *Proceedings of 12th Workshop on Algebraic Developement Techniques*, Tarquinia, June 1997, LNCS 1376, pp. 174-189.

[CGH98] Corradini, A., Grosse-Rhode, M., Heckel, R., Structured transition systems as lax coalgebras. Proc. 1st Workshop on Coalgebraic Methods in Computer Science, Lisbon, 1998, volume 11 of Electronic Notes in Theoretical Computer Science.

[Die79] Diers, Y., Familles universelles de morphismes. *Annales de la Société Scientifique de Bruxelles*, 93, nr 3, 1979, 175-195.

[DEGL84] Drosten, K., Ehrig, H., Gogolla, M., Lipeck, U., Algebraic and operational semantics of specifications allowing exceptions and errors. *Theoretical Computer Science* 34, 289-313, 1984.

[EM85] Ehrig, H., Mahr, B. *Fundamentals of Algebraic Specification I: Equations and Initial Semantics.* Springer-Verlag 1985.

[Fio96] Fiore, M.,P. A Coinduction Principle for Recursive Data Types Based on Bisimulation. *Information and Computation*, 127, 186-198, 1996.

[GGM76] Giarratana, V., Gimona, F., Montanari, U. Observability concepts in abstract data type specification. *Proc. 5th Intl. Symp. Mathematical Foundations of Computer Science*, Gdańsk 1976, LNCS 45, Springer-Verlag 1976, 576-587.

[vGG89] van Glabeek, R.J., Goltz, U. Equivalence notions for concurrent systems and refinement of actions. *Proc. of MFCS'89*, Springer-Verlag LNCS 379, 237-248, 1989.

[GT74] Goguen, J.A., Thatcher, J.W. Initial algebra semantics. *IEEE Conf. Rec. SWAT* 15, 63-77, 1974.

[GM82] Goguen, J.A., Meseguer, J. Universal Realization, Persistent Interconnection and Implementation of Abstract Modules. *Proc. ICALP*'82, LNCS 140, 265-281, 1982.

[GTW77] Goguen, J.A., Thatcher, J.W., Wagner, E.G. Initial algebra semantics and continuous algebras. *Journal of ACM* 24, 68-95, 1977.

[GTW78] Goguen, J.A., Thatcher, J.W., Wagner, E.G. An initial algebra approach to the specification, correctness and implementation of abstract data types. *Current Trends in Programming Methodology 4, Data Structuring*, Yeh, R.T., ed., 80-149, Prentice-Hall 1978.

[Gor95] Gordon, A. A tutorial on co-induction and functional programming. *Proc. of the 1994 Glasgow Workshop on Functional Programming*, Springer Workshops in Computing, 1995, 78-95.

[Gra79] Graetzer, G. *Universal Algebra*, Springer-Verlag, 1979.

[GP83] Guessarian, I., Parisi-Presicce, F. Iterative vs. regular factor algebras. SIGACT News 15(2), 32-44, 1983.

[Hen88] Hennesy, M. *Algebraic Theory of Processes.* MIT Press series in the foundations of computing, 1988.

[Hen89] Hennicker, R. Observational implementations. *Proc. 6th Annual Symposium on Theoretical Aspects of Computer Science* STACS'89, LNCS 349, 59-71, Springer Verlag 1989.

[Hen91] Hennicker, R. Context induction: A proof principle for behavioural abstractions and algebraic implementations. *Formal Aspects of Computing*, 3(4):326-345, 1991.

[HW85] Hennicker, R., Wirsing, M. Observational specification: a Birkhoff Theorem. In Hans-Jorg Kreowski, ed., *Recent Trends in Data Type Specification*, Selected papers from 3rd *Workshop on Theory and Applications of Abstract Data Types*, Informatik Fachberichte 116, 119-135, Bremen, 1985.

[Hoa72] Hoare, C.A.R. Proofs of correctness of data representations. *Acta Informatica* 1(1972), 271-281.

[Hoa85] Hoare, C.A.R. *Communicating Sequential Processes.* Prentice-Hall, 1985.

[HN98] Hune, T., Nielsen, M. Bisimulation and Open Maps for Timed Transition Systems. *Fundamenta Informaticae*, 1998, to appear.

[Jac95] Jacobs, B. Objects and classes, coalgebraically. Technical Report CS-R9536, CWI, 1995.

[Jac96a] Jacobs, B. Automata and Behaviours in Categories of Processes. CWI Report CS-R9607, 1996.

[Jac96b] Jacobs, B. Coalgebraic specifications and models of deterministic hybrid systems. In M. Wirsing and M. Nivat, editors, *Algebraic Methods and Software Technology*, LNCS 1101, 520-535, Springer-Verlag, 1996.

[Jac97] Jacobs, B. Behaviour-refinement of coalgebraic specifications with coinductive correctness proofs. In M. Bidoit and M. Dauchet, editors, *TAPSOFT'97: Theory and Practice of Sortware Development*, LNCS 1214, 787-802, Springer-Verlag, 1997.

[JM94] Joyal, A., Moerdijk, I. A completeness theorem for open maps. *Annals of Pure and Applied Logic* 70(1994), 51-86.

[JNW93] Joyal, A., Nielsen, G. Winskel, Bisimulation and open maps. *Proc. 8th Annual Symposium on Logic in Computer Science* LICS'93, 1993, 418-427. Also appeared in: *Information and Computation*, 127 (1996), 164-185.

[Kna93] Knapik, T. Specifications algebriques observationnelles modulaires: une semantique fondee sur une relation de satisfaction observationnelle. PhD Thesis, Ecole Normal Superieur, Paris, 1993.

[Las97] Lasota, S. Open Maps as a Bridge between Algebraic Observational Equivalence and Bisimilarity. *Proceedings of 12th Workshop on Algebraic Developement Techniques*, Tarquinia, June 1997, LNCS 1376, 285-299, Springer-Verlag. The full version appeared as a technical report 97-12(249) in Institute of Informatics, Warsaw University, 1997.

[Las98a] Lasota, S. Partial-Congruence Factorization of Bisimilarity Induced by Open Maps. *Proc. 25th ICALP* (1998), Aalborg, LNCS 1443, 91-102, Springer-Verlag, 1998.

[Las98b] Lasota, S. Weak Bisimilarity and Open Maps. *Proc. 25th Seminar on Current Trends in Theory and Practice of Informatics SOFSEM* '98, LNCS 1521, 381-388, Springer-Verlag, 1998.

[Las99] Lasota, S. Coalgebra morphisms subsume open maps. *Proc. 2nd Workshop on Coalgebraic Methods in Computer Science CMCS*'99, Amsterdam 1999, volume 19 of Electronic Notes in Theoretical Computer Science, 23-38, 1999.

[Law63] Lawvere, F., W. Functorial semantics of algebraic theories. Proc. Nat. Acad. Sci. USA 52, pp. 869-873, 1963.

[Mac71] MacLane, S. *Categories for Working Mathematician*. Springer, New York, 1971.

[MM94] MacLane, S., Moerdijk, I. *Sheaves in Geometry and Logic.* Springer, New York, 1994.

[Mal96] Malcolm, G. Behavioural equivalence, bisimilarity, and minimal realisation. *Recent Trends in Data Type Specification, 11th Workshop on Specification of Abstract Data Types, WADT'95*, Oslo, 1995. M. Haveraaen, O. Owe, O.-J. Dahl, editors, Springer-Verlag LNCS 1130, 1995, 359-378.

[Mil80] Milner, R. *Calculus of Communicating Systems.* LNCS 92, Springer-Verlag, 1980.

[Mil89] Milner, R. *Communication and concurrency.* Prentice-Hall International Series in Computer Science, C. A. R. Hoare series editor, 1989.

[Mor68] Morris, J. H. Lambda-Calculus Models of Programming Languages. Ph. D. thesis, MIT, 1968.

[NW95] Nielsen, M., Winskel, G. *Models for concurrency.* Chapter 1 of *The Handbook of Logic in Computer Science*, vol. 4, Oxford University Press, eds. S. Abramsky, D. M. Gabbay and T.S. Gabbay, 1995.

[NW96] Nielsen, M., Winskel, G. Petri Nets and Bisimulation. *Theoretical Computer Science* 153, 211–244, 1996.

[NO88] Nivela, P., Orejas, F. Initial behaviour semantics for algebraic specifications. *Selected papers from the 5th Workshop on Specification of Abstract Data Types*, Gullane. LNCS 332, 184-207, Springer-Verlag 1988.

[Par81] Park, D.M.R. Concurrency and Automata on Infinite Sequences. *Proc. 5th G.I. Conference*, LNCS 104, Springer-Verlag, 1981.

[Pit93] Pitts, A., M. Relational properties of recursively defined domains. *Proc. 8th Annual Symposium on Logic in Computer Science* LICS, 86-97, 1993,

[Pit94] Pitts, A., M. A co-induction principle for recursively defined domains. *Theoretical Computer Science*, 124(2):195-219, 1994.

[Pit95] Pitts, A., M. Categorical Logic. *Handbook of Logic in Computer Science*, S. Abramsky, D. M. Gabbay, T. S. E. Mailbaum (eds), volume VI, Oxford University Press.

[Plo81] Plotkin, G.D. A structural approach to operational semantics. Technical Report DAIMI FN-19, Computer Science Department, Aarhus University, 1981.

[RT88] Rabinovich, A., Trakhtenbrot, B. Behaviour structures and nets of processes. *Fundamenta Informaticae*, 11:357-404, 1988.

[Rei81] Reichel, H. Behavioural equivalence – a unifying concept for initial and final specification methods. *Proc. 3rd Hungarian Computer Science Conference, Mathematical Models in Computer Systems*, M. Arato, L. Varga, eds., Budapest, 27-39, 1981.

[Rei85] Reichel, H. Behavioural Validity of Conditional Equations in Abstract Data Types. In *Proc. Vienna Conference on Contributions to General Algebra*, 301-324, 1985.

[Rei95] Reichel, H. An approach to object semantics based on terminal co-algebras. *Mathematical Structures in Computer Science*, 5:129-152, 1995.

[Rut96] Rutten, J.J.M.M. Universal coalgebra: a theory of systems. CWI Report CS-R9652, 1996.

[ST87] Sannella, D., Tarlecki, A. On observational equivalence and algebraic specification. *J. Computer and System Sciences* 34(1987), 150-178.

[ST88] Sannella, D., Tarlecki, A. Towards formal development of programs from algebraic specifications: implementations revisited. *Acta Informatica* 25:233-281, 1988.

[ST97] Sannella, D., Tarlecki, A. Essential Concepts of Algebraic Specification and Program Development. *Formal Aspects of Computing* 9:229-269, 1997.

[SW83] Sannella, D., Wirsing, M. A kernel language for algebraic specification and implementation. *Proc. Intl. Conf. on Foundations of Computation Theory*, Borgholm. LNCS 158, 413-427, Springer-Verlag, 1983.

[Sch87] Schoett, O. *Data abstraction and correctness of modular programming.* PhD thesis, CST-42-87, Department of Computer Science, University of Edinburgh, 1987.

[Sch90] Schoet, O. Behavioural correctness of data representation. *Science of Computer Programming*, 14:43-57, 1990.

[Tar86] Tarlecki, A. Quasi varieties in abstract algebraic institutions. *J. Computer and System Sciences* 33(1986), 333-360.

[TW86] Tarlecki, A., Wirsing, M. Continuous abstract data types. *Fundamenta Informaticae* 9(1986), 95-126.

[Tiu78] Tiuryn, J. Fixed-points and algebras with infinitely long expressions. Part I. Regular algebras. *Fundamenta Informaticae* 2(1978), 102-128.

[Tiu79] Tiuryn, J. Fixed-points and algebras with infinitely long expressions. Part II. Mu-clones of regular algebras. *Fundamenta Informaticae* 2(1979), 317-336.

[Win99] Winskel, G. Event Structures as Presheaves – Two Representation Theorems. BRICS Report RS-99-7, 1999.