Observational Structures and their Logics*

Egidio Astesiano Alessandro Giovini Gianna Reggio

Department of Mathematics
University of Genova - Italy

To the memory of David Park[1]

Abstract

A powerful paradigm is presented for defining semantics of data types which can assign sensible semantics also to data representing processes. Processes are abstractly viewed as elements of observable sort in an algebraic structure, independently of the language used for their description. In order to define process semantics depending on the observations we introduce *observational structures*, essentially first-order structures where we specify how processes are observed. Processes are observationally related by means of *experiments* considered similar depending on a *similarity law* and relations over processes are propagated to relations over elements of non-observable sort by a *propagation law*. Thus an observational equivalence is defined, as union of all observational relations, which can be seen as a very abstract generalization of bisimulation equivalences introduced by David Park.

Though being general and abstract our construction allows to extend and improve interesting classical results. For example it is shown that for finitely observable structures the observational equivalence is obtainable as a limit of a denumerable chain of iterations; our conditions, which apply to algebraic structures in general, when instantiated in the case of labelled transition systems, are more liberal than the finitely branching condition. More importantly, we show how to associate with an observational structure various modal *observational logics*, related to sets of experiment schemas, that we call *pattern sets*. The main result of the paper proves that for any family of pattern sets *representing the simulation law* the corresponding modal observational logic is a Hennessy-Milner logic: two observable objects are observationally equivalent if and only if they satisfy the same set of modal observational formulas. Indeed observational logics generalize to first-order structures various modal logics for labelled transition systems. Applications are shown to multilevel parallelism, higher-order concurrent calculi, distributed and branching bisimulation.

---

*Work partially funded by COMPASS–Esprit Basic Research Group No. 3264 and by CNR–PF Sistemi Informatici e Calcolo Parallelo.

[1]David Park, the inventor of bisimulation semantics, was to the first author not just the pioneer who friendly introduced him to semantics long time ago at the University of Warwick, but especially a scientist, whose research interests were rooted in classical european culture and in a genuine concern for social life.

The theory presented in the paper is not at all confined to give semantics of processes. Indeed it provides a general semantic paradigm for abstract data type specifications, where some data are processes. In order to support this claim, in the final section we briefly consider algebraic specifications and give small examples of specifications integrating processes, data types and functions.

Introduction

Various formalisms and languages have been proposed for describing processes, each one admitting a variety of interesting semantics, serving different purposes. The variety of formalisms and semantics has raised two fundamental issues: relating them (for example, *CCS* and Petri Nets) and investigating unifying viewpoints. A considerable amount of work has been done in the first direction, while only recently (see [AD, FM, KLP, M]) the second one has been pursued (though some early pioneering work was already in [Wi]).

An area where abstraction from the particular languages is essential is the algebraic specification of abstract data types. Indeed whenever some data are processes, in order to keep a reasonable level of abstraction, processes are to be specified just as special elements in some algebraic structure and moreover their semantics has to fit into the overall semantics of the specification. Now it is rather well-known that the classical notions of semantics for algebraic specifications turn out to be not adequate for expressing sensible semantics for processes. Thus the usual semantic paradigms for abstract data type specifications have to be extended. For example in [EPBRDG] it is shown how to build a good semantics for processes, with the use of projection spaces and initial continuous algebras; their work is much in the spirit of the process algebra approach (see [BK]) where semantic equivalences for processes are (explicitly) axiomatized. In this paper we follow an alternative way which is more similar to the approach developed by Milner (see [Mi]), on the basis of the key concept of bisimulation introduced by D.Park in [P]. Informally, we assume that the axioms of the specification, together with the usual axioms for static data types, qualify the data which are processes as dynamic entities (see the rules about transitions in *CCS*) and then from these axioms various semantics can be given depending on the observations, which however are not directly axiomatized. More specifically our construction aims at finding classes of semantics which enjoy the property of being a maximum fixpoint of a suitable transformation and hence also the powerful associated proof technique widely exploited by Milner in his fundamental work on *CCS* and *SCCS*.

Let us give an outline of the content.

Processes are abstractly viewed as observable elements of an algebraic structure, that we use for defining a semantics embodying an observational viewpoint and which is called *observational structure* (section 1.3). Essentially it consists of a first-order structure (or algebra with predicates) equipped with

- *experiments*: (possibly infinitary) first-order contexts for observable elements;

- a *similarity law* for experiments: a function which, given a (similarity) relation on the elements of the algebra, generates a similarity relation on experiments;
- a *propagation law* for relations: a function which propagates a (similarity) relation on the observable elements to a (similarity) relation on elements of the other sorts.

With each observational structure a family of *observational relations* is associated, with a maximum that we call *observational equivalence*. This equivalence, as expected, is not always a congruence; thus it is shown how to derive canonically an approximating congruence and also how to define observational equivalences which are congruences. Whenever this equivalence is a congruence we get an observational semantics by the usual quotient operation.

Our construction is a much abstract version of Park's construction of maximum bisimulation. Indeed, observational structures capture the essential ingredients for defining over algebraic structures those semantics which share with the original notion of bisimulation semantics the feature of being maximum fixpoints of suitable transformations. Hence the associated proof technique is effective: in order to show that two elements are semantically equivalent, just find an observational relation to which they belong. As a desired consequence many known bisimulation semantics for processes (presently, all known to us) are special cases of this construction. But *observational structures are not at all confined to a generalization of bisimulation semantics for processes.* Indeed because of their abstract nature and of the flexibility in the choice of the similarity laws for experiments and of the propagation laws for relations, they can be applied to give a wide range of semantics for abstract data types. It can be shown indeed that the full class of well-known semantics, like initial, final and various behavioural semantics, are special cases of this paradigm. We do not emphasize this point here, where our main purpose is to relate our approach to concurrency. Note in particular that observational structures allow to define sensible semantics for processes, whose specification includes axioms for identifying different configurations (states). This approach, fully advocated in [AR], where processes are seen as special data types, is now more and more appearing in one way or another; for example, it plays an important role, within the special setting of multiset rewriting, in the Chemical Abstract Machine technique of Berry and Boudol [BB] and is a central idea in the theory of "Rewriting as a unified model of concurrency" of Meseguer [M].

Together with introducing the new concept of observational structure and semantics and showing how it captures a wide range of semantics, this paper aims at demonstrating that the level of abstraction/generalization is the right one; in particular that it allows to state interesting fundamental results. Since here we are mainly interested in relating our work to the treatment of concurrency, we show that in our setting it is possible to extend, with improvements, two classical results about labelled transition systems. In section 1.4 it is shown that for finitely observable structures the observational equivalence is obtainable as an ω-iteration; in the particular case of labelled transition systems our conditions are more general that the classical "finitely branching" condition.

Then in section 2 we show how to associate with an observational structure various classes of *observational formulas* which play the role of Hennessy-Milner modal logics. Our contribution is not just a generalization but clarifies, we believe, some basic issues. First it is shown how every basic modality (the analogous of $< a >$ in the original logic) is associated with a set of pattern sets, i.e. schemas of experiments. Then we investigate under which conditions a family of pattern sets determines a logic characterizing the observational equivalence in the sense of a Hennessy-Milner logic: two elements are observationally equivalent iff they satisfy the same set of such formulas. The condition found is that the similarity law of experiments has to be completely determined (we say *represented*) by the given family of pattern sets. Hence, our result, that we call "generalized Hennessy-Milner theorem" is not stating that a particular logic characterizes an observational equivalence, but it is stating conditions for various "modal" logics to do so. Our conditions show that a family of pattern sets representing a similarity law does always exist, when we use the family of observational equivalence classes. However, this does not give an explicit characterization. The result points out that interesting characterizing "modal" logics are obtained in correspondence of families of pattern sets not only representing the simulation law but having a, possibly finite, explicit simple description. The strength of our result is better appreciated recalling that it may be applied to specifications with axioms about data structures and of course to higher-order concurrent calculi, since processes in our approach are just special data. Higher-order calculi are discussed while introducing the modal logic in section 2.1; then in section 2.3 modal logic characterizations of distributed and branching bisimulations as applications of the generalized Hennessy-Milner theorem are given. The examples show that our modalities are the analogous of those introduced by Hennessy-Milner for strong bisimulation and by other authors for different equivalences (e.g. in [DV]).

Throughout the paper we use various versions of the well-known *CCS* and generalizations to illustrate ideas, definitions and applications. But in section 3 we briefly show how our treatment finds its application in an algebraic specification framework, in order to integrate the specification of processes, functions and data types. This integration was the original motivation of our work (see [AR, AGR2] for the general approach).

The problem of a sensible extension to an algebraic setting of the notion of bisimulation has been first tackled in [AW], where a lattice of simulation relations is defined, whose greatest element can be seen as a possible correspondent of Park and Milner's notion of bisimulation in an algebraic framework; in [AGR1] a different concept closer to the original definition is proposed. Applications of the notion of generalized bisimulation to concurrency can be found in [AR] (where a family of parametric concurrent calculi integrating processes, functions and abstract data types is defined and its properties are studied) and in [AGR2] (where several examples of processes used as data types are given); while applications to the semantics of abstract data types can be found in [AGR1]. Our work, together with generalizing the Hennessy-Milner work (see [Mi]) to general algebraic structures, is clearly much related to the work by De Nicola and Hennessy on testing equivalences (see [DH]), and the relationship will be partly clarified in the paper. We also feel that in the framework of observational structures it is possible to formalize and deal with the hierarchies of semantics for concurrent processes presented by Abramsky in [A1]; this will be the subject of further work.

Arnold and Dicky [AD] and Ferrari and Montanari [FM] work in a partly similar direction to ours, aiming at a general framework for the semantics of concurrency (but without considering abstract data type specifications). Their approaches are however different; they define classes of models (Φ-algebras in [AD], the $\mathcal{U}CCS$ category in [FM]) and of morphisms (quasi-saturating homomorhpisms in [AD], abstraction homomorphisms in [FM], a notion introduced in [C]) and get the notion of maximum observational equivalence via terminality. A deeper analysis of the relationship between our and their work would probably be of interest. Also, it is a research topic to be examined whether with each observational structure can be associated a category such that the observational equivalence (or, the maximum congruence contained in it) can be obtained via terminality; some preliminary investigations can be found in [GR].

1 Observational Structures

The purpose of this section is to motivate the formal definitions given in section 1.3 by means of some simple examples, and also to introduce the notation.

We briefly summarize our formal framework, which is the usual one of *many-sorted total algebras with predicates*, i.e. many-sorted first-order structures. The basic definitions and results can be found in [GM]; here we repeat just the essential notions.

A *signature* Σ consists of a set of *sorts* (S), a family of *operation symbols* ($F = \{F_{\omega,s}\}_{\omega \in S^*, s \in S}$) and a family of *predicate symbols* ($P = \{P_\omega\}_{\omega \in S^*}$); moreover we denote by

- $f: s_1 \times \cdots \times s_n \to s$ the fact that $f \in F_{s_1 \ldots s_n, s}$;
- $p: s_1 \times \cdots \times s_n$ the fact that $p \in P_{s_1 \ldots s_n}$;
- $T_\Sigma(X)$ the *term algebra* on Σ and the S-sorted family of variables $X = \{X_s\}_{s \in S}$ and we write $t: s$ for $t \in (T_\Sigma(X))_s$;
- $\mathcal{FOF}_\Sigma(X)$ the set of the *first-order formulas* (with possibly infinitary conjunctions) on Σ and X; if $\phi \in \mathcal{FOF}_\Sigma(X)$, then $fv(\phi)$ is the set of the *free variables* of ϕ.

A Σ-*algebra* A is a triple $(\{A_s\}_{s \in S}, \{f^A\}_{f \in F}, \{p^A\}_{p \in P})$ s.t. for all $s \in S$ A_s is a set, for all $f: s_1 \times \cdots \times s_n \to s$ $f^A: A_{s_1} \times \cdots \times A_{s_n} \to A_s$ is a total function and for all $p: s_1 \times \cdots \times s_n$ $p^A \subseteq A_{s_1} \times \cdots \times A_{s_n}$. A is *term-generated* iff for all $s \in S$, $a \in A_s$ there exists $t \in (T_\Sigma(\emptyset))_s$ s.t. a is the interpretation of t in A. If $\phi \in \mathcal{FOF}_\Sigma(X)$ and A is a Σ-algebra, we denote as usual $A \models \phi$ the fact that ϕ holds in A.

1.1 Similarity of experiments

In this section we first rephrase the well-known (finite) CCS calculus of [Mi] (denoted here by $CCS0$) using the algebraic notations introduced above; we then recall the definition of *strong bisimulation (relation)* on $CCS0$ and introduce the idea of defining bisimulation starting from a set of *experiments* and from a notion of *similarity* between experiments. By means of some other small examples we then further motivate this point of view.

Strong Bisimulation for $CCS0$ The signature of $CCS0$ is the following, where we use the "$-$"-notation for defining mixfix operations:

sig Σ_{CCS0} =
  sorts be, act
  opns
    nil: $\rightarrow be$
    $- \cdot -: act \times be \rightarrow be$
    $- + -: be \times be \rightarrow be$
    $- \mid -: be \times be \rightarrow be$
    $\{\alpha: \rightarrow act \mid \alpha \in ACT\}$
    $\overline{-}: act \rightarrow act$
  preds
    $- \xrightarrow{-} -: be \times act \times be$

where ACT is a set of symbols for actions such that $\tau \in ACT$.

The "usual" operational model for $CCS0$ is just the term-generated algebra over the signature Σ_{CCS0} such that all and only the identifications which can be inferred from the equalities $\overline{\overline{a}} = a$ for all $a: act$, and $\overline{\tau} = \tau$ hold, and such that the interpretation of the predicate $\longrightarrow$ is the one given by means of the following inductive rules (where $a: act$ and $b, b', b'', b_1, b_1': be$).

$$\frac{}{a \cdot b \xrightarrow{a} b}$$

$$\frac{b \xrightarrow{a} b'}{b + b'' \xrightarrow{a} b'} \qquad \frac{b \xrightarrow{a} b'}{b'' + b \xrightarrow{a} b'}$$

$$\frac{b \xrightarrow{a} b'}{b|b'' \xrightarrow{a} b'|b''} \qquad \frac{b \xrightarrow{a} b'}{b''|b \xrightarrow{a} b''|b'}$$

$$\frac{b \xrightarrow{a} b_1 \quad b' \xrightarrow{\bar{a}} b_1'}{b|b' \xrightarrow{\tau} b_1|b_1'} \qquad \text{for } a \neq \tau$$

In the sequel we denote this model simply by $CCS0$. (Note that, in the algebraic terminology, $CCS0$ is the *initial model* of the algebraic specification having as signature Σ_{CCS0} and as axioms the equalities and the conditional axioms corresponding to the inductive rules given above, see the example 1 in section 3.)

It is well-known that the above model is not satisfactory as a semantic model for $CCS0$, since it distinguishes too much (for example, $b' + b''$ is different from $b'' + b'$); in this sense one is looking for better semantics for $CCS0$.

In general a semantics on an algebra A is given by means of a *congruence* on A; a congruence can be seen as an A-family satisfying additional constraints, where an A-*family* is a couple $(\{R_s\}_{s \in S}, \{R_p\}_{p \in P})$, such that for all $s \in S$, $R_s \subseteq A_s^2$ and for all $p: s_1 \times \cdots \times s_n \in P$, $R_p \subseteq A_{s_1} \times \cdots \times A_{s_n}$. Being a congruence means that the identifications in R_s and the validity in R_p are coherent between them and with the algebraic structure of A (see def. 1.1 for a complete definition). If R is a congruence on A, then A/R is the algebra, where the carriers are the quotients A_s/R_s, the operations and the predicates are defined respectively by $f^{A/R}([a_1], \ldots, [a_n]) = [f(a_1, \ldots, a_n)]$ and $([a_1], \ldots, [a_n]) \in p^{A/R}$ iff $(a_1, \ldots, a_n) \in p^A$; A/R is the model corresponding to the semantics given by R (the *semantic model*).

Hence, in this framework, a semantics for $CCS0$ is a couple (a $CCS0$-family) $R = ((R_{act}, R_{be}), R_{\longrightarrow})$, where R_{act} and R_{be} are binary relations on $CCS0_{act}$ and $CCS0_{be}$ respectively, and $R_{\longrightarrow} \subseteq \longrightarrow^{CCS0}$.

The *strong bisimulation* semantics corresponds to the idea that two $CCS0$ behaviours should be identified if and only if they behave in the same way if we can only observe the actions which label their transitions. As it is well known this semantics is given taking the quotient $CCS0/\!\sim$, where $\sim$ is the $CCS0$-family corresponding to the so-called *maximum strong bisimulation relation*.

A $CCS0$-family R is a *strong bisimulation relation* (see [P, Mi]) iff

i) $b' \; R_{be} \; b''$ implies

- for all $a: act$, $b'_1: be$, if $b' \xrightarrow{a} b'_1$ then there exists $b''_1: be$ s.t. $b'' \xrightarrow{a} b''_1$ and $b'_1 \; R_{be} \; b''_1$;
- for all $a: act$, $b''_1: be$, if $b'' \xrightarrow{a} b''_1$ then there exists $b'_1: be$ s.t. $b' \xrightarrow{a} b'_1$ and $b'_1 \; R_{be} \; b''_1$;

ii) R_{act} is the identity relation;

iii) $R_{\longrightarrow} \subseteq \longrightarrow^{CCS0}$.

The *maximum strong bisimulation* $\sim$ does exist and is the union of all the strong bisimulations.

Now let us call $x \xrightarrow{a} b$, where x is a variable, an *experiment* for $CCS0$, for every $a: act$ and every $b: be$; note that $x \xrightarrow{a} b$ is a first-order formula, since $\longrightarrow$ is a predicate symbol. Then we can rephrase the definition of strong bisimulation replacing clause i) with the following:

i) $b' \; R_{be} \; b''$ implies

- for all experiments e' if b' *passes* e', then there exists a *similar* experiment e'', such that b'' *passes* e'';
- for all experiments e'' if b'' *passes* e'', then there exists a *similar* experiment e', such that b' *passes* e'.

Clearly, if $e = x \xrightarrow{a} b$, "b' *passes* e" can be formally stated as "$e[b']$ holds in $CCS0$", where $e[b'] = e[b'/x] = b' \xrightarrow{a} b$, since $b' \xrightarrow{a} b$ is a first-order formula. In this case we define $x \xrightarrow{a} b'$ to be *similar* to all and only the experiments of the form $x \xrightarrow{a} b''$ with $b' \; R_{be} \; b''$.

Notice that the similarity relation between experiments depends on R; hence we introduce a function $\mathcal{S}$ that we call *similarity law* associating with each R a binary relation $\mathcal{S}(R)$ on experiments, which is so defined in this case: $x \xrightarrow{a'} b' \; \mathcal{S}(R) \; x \xrightarrow{a''} b''$ iff $a' = a''$ and $b' \; R_{be} \; b''$.

Weak Bisimulation If we decide that some actions, say τ actions, should not be observable, then we need a semantic equivalence which is less fine than strong bisimulation, since two behaviours whose activity differ only in the nonobservable actions performed should be made equivalent. This is achieved by defining the well-known *weak bisimulation*, which is obtained by introducing a new predicate $\Longrightarrow : be \times act \times be$ defined by the following inductive rules

$$\frac{}{b \overset{\tau}{\Longrightarrow} b} \qquad \frac{b \xrightarrow{\tau} b' \quad b' \overset{a}{\Longrightarrow} b''}{b \overset{a}{\Longrightarrow} b''} \qquad \frac{b \overset{a}{\Longrightarrow} b' \quad b' \xrightarrow{\tau} b''}{b \overset{a}{\Longrightarrow} b''},$$

and considering a different kind of experiments having form $x \overset{a}{\Longrightarrow} b$. Weak bisimulation is defined using the same definition schema of strong bisimulation by just changing the set of experiments and by using an analogous similarity law.

Divergence Sensitive Weak Bisimulation Let us extend $CCS0$ to include also some infinite behaviours (for example, either by means of a fixpoint combinator, or directly by means of operators like τ^ω defined by recursive equations, as for example $\tau^\omega = \tau \cdot \tau^\omega$). It is well-known that weak bisimulation does not distinguish properly between terminating and nonterminating behaviours (for example, τ^ω is weakly equivalent to **nil**); to get a finer semantic equivalence we introduce a new experiment, *Stop*, defined by the following infinitary first-order formula:

$$Stop = \nexists \{b_i, a_i\}_{i \in \omega}.(b_0 = x) \wedge (\bigwedge_{i \in \omega} b_i \xrightarrow{a_i} b_{i+1})$$

where the b_i's and a_i's are variables of sort be and act respectively. *Stop* succeeds on all and only the terminating behaviours. To be equivalent we require now that not only two behaviours have to exhibit the same visible actions, but they also have to agree w.r.t. termination. The definition schema of bisimulation rephrased using the concept of experiment handles already this case by taking as experiments $\{x \overset{a}{\Longrightarrow} b \mid a{:}\, act, b{:}\, be\} \cup \{Stop\}$ (and clearly *Stop* is only similar to itself), since clause i) is quantified on all experiments; the maximum bisimulation relation exists and identifies in this case all behaviours which behave in the same way w.r.t. all these experiments.

Observing Multilevel Parallelism It is useful to slightly generalize the definition schema by allowing several observed sorts. For example, suppose that we extend $CCS0$ with:

- a new sort *net* whose elements model $CCS0$ behaviours seen as nodes of a network (inductively defined as a single behaviour or a parallel composition $n_1||n_2$ of two networks) whose activities proceed in a free parallel way, except when restricted by the "/" operation;
- a new sort *lab* whose elements label the network transitions; these labels include behaviour labels and can be composed in parallel; we assume to this end a binary operation "$*$".

Clearly in this case we have two transition relations: $\longrightarrow$ on behaviours as before, and $\Longrightarrow: net \times lab \times net$ on nets defined by:

$$\frac{b \xrightarrow{a} b'}{b \xRightarrow{a} b'}$$

$$\frac{n_1 \xRightarrow{l} n_1'}{n_1 || n_2 \xRightarrow{l} n_1' || n_2} \qquad \frac{n_2 \xRightarrow{l} n_2'}{n_1 || n_2 \xRightarrow{l} n_1 || n_2'}$$

$$\frac{n_1 \xRightarrow{l_1} n_1' \quad n_2 \xRightarrow{l_2} n_2'}{n_1 || n_2 \xRightarrow{l_1 * l_2} n_1' || n_2'} \qquad \frac{n \xRightarrow{l} n'}{n/l' \xRightarrow{l} n'/l'} \quad l \neq l'.$$

Call this calculus "net-CCS". Both arrows can be used to build experiments for observing behaviours and nets, hence we have experiments of the form $x_{be} \xrightarrow{a} b$ and of the form $x_{net} \xRightarrow{l} n$; we want that the semantic identifications are made on behaviours and on nets accordingly to these experiments. It is easy to extend the definition of bisimulation by quantifying clause i) over all observed sorts. Let $O = \{be, net\}$ be the set of *observed sorts*,

$$Exp = \{x_{be} \xrightarrow{a} b,\, x_{net} \xRightarrow{l} n \mid a\colon act, b\colon be, l\colon lab, n\colon net\}$$

the set of *experiments*, and for all R let $\mathcal{S}(R)$ be the following similarity relation:

$$x_{be} \xrightarrow{a'} b' \; \mathcal{S}(R) \; x_{be} \xrightarrow{a''} b'' \quad \text{iff} \quad a' = a'' \quad \text{and} \quad b' \; R_{be} \; b'',$$

$$x_{net} \xRightarrow{l'} n' \; \mathcal{S}(R) \; x_{net} \xRightarrow{l''} n'' \quad \text{iff} \quad l' = l'' \quad \text{and} \quad n' \; R_{net} \; n''.$$

A net-CCS-family is a *multilevel bisimulation* iff

i) for all $o \in O$, $t' \; R_o \; t''$ implies for all $e' \in Exp$ with free variable of sort o

  – if $e'[t']$ holds, then there exists $e'' \in Exp$ such that $e''[t'']$ holds and $e' \; \mathcal{S}(R) \; e''$;

  – if $e'[t'']$ holds, then there exists $e'' \in Exp$ such that $e''[t']$ holds and $e'' \; \mathcal{S}(R) \; e'$;

ii) for all $s \notin O$ R_s is the identity relation;

iii) for all $p \in P$, $R_p \subseteq p^A$.

Since $\mathcal{S}$ is monotonic, then there exists the maximum multilevel bisimulation, which is also the maximum fixed point of an appropriate function.

1.2 Propagating Identities

In the examples introduced in the previous section, the semantics of the objects of the nonobserved sorts *act* and *lab* is fixed: the semantic identifications made on behaviours (and on nets) do not introduce new identifications on actions (and on labels). Clearly, this is not always the case, and we explain this point by means of an example.

CCS^+: a Higher Order CCS We extend $CCS0$ by allowing handshaking communication with exchange of behaviours (see [AR, T]); formally we add to $CCS0$ an action operation $SEND\colon be \to act$; a behaviour b can hence perform a $SEND(b')$ action, where b' is another behaviour, and the intuitive meaning is that b' is being sent as a value which can be received by some other process performing a corresponding $\overline{SEND(b')}$ action.

In this case we want that, given b' and b'', if b' is semantically equivalent to b'', then also the action $SEND(b')$ should be semantically equivalent to $SEND(b'')$, where the propagation of the semantic identifications to other sorts is represented by means of a *propagation function* $\mathcal{P}$, for all $s \in S$, $\mathcal{P}(R)_s$ is the propagation of R to the elements of sort s (we require $\mathcal{P}(R)_o = R_o$ for all $o \in O$). In this case we have that given R, if $b'\ R\ b''$, then $SEND(b')\ \mathcal{P}(R)\ SEND(b'')$, so the propagation law $\mathcal{P}$ is defined for all R as follows:

$$\begin{aligned}
\mathcal{P}(R)_{act} \quad = \quad & \{(\alpha,\alpha), (\overline{\alpha},\overline{\alpha}) \mid \alpha \in ACT\} \cup \\
& \{(SEND(b'), SEND(b'')), (\overline{SEND(b')}, \overline{SEND(b'')}) \mid b'\ R_{be}\ b''\}.
\end{aligned}$$

To complete the example, we have to define the similarity relation between experiments: it seems reasonable to consider a generic experiment $x \xrightarrow{a} b$ to be equivalent to all the experiments of the form $x \xrightarrow{a'} b'$ with $a\ \mathcal{P}(R)\ a'$ and $b\ R\ b'$. In particular if $a = SEND(b_1)$, then

$$x \xrightarrow{SEND(b_1)} b \quad \text{is similar to} \quad x \xrightarrow{SEND(b_2)} b'$$

for all $b\ R\ b'$, $b_1\ R\ b_2$. Hence the similarity law $\mathcal{S}$ can be defined in this case in terms of $\mathcal{P}$ as follows: for all R

$$x \xrightarrow{a} b\ \mathcal{S}(R)\ x \xrightarrow{a'} b'$$

for all a, a', b, b' such that $a\ \mathcal{P}(R)\ a'$, $b\ R\ b'$.

1.3 Observational Structures and their Semantics

The discussions, definitions and examples of the previous sections are collected in the notion of *observational structure* and of *(maximum) observational relation*.

In this section A denotes an algebra on a signature $\Sigma = (S, F, P)$, and $O \subseteq S$ denotes the set of the observed sorts. A semantics on A is represented by an A-*family* which is defined as follows.

Def. 1.1 *For $S' \subseteq S$, an (A,S')-*family *is an S'-indexed family $R = \{R_s\}_{s \in S'}$ s.t. for all $s \in S'$ $R_s \subseteq A_s^2$.*

*A couple $(R_S, \{R_p\}_{p \in P})$, where R_S is an (A,S)-family and $R_p \subseteq A_{s_1} \times \cdots \times A_{s_n}$ for all $p: s_1 \times \cdots \times s_n \in P$, is called A-*family.

If R is an A-family and $S' \subseteq S$, then $R|_{S'}$ is the (A,S')-family $\{R_s\}_{s \in S'}$.

A family R is reflexive *iff for all s R_s is reflexive; similarly for* symmetric, transitive *and* an equivalence.

An (A,S')-family R is a congruence *iff it is an equivalence and for all operations $f: s_1 \times \cdots \times s_n \to s \in F$ with $s_1, \ldots, s_n, s \in S'$*

$$a_i\, R_{s_i}\, a'_i \text{ for } i = 1, \ldots, n \text{ implies } f(a_1, \ldots, a_n)\, R_s\, f(a'_1, \ldots, a'_n).$$

An A-family R is a congruence *iff $R|_S$ is a congruence and for all predicates $p: s_1 \times \cdots \times s_n \in P$, $p_A \subseteq R_p$ and $a_i\, R_{s_i}\, a'_i$ for $i = 1, \ldots, n$, $(a_1, \ldots, a_n) \in R_p$ imply $(a'_1, \ldots, a'_n) \in R_p$.* □

Def. 1.2 (Experiments)
The set $\mathbf{Exp}(\Sigma, O)$ of experiments *over Σ and O is defined by*

$$\mathbf{Exp}(\Sigma, O) = \{\phi \in \mathcal{FOF}_\Sigma(X) \mid card(fv(\phi)) = 1 \wedge fv(\phi) \subseteq \bigcup_{o \in O} \{x_o\}\}.$$

If $fv(e) = \{x_o\}$ we write $e: o$. □

Given an experiment $e \in \mathbf{Exp}(\Sigma, O)$ such that $e: o$, an element $a \in A_o$ and a variable valuation v s.t. $v(x_o) = a$, we write $A \models e[a]$ to denote that e holds in A under the valuation v. Usually we do not insist in specifying the sort of an experiment whenever this is clear from the context.

Def. 1.3 (Similarity Laws)
S-law(A, O) *is the set of all monotonic functions from A-families into the set of binary relations on $\mathbf{Exp}(\Sigma, O)$ respecting the sorts of the experiments.* □

Def. 1.4 (Propagation Laws)
P-law(A, O) *is the set of all monotonic functions $\mathcal{P}$ from (A,O)-families into A-families s.t. $\mathcal{P}(R)_o = R_o$ for all $o \in O$.* □

The fact that similarity and propagation laws are monotonic is needed to prove prop. 1.8.

In section 2.3 we use the notation $\mathcal{P}_A$ for the propagation law s.t.:

- $\mathcal{P}_A(R)_s = \{(a,a) \mid a \in A_s\}$ for all $s \in S - O$;
- $\mathcal{P}_A(R)_p = p^A$ for all $p \in P$.

Def. 1.5 (Observational Structures)
An observational structure *is a 6-uple $(\Sigma, A, O, Exp, \mathcal{S}, \mathcal{P})$ where*

- *$\Sigma = (S, F, P)$ is a signature;*

- A *is a* Σ*-algebra (the structure on which we want to define a semantics);*
- $O \subseteq S$ *is a set of sorts (*observed sorts*, the sorts of the objects on which we perform some experiments);*
- $Exp \subseteq \mathbf{Exp}(\Sigma, O)$*;*
- $\mathcal{S} \in \mathbf{S\text{-}law}(A, O)$*;*
- $\mathcal{P} \in \mathbf{P\text{-}law}(A, O)$*.* $\square$

We use OS to denote a generic observational structure $(\Sigma, A, O, Exp, \mathcal{S}, \mathcal{P})$.

Def. 1.6 *An A-family R is an* observational relation *for OS (shortly, an* o-relation*) iff*

i) $\forall o \in O,\ \forall a', a'' \in A_o\ a'\ R_o\ a''$ *implies*

$*$ $\forall e' \in Exp,\ A \models e'[a']$ *implies* $\exists e'' \in Exp$ *s.t.* $e'\ \mathcal{S}(R)\ e''$ *and* $A \models e''[a'']$*;*

$**$ $\forall e'' \in Exp,\ A \models e''[a'']$ *implies* $\exists e' \in Exp$ *s.t.* $e'\ \mathcal{S}(R)\ e''$ *and* $A \models e'[a']$*;*

ii) $\forall s \in S - O,\ R_s \subseteq \mathcal{P}(R\,|_O)_s$*;*

iii) $\forall p \in P,\ R_p \subseteq \mathcal{P}(R\,|_O)_p$*.* $\square$

As for the case of strong bisimulation, for each OS there is a monotonic function $\mathcal{F}_{OS}$ on A-families, which can be used to characterize the observational relations and whose maximum fixed point (which does always exist) is the maximum observational relation.

Def. 1.7 *For all A-families R,*

$$\mathcal{F}_{OS}(R) = \mathcal{P}(\{\{(a', a'') \mid a', a'' \in A_o,\ * \text{ and } ** \text{ hold}\}\}_{o \in O}). \quad \square$$

Prop. 1.8 *The following facts hold:*

1. *an A-family R is an o-relation iff* $R \subseteq \mathcal{F}_{OS}(R)$*;*
2. $\mathcal{F}_{OS}$ *is monotonic over the complete lattice of A-families, ordered by inclusion;*
3. *the (arbitrary) union of o-relations is an o-relation;*
4. $\sim_{OS} =_{\text{def}} \bigcup\{R \mid R \subseteq \mathcal{F}_{OS}(R)\}$ *is an o-relation and* $\sim_{OS} = \text{maxfix}\,\mathcal{F}_{OS}$*.*

Proof. The proof is routine; note that the monotonicity of $\mathcal{F}_{OS}$ follows from the fact that both $\mathcal{P}$ and $\mathcal{S}$ are monotonic. $\square$

Sometimes we denote $\sim_{OS}$ simply by $\sim$ and call it *the maximum* o-relation of OS.

Notice that $a' \sim_s a''$ iff there exists an o-relation R s.t. $a'\ R_s\ a''$; moreover $(a_1, \ldots, a_n) \in \sim_p$ iff there exists an o-relation R s.t $(a_1, \ldots, a_n) \in R_p$.

In general we cannot ensure the maximum o-relation to be either reflexive, or transitive, or symmetric; to this end additional requirements on $\mathcal{P}$ and $\mathcal{S}$ can be made; below we show just an example. Note that due to this fact also preorders defined as bisimulations (e.g. applicative bisimulations for λ-calculus of [A2] and the prebisimulations for CCS of [Wa]) could be seen as o-relations of appropriate observational structures).

Prop. 1.9 *If for all A-families R we have that $\mathcal{S}(R^*) = \mathcal{S}(R)^*$ and if for all equivalences R we have that $\mathcal{P}(R)$ is an equivalence, then $\sim$ is an equivalence, where R^* is the smallest equivalence containing R.*

Proof. $\sim|_O$ is an equivalence (due to the hypothesis on $\mathcal{S}$); the hypothesis on $\mathcal{P}$ ensures then that $\sim$ itself is an equivalence. $\square$

If $\mathcal{S}$ and $\mathcal{P}$ are as in proposition 1.9, then we say that $\mathcal{S}$ *reflects equivalences* and $\mathcal{P}$ *propagates equivalences.*
Important note. We do not require that $\mathcal{S}$ reflects equivalences and $\mathcal{P}$ propagates equivalences in the definition of observational structure since these conditions are only sufficient; moreover in significant cases $\mathcal{S}$ and $\mathcal{P}$ do not satisfy these requirements and still $\sim_{OS}$ is an equivalence (even a congruence). This seems peculiar to the bisimulation-like approach, where also the observational relations are not in general equivalences, even when the maximum is so. Of course, *we are only interested in maximum o-relations which are equivalences and so from now on we assume that $\sim_{OS}$ is an equivalence and we call it observational equivalence.*

We are now going to show how we can approximate observational equivalence by an associated congruence, which is the observational equivalence of a suitable observational structure, following a classical approach often used in concurrency.

Observational models If $\sim$ is congruence, then the quotient algebra $A/\sim$ is the observational model associated with OS.

However, even when $\sim$ is an equivalence, it may be that $\sim$ is not a congruence (for example, in the case of weak bisimulation for CCS). This happens either when the observations made by the experiments are not coherent with the algebraic structure or when $\mathcal{P}$ does not generate congruences on the non-observed sorts. Sufficient conditions ensuring $\sim$ to be a congruence can be found for the case of transition systems in [GV] and for the algebraic case in [GR].

When $\sim$ is not a congruence, we can try to approximate $\sim$ by means of the greatest congruence respecting the observational requirements. This approximation is the maximum o-relation associated with an observational structure obtained by replacing each experiment $e\colon o$ of OS by the set of experiments $e[c[x_{o'}]]$ for all contexts $c[x_{o'}]\colon o$ with $o' \in O$, as it is formally shown below.

Def. 1.10 *Assume $e \in \mathbf{Exp}(\Sigma, O)$, $e\colon o$, $e' \in \mathbf{Exp}(\Sigma, O)$ is a* context filling *of e iff $e' = e[c[x_{o'}]]$ with $o' \in O$ and $c[x_{o'}]$ context of sort o with a hole of sort o'.* $\square$

Def. 1.11 *OS^{CF} (CF for context filling) denotes the observational structure $(\Sigma, A, Exp^{CF}, \mathcal{S}^{CF}, \mathcal{P})$ where:*

- *Exp^{CF} is the set of the context fillings of the experiments in Exp and*
- *$e\ \mathcal{S}^{CF}(R)\ e'$ iff there exist $\overline{e}, \overline{e'} \in Exp$ and a context $c[x_{o'}]$ s.t. $e = \overline{e}[c[x_{o'}]]$, $e' = \overline{e'}[c[x_{o'}]]$ and $\overline{e}\ \mathcal{S}(R)\ \overline{e'}$.* $\square$

In the following we say that a propagation relation $\mathcal{P}$ *propagates congruences* iff for all (A, O)-family R, if R is a congruence, then $\mathcal{P}(R)$ is a congruence.

Prop. 1.12 *Assume that A is term-generated, $\mathcal{P}$ propagates congruences and there do not exist $e, e' \in Exp$, $e \neq e'$, and two contexts $c[x_{o'}]$, $c'[x_{o'}]$ s.t. $e[c[x_{o'}]] = e'[c'[x_{o'}]]$. Then*

$$\sim_{OS^{CF}} = \bigcup \{R \mid R \text{ is an o-relation for } OS \text{ and is a congruence}\}$$

(thus $\sim_{OS^{CF}}$ is a congruence and is contained in $\sim_{OS}$).

Proof. Notice that under the proposition hypotheses we have that $e \in Exp^{CF}$ can be decomposed as $\overline{e}[c[x_{o'}]]$ in a unique way; thus by the definition of $\mathcal{S}^{CF}$,

$$(\times) \qquad \overline{e}[c[x_{o'}]]\ \mathcal{S}^{CF}(R)\ e' \quad \text{implies} \quad e' = \overline{e'}[c[x_{o'}]] \text{ with } \overline{e}\ \mathcal{S}(R)\ \overline{e'}.$$

Then for $(\times)$ and the definitions it is routine to check that if R is an o-relation for OS^{CF}, then R is an o-relation for OS) and that conversely if R is an o-relation for OS which is a congruence, then it is also an o-relation for OS^{CF}. □

Notice that prop. 1.12 offers also sufficient conditions for $\sim_{OS}$ to be a congruence: if $OS = (OS')^{CF}$ for some observational structure OS' satisfying the hypotheses of prop. 1.12, then $\sim_{OS}$ is a congruence. Moreover prop. 1.12 suggests also another way of handling the cases where observations and algebraic structure are not coherent: just by defining *observational congruences* instead of observational relations and taking the maximal one.

Classes of observational structures Observational structures can be grouped into classes having particular features. On this ground we can define and study hierarchies of observational structures on the same algebraic structure and their relationship. A detailed investigation is out of the scope of the paper. We single out two classes which will be used here.

Testing Structures Testing structures are a very simple but important class of observational structures used in section 2 to state and prove the generalized version of Hennessy-Milner theorem. They generalize the framework of testing semantics for processes introduced in [DH] and are essentially observational structures where two experiments are similar iff they are the same experiment.

Def. 1.13 *A testing structure is an observational structure $(\Sigma, A, O, Exp, \mathcal{ID}, \mathcal{P})$, where $\mathcal{ID}$ is the similarity law defined by*

$$\mathcal{ID}(R) = \{(e', e'') \mid e', e'' \in Exp \text{ logically equivalent}\},$$

for all R. □

Clearly $\mathcal{ID}$ reflects equivalences, so if $\mathcal{P}$ propagates equivalences the maximum o-relation associated with a testing structure is an equivalence; moreover if A is term-generated and Exp is closed by context filling, then the maximum o-relation is a congruence.

Initial Observational Structures These are the observational structures associated with an algebraic specification, by taking the initial model as the algebra of data and deriving canonically the propagation and the similarity law, on the bases of the associated equational deduction system (it can be shown that the construction corresponds to a free (initial) construction in the usual algebraic sense). They are those introduced in [AGR1], where also conditions for the observational equivalence to give a model are stated. Initial observational structures are formally defined in section 3, where examples are also shown.

1.4 A Hierarchy of Approximations

Recall that OS denotes a generic observational structure $(\Sigma, A, O, Exp, \mathcal{S}, \mathcal{P})$.

We build a class of A-families $(\cong_\lambda)_{\lambda \in \mathcal{O}}$ (where $\mathcal{O}$ is the class of the ordinals) approximating the maximum o-relation $\sim_{OS}$. We assume in this section that $\mathcal{S}$ reflects equivalences and $\mathcal{P}$ propagates equivalences.

Def. 1.14 *The class* $(\cong_\lambda)_{\lambda \in \mathcal{O}}$ *is defined as follows:*

- $\cong_0 = \mathcal{P}(\{A_o^2\}_{o \in O})$*;*
- $\cong_{\lambda+1} = \mathcal{F}_{OS}(\cong_\lambda)$ *(*$\mathcal{F}_{OS}$ *is given in def. 1.7);*
- *if* λ *is a limit ordinal, then* $\cong_\lambda = \bigcap_{\lambda' < \lambda} \cong_{\lambda'}$.

Finally, $\cong = \bigcap_{\lambda \in \mathcal{O}} \cong_\lambda$. □

Prop. 1.15 *For all ordinal numbers* λ*,* $\mu \in \mathcal{O}$*:*

1. $\mu < \lambda$ *implies* $\cong_\lambda \subseteq \cong_\mu$*;*
2. $\cong_\lambda$ *is an equivalence;*
3. $\sim \subseteq \cong_\lambda$*;*
4. *if* $card(\lambda) > card(A_s)$ *for all* $s \in S$*, then* $\cong_\lambda = \cong$*;*
5. $\cong = \sim$.

Proof. Standard results. □

This proposition implies that in case of carriers of denumerable cardinality the maximum fixed point of $\mathcal{F}_{OS}$ can be obtained by iterating $\mathcal{F}_{OS}$ up to the first ordinal whose cardinality is greater than that of ω. In general, as it is well-known, $\cong_\omega$ is *not* a fixed point for $\mathcal{F}_{OS}$. In particular cases however it is sufficient to stop to ω (for transition systems, this class extends the class of *finitely branching* transition systems for which $\cong_\omega = \sim$ has been proved, see [Mi]). The rest of this section is devoted to prove this claim.

The basic idea is that we can stop at $\cong_\omega$ whenever each element of observable sort passes only "few" "equivalent" experiments. In the case of finitely branching transition systems "equivalent" means $\mathcal{S}(\cong_0)$ and "few" means finite; in our general definition "equivalent" means $\mathcal{S}(\cong_k)$ for some $k \geq 0$ and "few" means finite modulo $\mathcal{S}(\cong_\omega)$.

Def. 1.16 *The observational structure OS is* finitely observable *iff for all $o \in O$, $a \in A_o$ and $e \in Exp$ there exists $k \geq 0$ s.t. the set*

$$D_{e,k}(a) = \{e' \mid e' \in Exp, e\, \mathcal{S}(\cong_k)\, e', A \models e'[a]\}$$

is finite modulo $\mathcal{S}(\cong_\omega)$. □

Theorem 1.17 *If OS is finitely observable and*

- *$\mathcal{S}(\bigcap_{i \in \mathbf{N}} R_i) = \bigcap_{i \in \mathbf{N}} \mathcal{S}(R_i)$ for all $\{R_i\}_{i \in \mathbf{N}}$, where each R_i is an A-family,*
- *$\mathcal{P}(\bigcap_{i \in \mathbf{N}} Q_i) = \bigcap_{i \in \mathbf{N}} \mathcal{P}(Q_i)$ for all $\{Q_i\}_{i \in \mathbf{N}}$, where each Q_i is an (A, O)-family,*

then $\cong_\omega = \sim_{OS}$. □

This theorem is an immediate consequence of the following lemma.

Lemma 1.18 *Under the conditions of theorem 1.17, $\cong_\omega \subseteq \sim_{OS}$.* □

Proof. It is sufficient to prove that $\cong_\omega$ is an o-relation for OS.

Assume $o \in O$, $a', a'' \in A_o$, $e' \in Exp$ s.t. $a' \cong_\omega a''$ and $A \models e'[a']$, we show that there exists $e'' \in Exp$ s.t. $e'\, \mathcal{S}(\cong_\omega)\, e''$ and $A \models e''[a'']$.

Since $\cong_\omega = \bigcap_{k \geq 0} \cong_k$, by definition of $\cong_k$ we have that for all $k \geq 0$ there exists $e''_k \in Exp$ s.t. $e' \mathcal{S}(\cong_k) e''_k$ and $A \models e''_k[a'']$.

Because OS is finitely observable, there exists $p \geq 0$ s.t. $D_{e',p}(a'')$ is finite modulo $\mathcal{S}(\cong_\omega)$.

Moreover,

$$D_{e',p}(a'') \supseteq D_{e',p+1}(a'') \supseteq \ldots$$

and $e''_k \in D_{e',k}(a'')$ for all $k \geq 0$. Thus $\{e''_k \mid k \geq 0\}$ is finite modulo $\mathcal{S}(\cong_\omega)$.

So there exists $q \geq 0$ s.t.

$$\forall h \geq 0 \quad \exists h' \geq h \quad \text{s.t.} \quad e''_{h'}\, \mathcal{S}(\cong_\omega)\, e''_q.$$

We claim that $e'\, \mathcal{S}(\cong_\omega)\, e''_q$ and since $A \models e''_q[a'']$ we have the thesis. Since $\mathcal{S}$ preserves intersections of A-families, it is sufficient to show that for all $h \geq 0$ $e' \mathcal{S}(\cong_h) e''_q$.

- $0 \leq h \leq q$; by definition $e' \mathcal{S}(\cong_q) e''_q$, which implies $e' \mathcal{S}(\cong_h) e''_q$, since because of $h \leq q$, from prop. 1.15 we have $\cong_q \subseteq \cong_h$;
- $h \geq q$; by definition of q there exists $h' \geq h$ s.t. $e''_{h'} \mathcal{S}(\cong_\omega) e''_q$ and thus $e''_{h'} \mathcal{S}(\cong_{h'}) e''_q$; moreover $e''_{h'} \mathcal{S}(\cong_{h'}) e'$ and, $\mathcal{S}(\cong_{h'})$ being an equivalence, $e' \mathcal{S}(\cong_{h'}) e''_q$; thus from prop. 1.15 we get $e' \mathcal{S}(\cong_h) e''_q$.

Conditions on nonobserved sorts are satisfied since for all $s \in S - O$ we have that

$$(\cong_\omega)_s = (\bigcap_{k \geq 0} \cong_k)_s = (\bigcap_{k \geq 0} \mathcal{P}(\cong_k|_O))_s = (\mathcal{P}(\bigcap_{k \geq 0} \cong_k|_O))_s = (\mathcal{P}(\cong_\omega|_O))_s.$$

Analogously for the conditions on the predicates. □

In the case of strong bisimulation for labelled transition systems with atomic values as labels our theorem requires that for all states s, for all experiments $x \xrightarrow{a} s_1$ there exists $k \geq 0$ s.t. the set

$$D_{x \xrightarrow{a} s_1, k}(s) = \{x \xrightarrow{a'} s' \mid x \xrightarrow{a'} s' \, \mathcal{S}(\cong_k) \, x \xrightarrow{a} s_1 \text{ and } A \models s \xrightarrow{a'} s'\}$$

is finite modulo $\mathcal{S}(\cong_\omega)$; i.e. that is finite the set of $\cong_\omega$-equivalence classes of elements of $\{s' \mid s \xrightarrow{a} s', s' \cong_{k-1} s_1\}$. While finitely branching condition requires that for all labels a the set $\{s' \mid A \models s \xrightarrow{a} s'\}$ is finite; i.e. with the above notation, that the set

$$D_{x \xrightarrow{a} s_1, 0}(s) = \{x \xrightarrow{a'} s' \mid x \xrightarrow{a'} s' \, \mathcal{S}(\cong_0) \, x \xrightarrow{a} s_1 \text{ and } A \models s \xrightarrow{a'} s'\}$$

is finite. Thus our conditions are less restrictive.

2 Observational Logic and its Properties

In this section we look for logics characterizing the observational equivalence (we will say HM logics).

We introduce the basic ideas in section 2.1 starting with the original HM logic, in order to help the reader. By looking at the first-order form of their modal formulas we illustrate how the basic modalities are determined by (sets of) schemas of experiments, that we call *patterns*; this gives us a way to define modalities in connection with pattern sets for general observational structures. This connection gives, in our opinion, a rather interesting and general characterization of modalities. It is shown as a further example how this connection works in the case of higher-order *CCS*.

In section 2.2 after giving the formal definitions we state what we call a generalized Hennessy-Milner theorem (proof in section 2.4). The central notion there is the *representability of a similarity law by a family of pattern sets*; essentially a similarity law is representable by a family of pattern sets whenever it is generated by it in a standard way. The main theorem asserts that *any family* of pattern sets representing a similarity law has an associated HM logic. Thus we have conditions for discovering different HM logics. In particular for any observational structure OS s.t. $\mathcal{S}(\sim_{OS})$ is an equivalence a trivial HM logic exists, the one associated with the family of pattern sets consisting of the $\mathcal{S}(\sim_{OS})$ equivalence classes. Clearly this logic is of no use at all. In order to have significant HM logics a suitable, as finitary as possible, family of pattern sets has to be discovered, what makes the validity of an interesting logic characterization not at all trivial.

To show the interest and the applicability of the result, having already considered in the introductory examples a higher-order calculus, we give further applications in section 2.3 to get HM logics for distributed and branching bisimulation. The examples demonstrate that our observational logic exactly extends those used by various authors for single semantic equivalences; in particular in analogy with what happens for finite CCS (see [Mi]), we do not need infinite conjunctions of observational formulas whenever there is only a finite set of experiments passed by an observable element (say a process).

2.1 From Experiments to Observational Logics

We recall the definition of the HM logic for CCS (see [HM] and also [Mi]). The set of formulas of the logic $\mathcal{HM}(CCS)$ is inductively defined as follows:

- $<a>\ \phi \in \mathcal{HM}(CCS)$ for all $a{:}\, act$, $\phi \in \mathcal{HM}(CCS)$;
- $\neg\phi \in \mathcal{HM}(CCS)$ for all $\phi \in \mathcal{HM}(CCS)$;
- $\bigwedge \Phi \in \mathcal{HM}(CCS)$ for all $\Phi \subseteq \mathcal{HM}(CCS)$;

where, if Φ is a set of formulas, $\bigwedge \Phi$ is the infinitary conjunction of all the formulas in Φ, i.e., $\bigwedge_{\phi\in\Phi} \phi$. Notice that $\bigwedge \emptyset$ corresponds to *true*. The satisfaction relation $\models \subseteq CCS_{be} \times \mathcal{HM}(CCS)$ is so defined:

- $b \models <a>\ \phi$ iff there exists b' such that $b \xrightarrow{a} b'$ and $b' \models \phi$;
- $b \models \neg\phi$ iff $b \not\models \phi$;
- $b \models \bigwedge \Phi$ iff $b \models \phi$ for all $\phi \in \Phi$.

A theorem due to Hennessy and Milner states that $b' \sim b''$ iff for all formulas $\phi \in \mathcal{HM}(CCS)$, $b' \models \phi$ iff $b'' \models \phi$ (recall that $\sim$ is the maximum strong bisimulation).

It is easy to translate all modal formulas of $\mathcal{HM}(CCS)$ into (semantically) equivalent first-order formulas; for example, the formula $<a>\ \phi$ becomes the first-order formula $\exists y.x \xrightarrow{a} y \wedge \overline{\phi}(y)$ (where $\overline{\phi}$ is the translation of ϕ). This translation enlightens the relationship between formulas and experiments: the formula $x \xrightarrow{a} y$ appearing in the translation can be thought as a *pattern* for generating experiments, since for all behaviours b we have that $x \xrightarrow{a} b$ is an experiment. For an experiment $e = x \xrightarrow{a} b$ let us denote by $e^*[x,y] = x \xrightarrow{a} y$ the corresponding pattern; the formula $<a>\ \phi$ is then equivalent to $\exists y.e^*[x,y] \wedge \overline{\phi}(y)$. Patterns arise naturally in the definition of the similarity relation between experiments for $CCS0$, where given two experiments $e_1[x]$ and $e_2[x]$ we have that

$$\begin{array}{lll} e_1[x]\ \mathcal{S}(R)\ e_2[x] & \text{iff} & e_1[x] = e_1^*[x,b_1],\ e_2[x] = e_2^*[x,b_2],\ b_1\ R\ b_2 \text{ and} \\ & & e_1^*[x,b_1] \text{ is logically equivalent to } e_2^*[x,b_2] \text{ in } CCS0. \end{array}$$

In what follows, for $\phi, \psi \in \mathcal{FOF}_\Sigma(X)$, we write $\phi = \psi$ meaning 'ϕ and ψ are logically equivalent'.

In the case of $CCS0$ the family of patterns $\{x \xrightarrow{a} y \mid a{:}\, act\}$ completely determines the value of $\mathcal{S}$ on all R; we say that $\mathcal{S}$ is *represented* by $\{x \xrightarrow{a} y \mid a{:}\, act\}$ whenever this happens. In order to fully appreciate immediately the central role of representability, it is convenient to anticipate the main result of this section: *whenever in an observational structure $\mathcal{S}$ is represented by a family of patterns $\mathcal{H}$, a Hennessy-Milner theorem holds for a generalized logic generated by $\mathcal{H}$.*

In this case, besides negation and conjunction, a family of patterns $\mathcal{H}$ introduces the set of modal combinators $< e^* > \phi_1 \cdots \phi_{n_{e^*}}$ for all $e^* \in \mathcal{H}$.

A further example may clarify this point. Consider the calculus CCS^+ (the $CCS0$ calculus where behaviours can be exchanged as values via handshaking communication, see sec. 1.2). In CCS^+ an experiment of the form $e[x] = x \xrightarrow{SEND(b)} b'$ should be seen as instantiation of the pattern $e^*[x, y_1, y_2] = x \xrightarrow{SEND(y_1)} y_2$: the introduction of the extra y_1 variable is needed if we want $\mathcal{S}$ to be represented by a family of patterns in the sense now discussed; indeed two experiments $e_1[x] = x \xrightarrow{SEND(b_1)} b'_1$ and $e_2[x] = x \xrightarrow{SEND(b_2)} b'_2$ are similar iff $e_1[x] = e^*_1[x, b_1, b'_1]$, $e_2[x] = e^*_2[x, b_2, b'_2]$, $e^*_1[x, y_1, y_2] = e^*_2[x, y_1, y_2]$ and $b_1 \; R \; b_2$, $b'_1 \; R \; b'_2$.

We extend now the logic to include experiments like these ones; starting from patterns we build modal formulas of the form

$$< e^* > \phi_1 \cdots \phi_n$$

semantically equivalent to the first-order formulas

$$\exists y_1, \ldots, y_n . e^*[x, y_1, \ldots, y_n] \wedge \phi_1(y_1) \wedge \cdots \wedge \phi_n(y_n).$$

In the case of CCS^+ the patterns for experiments are the following:

$$x \xrightarrow{a} y_1 \quad \text{for all } a \text{ s.t. } a \in ACT \text{ or } \overline{a} \in ACT,$$

$$x \xrightarrow{SEND(y_1)} y_2, \qquad x \xrightarrow{\overline{SEND(y_1)}} y_2,$$

which generate the modal formulas $< a > \phi$, $< S > \phi_1\phi_2$ and $< \bar{S} > \phi_1\phi_2$ respectively. For a similar use of modalities and similar formulas see the first logic in [DV].

The general notion which comes out from these examples is the following: for an observational structure OS, $\mathcal{S}$ is represented by a family of patterns of experiments $\mathcal{H}$ iff the following condition holds:

$$\begin{array}{lll} e_1[x] \; \mathcal{S}(R) \; e_2[x] & \text{iff} & e_1[x] = e^*_1[x, t'_1, \ldots, t'_n], \\ & & e_2[x] = e^*_2[x, t''_1, \ldots, t''_n], \\ & & e^*_1[x, y_1, \ldots, y_n] = e^*_2[x, y_1, \ldots, y_n] \text{ in } OS \text{ and} \\ & & \text{for all } i = 1, \ldots, n \text{ we have that } t'_i \; R \; t''_i. \end{array}$$

This notion of representability of $\mathcal{S}$ can be extended, in order to greatly relax the constraints in the main theorem. As it is defined above, an experiment e is similar modulo R to all and only the experiments obtained by instantiating its pattern e^* on R-equivalent observed objects. In general, it can be that $\mathcal{S}$ puts in relation experiments which do not correspond to R-equivalent instances of the same "pattern"; we can generalize the definition of representability to "families of pattern sets" as follows: if $\mathcal{H}$ is a family of pattern sets (i.e., each $H \in \mathcal{H}$ is a set of patterns) we say that $\mathcal{S}$ *is represented by* $\mathcal{H}$ essentially if $e_1[x] \; \mathcal{S}(R) \; e_2[x]$ iff for some $H \in \mathcal{H}$ there are two patterns $e^*_1[x, y_1, \ldots, y_n]$, $e^*_2[x, y_1, \ldots, y_n] \in H$, such that $e_1[x] = e_1^*[x, t'_1, \ldots, t'_n]$, $e_2[x] = e_2^*[x, t''_1, \ldots, t''_n]$ and for $i = 1, \ldots, n$ we have that $t'_i \; R \; t''_i$ (i.e., e_1 and e_2 are R-equivalent instances of two patterns belonging to the same $H \in \mathcal{H}$). While in the previous cases a representation $\mathcal{H}$ was just a family of patterns, now we replace each pattern e^* with a set of patterns H such that two patterns in H are similar whenever instantiated on R-equivalent objects. The simpler cases correspond thus to the case where each H is a singleton.

In the following section we develop the technical details of this idea. Here we point out that the effect on formulas of using a family of pattern sets (instead of a family of patterns) is the replacement of formulas of the form, where e^* is a pattern,

$$< e^* > \phi_1 \cdots \phi_n$$

semantically equivalent to

$$\exists y_1, \ldots, y_n . e^*[x, y_1, \ldots, y_n] \wedge \bigwedge_{i=1}^{n} \phi_i[y_i],$$

with formulas of the form, where H is a set of patterns,

$$< H > \phi_1 \cdots \phi_n$$

semantically equivalent to

$$\exists y_1, \ldots, y_n . \bigvee_{e^* \in H} e^*[x, y_1, \ldots, y_n] \wedge \bigwedge_{i=1}^{n} \phi_i[y_i].$$

The disjunction $\bigvee_{e^* \in H}$ models the fact that since all R-equivalent instances of the patterns in H are similar, we allow an instance of any of them to succeed.

The use of modalities corresponds to the semantics given in the cases when processes are modelled by labelled transition systems; whenever the data are labelled transition systems there is a corresponding Kripke structure (see [S] for a general discussion). But our formalism permits to extend the approach to generic specifications of data types and also to treat higher-order cases, without any ad hoc construction.

2.2 A Generalized Hennessy-Milner Theorem

Def. 2.1 *A* family of pattern sets *is a family $\mathcal{H}$ such that $H \in \mathcal{H}$ implies that $H \subseteq \mathcal{FOF}_\Sigma(X_O)$ with X_O variables of observed sorts, for all $\phi \in H$, $fv(\phi) \subseteq \{x, y_1, \ldots, y_n\}$ and $\phi[x, a_1, \ldots, a_n] \in Exp$ for all $a_1, \ldots, a_n \in A$ of appropriate sort. If $x\colon o$, and $x_i\colon o_i$ then we say that H has* type *$o_1 \times \cdots \times o_n \to o$.* □

Def. 2.2 *For any family of pattern sets $\mathcal{H}$ we define inductively the family $\mathcal{H}^* = \{\mathcal{H}^*_o\}_{o \in O}$ of* observational formulas w.r.t. *$\mathcal{H}$ as follows:*

- $< H > \phi_1 \cdots \phi_n \in \mathcal{H}^*_o$
  for all $H \in \mathcal{H}$ with type $o_1 \times \ldots \times o_n \to o$ and all $\phi_j \in \mathcal{H}^*_{o_j}$, $j = 1, \ldots, n$;

- $\neg\phi \in \mathcal{H}^*_o$, *for all* $o \in O$, *for all* $\phi \in \mathcal{H}^*_o$;
- $\bigwedge \Phi \in \mathcal{H}^*_o$, *for all* $o \in O$, *for all* $\Phi \subseteq \mathcal{H}^*_o$.

The satisfaction relation $\models$ *is defined as follows: for all* $o \in O$, $a \in A_o$ *and* $\phi \in \mathcal{H}^*_o$

- $a \models < H > \phi_1 \cdots \phi_n$ *iff there exist* $e^* \in H$, $a_1, \ldots, a_n$ *of appropriate sort such that* $A \models e^*[a, a_1, \ldots, a_n]$ *and* $a_k \models \phi_k$ *for* $k = 1, \ldots, n$;
- $a \models \neg\phi$ *iff* $a \not\models \phi$;
- $a \models \bigwedge \Phi$ *iff* $a \models \phi$ *for all* $\phi \in \Phi$. □

From now on, we shall always omit the type information on the observational formulas.

Def. 2.3 *Given an observational structure OS with similarity law* $\mathcal{S}$ *and a family of pattern sets* $\mathcal{H}$, $\mathcal{S}$ *is* represented *by* $\mathcal{H}$ *if:*

- *for all* $e \in Exp$ *there exist a unique* $H(e) \in \mathcal{H}$ *and a unique* $\overline{e}[x, y_1, \ldots, y_n] \in H(e)$ *such that* $e = \overline{e}[x, v_1, \ldots, v_n]$ *for some* $v_1, \ldots, v_n$;
- *for all* $e, e' \in Exp$, *for all A-families* R, $e\ \mathcal{S}(R)\ e'$ *iff* $H(e) = H(e')$ *and* $\exists v_1, \ldots, v_n, v'_1, \ldots, v'_n$ *such that* $e = \overline{e}[x, v_1, \ldots, v_n]$, $e' = \overline{e'}[x, v'_1, \ldots, v'_n]$ *with* $v_i\ R\ v'_i$ *for* $i = 1, \ldots, n$, *where* $H(e)$, $H(e')$, $\overline{e}$ *and* $\overline{e'}$ *are the pattern sets and the patterns associated respectively with e and e′ by the above property.* □

We can now state the main result, i.e. informally, *for any given representation* $\mathcal{H}$ *of* $\mathcal{S}$, the testing structure (see definition 1.13) $OS_{\mathcal{H}^*}$ having as experiments $\mathcal{H}^*$ originates the same maximum o-relation as OS does. For this we use the characterization of the maximum observational relation as a limit of a (transfinite) sequence of approximations introduced in section 1.4. We define for each ordinal λ a testing structure $OS_{\mathcal{H}^\lambda}$ having as experiments the set $\mathcal{H}^\lambda$ of formulas in $\mathcal{H}^*$ having "depth" smaller than λ; indicating by $\sim_{\mathcal{H}^\lambda}$ the maximum observational relation for $OS_{\mathcal{H}^\lambda}$, we show that for all $\lambda \in \mathcal{O}$, $\cong_\lambda = \sim_{\mathcal{H}^\lambda}$.

Since $\mathcal{H}^* = \cup_{\lambda \in \mathcal{O}} \mathcal{H}^\lambda$ we have clearly

$$\sim_{\mathcal{H}^*} = \bigcap_{\lambda \in \mathcal{O}} \sim_{\mathcal{H}^\lambda} = \bigcap_{\lambda \in \mathcal{O}} \cong_\lambda = \sim.$$

Because two experiments are similar in a testing structure iff they are logically equivalent, $\mathcal{H}^*$ is an observational logic characterizing the maximum o-relation of OS: two objects are equivalent iff they satisfy the same set (modulo logical equivalence) of formulas in $\mathcal{H}^*$. The formal statements follow.

Def. 2.4 *The* Depth *of the observational formulas is inductively defined by*

- $\text{Depth}(< H > \phi_1 \cdots \phi_n) = 1 + \sup_{i=1,\ldots,n} \text{Depth}(\phi_i)$;
- $\text{Depth}(\neg\phi) = \text{Depth}(\phi)$;

- $\mathrm{Depth}(\bigwedge \Phi) = \sup\{\mathrm{Depth}(\phi) \mid \phi \in \Phi\}$. $\square$

Def. 2.5 *For all ordinal numbers λ, $OS_{\mathcal{H}^\lambda}$ denotes the testing structure*

$$(\Sigma, A, O, \mathcal{H}^\lambda, \mathcal{ID}, \mathcal{P}),$$

where $\mathcal{H}^\lambda = \{\phi \mid \phi \in \mathcal{H}^* \wedge \mathrm{Depth}(\phi) \leq \lambda\}$ and $\sim_{\mathcal{H}^\lambda}$ its maximum o-relation. Moreover $OS_{\mathcal{H}^*}$ denotes the testing structure $(\Sigma, A, O, \mathcal{H}^*, \mathcal{ID}, \mathcal{P})$, and $\sim_{\mathcal{H}^*}$ its maximum o-relation. $\square$

Theorem 2.6 (Generalized Hennessy-Milner Theorem)
Let OS be an observational structure with similarity law $\mathcal{S}$ and $\mathcal{H}$ a family of pattern sets s.t. $\mathcal{S}$ is represented by $\mathcal{H}$. Then the following facts hold:

i) for all ordinal numbers λ, $\cong_\lambda = \sim_{\mathcal{H}^\lambda}$;

ii) $\sim = \sim_{\mathcal{H}^*}$, i.e., for all $o \in O$ and all $a', a'' \in A_o$ $a' \sim_o a''$ iff for all $\phi \in \mathcal{H}^*$ ($\phi[a']$ holds iff $\phi[a'']$ holds);

iii) if for all $o \in O$, $a \in A_o$ the set $\{e \mid e \in Exp, A \models e[a]\}$ is finite, then i) and ii) hold for the subset of $\mathcal{H}^*$ with finitary conjunctions.

Proof. See section 2.4. $\square$

We stress that the theorem does not assert that the Hennessy-Milner characterization holds for one particular class of observational formulas; it gives instead conditions for such a result to hold. Many observational structures have a representable $\mathcal{S}$; indeed, if $\mathcal{S}(\sim_{OS})$ is an equivalence relation, then $\mathcal{S}$ is represented by the family of the equivalence classes of *Exp* w.r.t. $\mathcal{S}(\sim_{OS})$. Moreover $\mathcal{S}$ may be represented by many different $\mathcal{H}$. Clearly, we are interested in the cases in which each $H \in \mathcal{H}$ is finite and the definition of $\mathcal{H}$ itself "does not depend on $\sim$"; notice that this is the case of all the examples given in the paper.

2.3 Applications

Distributed Bisimulation for *CCS* We show the treatment of *distributed bisimulation* for a *CCS*-like language (see [CH]) using observational structures. We show that not only the basic definition is an instance of our schema, but also that we have a characterization of the maximum distributed bisimulation by a corresponding HM logic.

We consider, as in [CH], a variation *dCCS* of *CCS0* obtained by replacing the predicate $\longrightarrow$ by

$$- \xrightarrow{-} (-, -) \colon be \times act \times be \times be$$

defined by the following inductive rules:

$$\frac{}{a \cdot b \xrightarrow{a} (b, b)}$$

$$\frac{b \xrightarrow{a} (b', b'')}{b + b_1 \xrightarrow{a} (b', b'')} \qquad \frac{b \xrightarrow{a} (b', b'')}{b_1 + b \xrightarrow{a} (b', b'')}$$

$$\frac{b \xrightarrow{a} (b', b'')}{b|b_1 \xrightarrow{a} (b', b''|b_1)} \qquad \frac{b \xrightarrow{a} (b', b'')}{b_1|b \xrightarrow{a} (b', b_1|b'')}.$$

We refer to [CH] for a detailed discussion of how this predicate can be used to model distribution; here we just recall that $b \xrightarrow{a} (b', b'')$ models the fact that b can perform the action a and produce what are called *the local residual* b' and *the concurrent residual* b''.

Notice that $dCCS$ does not model handshaking communication (it is possible to extend the definition of distributed bisimulation to handle communication but here for simplicity we omit its treatment; all the results shown in this section apply also in such cases).

The observational structure for distributed bisimulation is:

$$\mathcal{DCCS} = (\Sigma_{dCCS}, dCCS, \{be\}, Exp, \mathcal{S}, \mathcal{P}_{dCCS}),$$

where $\mathcal{P}_{dCCS}$ is the propagation law associated with the algebra $dCCS$ (see sec. 1.3), $Exp = \{x \xrightarrow{a} (b', b'') \mid a\colon act, b', b''\colon be\}$, and for all R

$$x \xrightarrow{a} (b', b'')\ \mathcal{S}(R)\ x \xrightarrow{a'} (b'_1, b'_2) \quad \text{iff} \quad a = a', b'\ R\ b'_1, b''\ R\ b'_2.$$

$\mathcal{S}$ is represented by $\mathcal{H} = \{H_a \mid a\colon act\}$, where $H_a = \{x \xrightarrow{a} (y_1, y_2)\}$. The observational formulas introduced by H_a are generated by $\neg$, $\bigwedge$ and the modal combinators $< a > \phi_1\phi_2$, where $b \models < a > \phi_1\phi_2$ iff for some b', b'' we have that $dCCS \models x \xrightarrow{a} (b', b'')$ and $b' \models \phi_1$, $b'' \models \phi_2$. For example,

$$a' \cdot \mathbf{nil} | a'' \cdot \mathbf{nil} \not\sim a' \cdot a'' \cdot \mathbf{nil} + a'' \cdot a' \cdot \mathbf{nil}$$

since they are distinguished by the formula

$$< a' > (< a'' > true\ true)(< a'' > true\ true)$$

which the second behaviour satisfies, while the first does not.

Branching Bisimulation Let TS be an algebraic transition system, i.e., an algebra on a signature Σ containing sorts be, act and predicates $\longrightarrow\colon be \times act \times be$ and $\Longrightarrow\colon be \times be$. We assume that Σ includes an operation $\tau\colon \to act$. The interpretation of $\Longrightarrow$ in TS is defined by the following inductive rules

$$\frac{}{b \Longrightarrow b} \qquad \frac{b \xrightarrow{\tau} b' \quad b' \Longrightarrow b''}{b \Longrightarrow b''}.$$

We simply write $b \Longrightarrow b' \xrightarrow{a} b''$ for $b \Longrightarrow b' \wedge b' \xrightarrow{a} b''$.

The following definition is just the rephrasing in our notation of the original definition of branching bisimulation as given in [GW, DV].

Def. 2.7 *A TS-family R is a* branching bisimulation *if it is symmetric and satisfies the following property (called* transfer property*):*

- *if* $r\ R_{be}\ s$ *and* $r \xrightarrow{a} r'$, *then either* $a = \tau$ *and* $r'\ R_{be}\ s$, *or* $\exists s_1, s'$ *such that* $s \Longrightarrow s_1 \xrightarrow{a} s'$, $r\ R_{be}\ s_1$ *and* $r'\ R_{be}\ s'$.
- R_{act} *is the identity on* TS_{act}.
- $R_{\longrightarrow} \subseteq \longrightarrow^{TS}$;
- $R_{\Longrightarrow} \subseteq \Longrightarrow^{TS}$. □

Exactly the same notion can be obtained by using the observational structure

$$\mathcal{BR} = (\Sigma, TS, \{be\}, Exp, \mathcal{S}, \mathcal{P}_{TS}),$$

where

- $\mathcal{P}_{TS}$ is the propagation law associated with the algebra TS (see sec. 1.3);
- $Exp = \{x = b, x = b \wedge x \xrightarrow{a} b', x \Longrightarrow b \xrightarrow{a} b', x \Longrightarrow b \wedge b = b' \mid a{:}\ act, b, b'{:}\ be\}$;
- for all R, $\mathcal{S}(R)$ is the equivalence closure of the relation defined by:

$$\begin{array}{rcll} x = b_1 \wedge x \xrightarrow{\tau} b_1' & \mathcal{S}(R) & x = b_2' & \text{for } b_1'\ R\ b_2' \\ x = b_1 \wedge x \xrightarrow{a} b_1' & \mathcal{S}(R) & x \Longrightarrow b_2 \xrightarrow{a} b_2' & \text{for } b_1\ R\ b_2,\ b_1'\ R\ b_2' \\ x \Longrightarrow b_1 \xrightarrow{\tau} b_1' & \mathcal{S}(R) & x \Longrightarrow b_2' \wedge b_2 = b_2' & \text{for } b_1\ R\ b_2',\ b_1'\ R\ b_2'. \end{array}$$

Notice that this structure is represented by a family of pattern sets, while there exist simpler observational structures, whose maximum o-relation is the maximum branching bisimulation, but they cannot be represented by any reasonable family of pattern sets (and hence cannot be used to generate a corresponding logic from our result).

Fact 2.8 *R is a branching bisimulation iff it is an o-relation for* $\mathcal{BR}$.

Proof. The proof requires some details but consists of routine checks. □

It is easily seen that a representation of $\mathcal{S}$ is $\mathcal{H} = \{H_a \mid a{:}\ act\}$ defined as follows:

$$\begin{array}{rcl} H_\tau & = & \{x = y_1 \wedge y_1 \xrightarrow{\tau} y_2, x = y_2, x \Longrightarrow y_1 \xrightarrow{\tau} y_2, x \Longrightarrow y_1 \wedge y_1 = y_2\}, \\ H_a & = & \{x = y_1 \wedge y_1 \xrightarrow{a} y_2, x \Longrightarrow y_1 \xrightarrow{a} y_2\} \quad \text{for all } a \neq \tau. \end{array}$$

The observational formulas introduced by $\mathcal{H}$ are hence generated by $\neg$, $\bigwedge$ and the modal combinators $<\tau>\ \phi_1\phi_2$ and $<a>\ \phi_1\phi_2$, for $a \neq \tau$, where:

- $b \models <\tau>\ \phi_1\phi_2$ iff either of the following holds:
  - $b \models \phi_1$ and there exists b' such that $b \xrightarrow{\tau} b'$ and $b' \models \phi_2$;
  - $b \models \phi_2$;

- there exists b', b'' such that $b \Longrightarrow b' \xrightarrow{\tau} b''$ and $b' \models \phi_1$, $b'' \models \phi_2$;
- there exists b' such that $b \Longrightarrow b'$ and $b' \models \phi_1 \wedge \phi_2$;

- $b \models <a> \phi_1\phi_2$ for $a \neq \tau$ iff either of the following holds:
  - $b \models \phi_1$ and there exists b' such that $b \xrightarrow{a} b'$ and $b' \models \phi_2$;
  - there exists b', b'' such that $b \Longrightarrow b' \xrightarrow{a} b''$ and $b' \models \phi_1$, $b'' \models \phi_2$.

This observational logic provided by our general approach is similar but quite less intuitive than the one originally given in [DV]; we think that the one of [DV] can be seen as an optimization of ours as it should be, since our logic is generated in a canonical way. The relationship between the two sets of formulas can be the subject of some interesting investigations.

However it is interesting to note that the modalities are similar to those in [DV], again supporting the feeling that our approach captures correctly the intuition behind.

2.4 Proof of the Generalized Hennessy-Milner Theorem

Proof of Theorem 2.6
Proof of i) We prove that $\cong_\lambda = \sim_{\mathcal{H}^\lambda}$ by induction on λ. In the following $\sim_{\mathcal{H}^\lambda}$ is simply written $\sim^\lambda$.

$\boxed{\lambda = 0}$ Now $\cong_0 = \mathcal{P}(\{A_o^2\}_{o\in O})$; since all formulas in $\mathcal{H}^0$ are combinations of $\neg$, $\bigwedge$ and $true$, then for all $\phi \in \mathcal{H}^0$, either for all a $A \models \phi[a]$ or for all a $A \not\models \phi[a]$, hence $\sim^0 = \mathcal{P}(\{A_o^2\}_{o\in O})$.

$\boxed{\lambda = \gamma + 1}$ We prove both inclusions.

$\underline{\cong_\lambda \subseteq \sim^\lambda}$ We show that $\cong_\lambda$ is an o-relation for $OS_{\mathcal{H}^\lambda}$.

- Suppose $a' \cong_\lambda a''$, for $a', a'' \in A_o$, $o \in O$. If $A \models \phi[a']$, with $\phi \in \mathcal{H}^\lambda$, then by cases on ϕ we show that $A \models \phi[a'']$.
  - If $\phi \in \mathcal{H}^\gamma$, since $\cong_\lambda \subseteq \cong_\gamma$ and by inductive hypothesis $\cong_\gamma = \sim^\gamma$, then $a' \sim^\gamma a''$; hence, by definition of $\sim^\gamma$, $A \models \phi[a'']$.
  - Suppose
  $$\phi = <H> \phi_1 \cdots \phi_n,$$
  with $\phi_k \in \mathcal{H}^\gamma$, for $k = 1, \ldots, n$.
  Since $A \models \phi[a']$, then there exist $\overline{e} \in H$, $v_1, \ldots, v_n$ such that $A \models \overline{e}[a', v_1, \ldots, v_n]$ and $A \models \phi_k[v_k]$ for $k = 1, \ldots, n$. But since $a' \cong_\lambda a''$, there exists $e' \in \mathit{Exp}$ such that $A \models e'[a'']$ and $\overline{e}[x, v_1, \ldots, v_n]\, \mathcal{S}(\cong_\gamma)\, e'$.
  Then since $\mathcal{H}$ represents $\mathcal{S}$, there exist $\overline{\overline{e}} \in H$, $v_1', \ldots v_n'$ s.t. $v_i \cong_\gamma v_i'$ and $e' = \overline{\overline{e}}[x, v_1', \ldots, v_n']$. By inductive hypothesis $\cong_\gamma = \sim^\gamma$, hence $v_i \sim^\gamma v_i'$ and, by definition of $\sim^\gamma$, since $\phi_k \in \mathcal{H}^\gamma$ and $A \models \phi_k[v_k]$, $A \models \phi_k[v_k']$. Hence $A \models \phi[a'']$.
  - The other two cases for ϕ are routine.

- Since $\cong_\lambda|_O \subseteq \sim^\lambda|_O$ and $\mathcal{P}$ is monotonic, we get $\mathcal{P}(\cong_\lambda|_O) \subseteq \mathcal{P}(\sim^\lambda|_O)$.
- The condition on predicates is verified similarly to the previous one.

$\underline{\cong_\lambda \supseteq \sim^\lambda}$ We show that for all $o \in O$ we have $(\cong_\lambda)_o \supseteq (\sim^\lambda)_o$.
Recall that $\cong_\lambda = \mathcal{F}(\cong_\gamma)$, and let $a' \sim^\lambda a''$. We have to show that if $A \models e'[a']$ with $e' \in Exp$, then there exists $e'' \in Exp$ such that $e' \mathcal{S}(\cong_\gamma) e''$ and $A \models e''[a'']$. Since $\mathcal{H}$ is a representation of $\mathcal{S}$, there exist $H \in \mathcal{H}$, $\overline{e} \in H$, $v_1, \ldots, v_n$ such that $e' = \overline{e}[x, v_1, \ldots, v_n]$. Let

$$\phi = < H > \mathit{true} \ \cdots \ \mathit{true},$$

since $\phi \in \mathcal{H}^\lambda$ and $a' \sim^\lambda a''$, we have that $A \models \phi[a'']$, i.e. for some $e \in H$ there exist $v'_1, \ldots, v'_n$ s.t.

$$A \models e[a'', v'_1, \ldots, v'_n].$$

Assume

$$\mathcal{V} = \{(v'_1, \ldots, v'_n) \mid A \models e[a'', v'_1, \ldots, v'_n] \text{ for some } e \in H\}.$$

We know that $\mathcal{V} \neq \emptyset$.
Suppose by contradiction that for all $v' = (v'_1, \ldots, v'_n) \in \mathcal{V}$, $v'_{i_{v'}} \not\cong_\gamma v_{i_{v'}}$ for some $i_{v'}, 1 \leq i_{v'} \leq n$, then by the inductive hypothesis there exists $\phi_{i_{v'}} \in \mathcal{H}^\gamma$ s.t. $A \models \phi_{i_{v'}}[v_{i_{v'}}]$ and $A \not\models \phi_{i_{v'}}[v'_{i_{v'}}]$. Let

$$\psi_i = \bigwedge_{\substack{v' \in \mathcal{V} \\ i_{v'} = i}} \phi_{i_{v'}},$$

now $\psi_i \in \mathcal{H}^\gamma$, so $\psi = < H > \psi_1 \cdots \psi_n$ is in $\mathcal{H}^\lambda$.
Moreover $A \models \psi[a']$ (substitute v_i for y_i) and $A \not\models \psi[a'']$, since for all $(v'_1, \ldots, v'_n) \in \mathcal{V}$ $A \not\models \phi_{i_{v'}}[v'_{i_{v'}}]$. This implies $a' \not\sim^\lambda a''$, contradiction.
So for some $(v'_1, \ldots, v'_n) \in \mathcal{V}$ we have that $A \models e[a'', v'_1, \ldots, v'_n]$ for some $e \in H$ with $v_i \cong_\gamma v'_i$. Hence, there exists $\overline{\overline{e}} \in H$ s.t. $A \models \overline{\overline{e}}[a'', v'_1, \ldots, v'_n]$ and since $\mathcal{H}$ represents $\mathcal{S}$, $\overline{e}[x, v_1, \ldots, v_n]$ $\mathcal{S}(\cong_\gamma)$ $\overline{\overline{e}}[x, v'_1, \ldots, v'_n]$, which concludes the proof.

$\boxed{\lambda \textbf{ limit ordinal}}$ This case is routine.

Proof of ii) Immediate consequence of i). **Proof of iii)** If for all $o \in O$, $a \in A_o$ the set $\{e \mid e \in Exp, A \models e[a]\}$ is finite, then also the set $\mathcal{V}$ used in the proof of i) is finite and so each ψ_i is a finite conjunction; and this is the only point of the proof of i) where infinite conjunctions are needed. □

3 Observational Specifications

Here we briefly illustrate the use of our formalism for algebraic specifications integrating the specifications of processes, data types and functions.

An *observational specification* is a particular case of observational structure in which we make explicit use of an algebraic specification SP; moreover the algebra component of the structure is the initial model of SP.

An *algebraic specification* SP is a couple (Σ, Ax) where Σ is a signature and Ax a set of *positive conditional axioms.* Positive conditional axioms are formulas of the form $\wedge_{i \in I} \alpha_i \supset \alpha$, where α_i, α are *atoms*, and atoms have form either $t_1 = t_2$ or $p(t_1, \ldots, t_n)$, with the t_i's terms of appropriate sort and p predicate symbol. A Σ-algebra which satisfies all the axioms in Ax is said a *model* of SP. Due to the restriction on the form of the axioms of the specifications there exists always an *initial model* I_{SP} which is term-generated and characterized by:

$$I_{SP} \models \alpha \quad \text{iff} \quad SP \vdash \alpha,$$

where $\vdash$ denotes the sound and complete system for many-sorted conditional deduction (see e.g., [GM]). In the following, given $t \in T_\Sigma$ we simply write t for the interpretation of t in I_{SP}.

Def. 3.1 *An observational specification is a* 5*-uple* $(SP, O, Exp, \mathcal{S}, \mathcal{P})$*, where* $SP = (\Sigma, Ax)$ *is a specification and* $(\Sigma, I_{SP}, O, Exp, \mathcal{S}, \mathcal{P})$ *is an observational structure.* $\Box$

In the case of observational specifications we define particular propagation and similarity laws derived by the axioms and introduce a canonical, we call initial, associated observational structure, with examples.

The free axiomatic propagation law provides the minimal propagation of the identifications on the observed elements to the whole structure (i.e., to all nonobserved sorts and all predicates) which preserves the algebraic structure and the validity of the specification axioms about non observed elements and predicates.

Def. 3.2 (Free Axiomatic Propagation Law) *Let* $SP = (\Sigma, Ax)$ *be a specification,* $\Sigma = (S, F, P)$ *and* $O \subseteq S$*. The* free axiomatic propagation law $\mathcal{FP}_{SP}$ *is defined as follows:*

- $t_1\, \mathcal{FP}_{SP}(R)_s\, t_2$ *iff* $SP + R \vdash^O t_1 = t_2$*, for all* $s \in S$*,*
- $(t_1, \ldots, t_n) \in \mathcal{FP}_{SP}(R)_p$ *iff* $SP + R \vdash^O p(t_1, \ldots, t_n)$*, for all* $p \in P$*,*

where $SP + R \vdash^O$ *is a particular deductive system which we now define.*

Assume

$$Ax_O = Ax - \{\phi \in Ax \mid \phi = (cond \supset) t_1 = t_2, t_1, t_2 \colon o, o \in O\}$$

(the axioms of SP *which do not imply equalities between terms of observed sort),*

$$Eq_R = \{t_1 = t_2 \mid (t_1, t_2) \in R_o, o \in O\}$$

(equalities between terms of observed sorts present in R*), and let* $SP + R \vdash^O$ *be the deductive system with proper axioms* $Ax_O \cup Eq_R$ *obtained by deleting from* $\vdash$ *all inference rules by which we could prove equalities between terms of observed sort, ie, the system consisting of the following inference rules:*

$$\frac{}{t = t} \qquad \frac{t = u}{u = t} \qquad \frac{t = u \quad u = v}{t = v} \qquad\qquad t, u, v : s \notin O$$

$$\frac{t_1 = t'_1 \quad \ldots \quad t_n = t'_n}{f(t_1, \ldots, t_n) = f(t'_1, \ldots, t'_n)} \qquad f : s_1 \times \cdots \times s_n \to s \in F, s \notin O$$

$$\frac{p(t_1, \ldots, t_n) \quad t_1 = t'_1 \quad \ldots \quad t_n = t'_n}{p(t'_1, \ldots, t'_n)} \qquad p : s_1 \times \cdots \times s_n \in P$$

$$\frac{\phi}{\sigma\phi} \qquad \sigma \textit{ substitution}$$

$$\frac{\bigwedge_{i \in I} \phi_i \supset \psi \quad \{\phi_j \mid j \in J\}}{\bigwedge_{i \in I - J} \phi_i \supset \psi}.$$

□

Since SP has only positive conditional axioms $\mathcal{FP}_{SP}$ is monotonic; moreover for all $t', t'' : o$, $o \in O$, $SP + R \vdash^O t' = t''$ iff $t' \; R_o \; t''$, so that for $o \in O$ we have that $\mathcal{FP}_{SP}(R)_o = R_o$; thus def. 3.2 truly defines a propagation law.

Given an propagation law $\mathcal{P}$, we can canonically define a similarity law $\mathcal{S}(\mathcal{P})$ s.t. for all R we consider equivalent two experiments if and only if they at most differ for subcomponents which are related by $\mathcal{P}(R)$.

Def. 3.3 (Similarity Law generated by a Propagation Law)
*Given a Σ-algebra A, $O \subseteq S$ and $\mathcal{P} \in$ **P-law**(A, O), $\mathcal{S}(\mathcal{P})$ is defined as follows: for all e, $e' \in$ **Exp**(Σ, O) and all A-families R,*

> *$e' \, \mathcal{S}(\mathcal{P})(R) \, e''$ iff there exist $\overline{e}$, $\overline{\overline{e}} \in \mathcal{FOF}_\Sigma(X)$, $t'_1, \ldots, t'_n$, $t''_1, \ldots, t''_n$ ground terms of appropriate sorts s.t. $e' = \overline{e}[t'_1/x_1, \ldots, t'_n/x_n]$, $e'' = \overline{\overline{e}}[t''_1/x_1, \ldots, t''_n/x_n]$, $\overline{e}$ is equivalent to $\overline{\overline{e}}$, and for $i = 1, \ldots, n$ we have that $t'_i \; \mathcal{P}(R) \; t''_i$.* □

It is easy to see that $\mathcal{S}(\mathcal{P})$ is truly a similarity law.

Fact 3.4 *$\mathcal{FP}_{SP}$ propagates equivalences; for all propagation laws $\mathcal{P}$, if $\mathcal{P}$ propagates equivalences, then $\mathcal{S}(\mathcal{P})$ reflects equivalences.* □

Def. 3.5 *An* initial observational structure *is an observational specification of the form $(SP, O, Exp, \mathcal{FP}_{SP}, \mathcal{S}(\mathcal{FP}_{SP}))$, shortly denoted by (SP, O, Exp).* □

Example 1 ***FCCS*: *CCS*0 with functions** Here we define an extension of $CCS0$ including functions having arguments and/or results of sort behaviour by means of the initial observational structure $\mathcal{FCCS}$; for simplicity we consider functions having only one argument of sort behaviour and returning a behaviour.

The purpose of this example is to show that our framework allows to treat rather uniformly varieties of concurrent calculi (always in the spirit of CCS, i.e., defining transitions).

We first give the specification $CCS0\text{-}SP$ corresponding to $CCS0$.

spec $CCS0\text{-}SP$ $=$

  enrich Σ_{CCS0} **by**

  preds

    $IsAct$: act

  axioms

    – properties of the labels

    $\{IsAct(\alpha) \mid \alpha \in ACT - \{\tau\}\}$

    $\overline{\tau} = \tau \qquad \overline{\overline{a}} = a$

    – static properties

    $b + b' = b' + b$

    $(b + b') + b'' = b + (b' + b'')$

    $b \mid b' = b' \mid b$

    $(b \mid b') \mid b'' = b \mid (b' \mid b'')$

    – definition of the transition relation

    $b \xrightarrow{a} b'$

    $b \xrightarrow{a} b' \supset b + b'' \xrightarrow{a} b'$

    $b \xrightarrow{a} b' \supset b|b'' \xrightarrow{a} b'|b''$

    $b \xrightarrow{a} b_1 \wedge b' \xrightarrow{\bar{a}} b'_1 \wedge IsAct(a) \supset b|b' \xrightarrow{\tau} b_1|b'_1$

The initial model of $CCS0\text{-}SP$ restricited to Σ_{CCS0} is just the algebra $CCS0$ given in section 1.1. Notice that here the static properties allow us to simply define the transition relation (we need less rules than in 1.1).

spec $FCCS$ $=$

  enrich $CCS0\text{-}SP$ **by**

  sorts fun

  opns

    $-(-)$: $fun \times be \to be$

    $\mathbf{fix}\,-$: $fun \to be$

    Σ: $fun \to be$

    $\{F\colon \to fun \mid F \in FUN\}$

  axioms

    APP

    (1) $(\bigwedge_{tb:be} f(tb) = f'(tb)) \supset f = f'$

    (2) $\mathbf{fix}\, f = f\, \mathbf{fix}\, f$

    (3) $f(b') \xrightarrow{a} b'' \supset \Sigma f \xrightarrow{a} b''$

where FUN is a set of function symbols used to represent behaviour functions (for example, in the following we take as FUN a set of λ-expressions) and APP is a set of axioms defining the application operation $-(-)$ for all $F \in FUN$.

"**fix**" is the fixpoint operator, and Σ is the usual nondeterministic choice indexed on behaviours; so $\mathbf{fix}\,\lambda x.b(x)$ and $\Sigma\lambda x.b(x)$ are written in the usual CCS notation as $\mathbf{fix}\, x.b(x)$ and $\Sigma_{x \in CCS0_{be}} b(x)$ respectively. Notice that the introduction of a functional sort allows the definition of these two operators as algebraic operations of a signature.

Axiom (1) requires term-extensionality on functions, axioms (3) defines the transitions of Σf behaviours exactly as in *CCS*, while axiom (2) directly defines the fixpoint operator **fix** instead of giving the transitions of the behaviours built with it, as is usually done when defining *CCS*. Notice finally that *FCCS* differs from *CCS* just for the restriction and the relabelling operations; notice also that the elements of sort *fun* are *open behaviours*, while the elements of sort *be* correspond to *processes*, in the usual *CCS* terminology.

$$\mathcal{FCCS} = (FCCS, Exp, \{be\}),$$

where $Exp = \{x \xrightarrow{a} b \mid a{:}\, act, b{:}\, be\}$ is the set of experiments used in sec. 1.1 to define strong bisimulation.

Since $\sim_{\mathcal{FCCS}}$ is a congruence, we can define the semantic model FCCS $= I_{FCCS}/{\sim_{\mathcal{FCCS}}}$. Two functions are equivalent in FCCS iff when applied to strongly bisimilar behaviours they return strongly bisimilar behaviours.

FCCS restricted to Σ_{CCS0} coincides with the algebra $CCS0$ defined in 1.1; but note that here, for example, we have stright that $b + b' = b' + b$ holds in $CCS0$; while using the definition of section 1.1 we have to prove it.

Consider, for example, the two functions

$$\begin{array}{lll} f_1 & = & \lambda x.(\mathbf{fix}\, \lambda y.\alpha \cdot (x \mid y) + \beta \cdot \mathbf{nil}) + \beta \cdot \mathbf{nil} \\ f_2 & = & \lambda x.\, \mathbf{fix}\, \lambda y.\alpha \cdot (y \mid x) + \beta \cdot \mathbf{nil} \end{array}$$

they are equivalent in FCCS, and indeed $f_1 \sim_{\mathcal{FCCS}} f_2$. To prove it, we can show that $R = {\sim_{\mathcal{FCCS}}} \cup \mathcal{FP}_{FCCS}(\{(f_1(b), f_2(b))\} \mid b{:}\, be\})$ is an o-relation for $\mathcal{FCCS}$, and then by applying the axiom $(\bigwedge_{tb:be} f(tb) = f'(tb)) \supset f = f'$ we get $f_1\, \mathcal{FP}_{FCCS}\,(R) f_2$, and so $f_1 \sim_{\mathcal{FCCS}} f_2$.

Example 2: Maps from identifiers into behaviours The purpose of this example is to show that our theory allows to integrate the specification of abstract data types and of dynamic objects and in particular to consider processes as data types and to use them for building new compound types (see [AGR2] for a more extensive treatment of this aspect).

We define maps from identifiers into behaviours by means of an initial observational structure $\mathcal{MCCS}$. One can think of using a map of that kind, for example, for storing processes modelling the execution of some `UNIX`-like commands.

spec *MCCS* =
  enrich *CCS*0-*SP*+*ID* **by**
  sorts *map*
  opns
    $[]{:} \to map$
    $-(-){:}\, map \times id \to be$
    $-[-/-]{:}\, map \times be \times id \to map$
  axioms
    $[](id) = \mathbf{nil}$
    $m[b/id](id) = b$

$$eq(id, id') = false \supset m[b/id](id') = m(id')$$
$$m[b_1/id][b_2/id] = m[b_2/id]$$
$$eq(id, id') = false \supset m[b/id][b'/id'] = m[b'/id'][b/id]$$

where ID is some specification for identifiers including an equality operation eq.

$$\mathcal{MCCS} = (MCCS, \{be\}, Exp),$$

where $Exp = \{x \xrightarrow{a} b \mid a\colon act, b\colon be\}$ is the set of the experiments corresponding to strong bisimulation.

Consider for example the two maps

$$\begin{aligned} m_1 &= [][id_1 \to a \cdot \mathbf{nil}][id_2 \to \mathbf{nil} \,|\, \mathbf{nil}] \\ m_2 &= [][id_2 \to a \cdot \mathbf{nil}][id_1 \to a \cdot \mathbf{nil}][id_2 \to \mathbf{nil}] \end{aligned}$$

where $ID \vdash eq(id_1, id_2) = false$; it is easy to see that

$$R = \mathcal{FP}_{MCCS}(\{(\mathbf{nil}, \mathbf{nil} \mid \mathbf{nil})\} \cup (Id_{I_{MCCS}})_{be})$$

is an o-relation, and hence that $MCCS + R \vdash^{\{be\}} m_1 = m_2$, ie, $m_1 \sim_{\mathcal{MCCS}} m_2$.

Acknowledgement. Comments and suggestions by competents and careful referees have helped us to amend and improve the presentation.

References

[A1] Abramsky S. "Observation Equivalence as a Testing Equivalence", *J TCS Vol. 53*, North Holland, 1987.

[A2] Abramsky S. "The Lazy Lambda Calculus", (Turner D. A. editor) *Research Topics in Functional Programming*, Addison-Wesley 1990.

[AD] Arnold A.; Dicky A. "An Algebraic Characterization of Transition Systems Equivalences", *Information and Computation*, Vol. 82, No. 2, August 1989, Academic Press.

[AGR1] Astesiano E.; Giovini A.; Reggio G. "Generalized Bisimulation in Relational Specifications", *STACS '88* (Cori R.; Wirsing M. eds.), Berlin, Springer-Verlag, 1988, (Lecture Notes in Computer Science n. 294).

[AGR2] Astesiano E.; Giovini A.; Reggio G. "Data in a Concurrent Environment", *Proc. of Concurrency '88* (Vogt F. ed.), Berlin, Springer-Verlag, 1988, (Lecture Notes in Computer Science n. 335).

[AR] Astesiano E.; Reggio G. "SMoLCS Driven Concurrent Calculi", *Proc. TAPSOFT '87, Vol. I*, (Ehrig H., Kowalsky R., Levi G., Montanari U. eds.), Berlin, Springer-Verlag, 1987, (Lecture Notes in Computer Science n. 249).

[AW] Astesiano E.; Wirsing M. "Bisimulation in Algebraic Specifications", *Proc. of the Colloquium on Resolution of Equations in Algebraic Structures* (Nivat M.; Ait-Kaci H. eds.), San Diego, Academic Press, 1989.

[BB] Berry G.; Boudol G. "The Chemical Abstract Machine", *Proc. POPL '90*, ACM Press, 1990.

[BK] Bergstra J.A.; Klop J.W. "Algebra of Communicating Processes", *Proc. CWI Symp. Math and Comp. Sci*, Amsterdam, North-Holland, 1986.

[C] Castellani I. "Bisimulation and Abstraction Homomorphisms", *J. Comp. System Sci.* 34, 1987.

[CH] Castellani I.; Hennessy M. "Distributed Bisimulations", *Journal of the ACM*, October 1989, Vol. 36, n. 4, New York, ACM Press.

[DH] De Nicola R.; Hennessy M. "Testing Equivalence for Processes", *J TCS* Vol. 34, North Holland, 1984.

[DV] De Nicola R.; Vaandrager F. "Three logics for Branching Bisimulation", *Proc. of 5th Symposium on Logics in Computer Science (LICS '90)*, IEEE Computer Society Press, 1990.

[EPBRDG] Ehrig H., Parisi-Presicce F.; Boehm P.; Rieckoff C.; Dimitrovici C.; Grosse-Rhode M. "Algebraic Data Type and Process Specifications Based on Projection Spaces", *Recent Trends in Data Type Specifications*, (Sannella D.; Tarlecki A. eds.), Berlin, Springer Verlag 1988, (Lecture Notes in Computer Science n. 332).

[FM] Ferrari G. L.; Montanari U. "Towards the Unification of Models for Concurrency", *Proc. CAAP'90*, Berlin, Springer-Verlag, 1990, (Lecture Notes in Computer Science n. 431).

[GM] Goguen J.; Meseguer J. "Models and Equality for Logical Programming", *Proc. of TAPSOFT '87, Vol. II*, (Ehrig H., Kowalsky R., Levi G., Montanari U. eds.), Berlin, Springer Verlag 1987, (Lecture Notes in Computer Science n. 250).

[GR] Giovini A.; Reggio G. "Bisimulation Models in Algebraic Specifications", *Proc. of the Third Italian Conference on Theoretical Computer Science*, (Bertoni A.; Böhm C.; Miglioli P. eds.), World Scientific, 1989.

[GV] Groote J. F.; Vaandrager F. "Structured Operational Semantics and Bisimulation as a Congruence" *Proc. ICALP '89*, Berlin, Springer Verlag, 1989 (Lecture Notes in Computer Science n. 372).

[GW] van Glabbeek R. J. ; Weijland W. P. "Branching Time and Abstraction in Bisimulation Semantics", *Information Processing'89* (G. X. Ritter ed.), North Holland, pp 613-618, 1989.

[HM] Hennessy M.; Milner R. "Algebraic Laws for Nondeterminism and Concurrency", *Journal of the ACM*, Vol. 26, n. 1, New York, ACM Press, 1985

[KLP] Kasanjan S.; Labella A.; Pettorossi A. "Observers, Experiments and Agents: a Comprehensive Approach to Parallelism", *Semantics of Systems of Concurrent Processes, Proc. of LIPT Spring School on Theoretical Computer Science* (Guessarian I. ed.), Berlin, Springer Verlag, 1990 (Lecture Notes in Computer Science n. 469).

[M] Meseguer J. *Rewriting as a Unified Model of Concurrency*, SRI Technical Report, 1990.

[Mi] Milner R. "Operational and Algebraic Semantics of Concurrent Processes", *Handbook of Theoretical Computer Science, Vol. B: Formal Models and Semantics*, Elsevier and MIT Press, 1990.

[P] Park D. "Concurrency and Automata on Infinite Sequences", *Proc. 5th GI Conference*, Berlin, Springer Verlag, 1981 (Lecture Notes in Computer Science n. 104).

[S] Stirling C. *Comparing Linear and Branching Time Temporal Logics*, LFCS Report, Univ. of Edinburgh, 1987.

[T] Thomsen B., "A Calculus of Higher-Order Communicating Systems", *Proc. POPL Conference*, 1989.

[Wa] Walker D. *Bisimulation equivalence and divergence in CCS*, LFCS Report, Univ. of Edinburgh, 1987.

[Wi] Winskel G. "Synchronization Trees", *J TCS*, May 1985.