DOI 10.1007/s00165-014-0327-6
BCS © 2014
Formal Aspects of Computing (2015) 27: 375–395

Formal Aspects of Computing

Refinement in hybridised institutions

Alexandre Madeira[1], Manuel A. Martins[2], Luís S. Barbosa[1], and Rolf Hennicker[3]

[1] HASLab INESC TEC and Univ. Minho, Braga, Portugal
[2] CIDMA-Center for R&D in Mathematics and Applications, Department of Mathematics, Univ. Aveiro, Aveiro, Portugal
[3] Ludwig-Maximilians-Universität München, Munich, Germany

Abstract. Hybrid logics, which add to the modal description of transition structures the ability to refer to specific states, offer a generic framework to approach the specification and design of reconfigurable systems, i.e., systems with reconfiguration mechanisms governing the dynamic evolution of their execution configurations in response to both external stimuli or internal performance measures. A formal representation of such systems is through transition structures whose states correspond to the different configurations they may adopt. Therefore, each node is endowed with, for example, an algebra, or a first-order structure, to precisely characterise the semantics of the services provided in the corresponding configuration. This paper characterises equivalence and refinement for these sorts of models in a way which is independent of (or parametric on) whatever logic (propositional, equational, fuzzy, etc) is found appropriate to describe the local configurations. A Hennessy–Milner like theorem is proved for hybridised logics.

Keywords: Hybridisation, Bisimulation, Refinement

1. Introduction

This paper discusses equivalence and refinement of *structured* transition systems. Or, to put it in another way, of models of specifications written in *hybridised* logics. These two qualifiers entail the need for a word of explanation. States in a *structured* transition system are endowed with a specific structure (e.g., algebraic, first order, etc.). In the development of software systems, one may think of such sort of states as (local) specifications of individual system configurations. The global transition structure, on the other hand, defines how the software evolves from a configuration to another. Such systems are called *reconfigurable* in the sense that they behave differently in different modes of operation (*configurations*) and commute between them along their lifetime.

At present, reconfigurable software is the norm than the exception: a typical, everyday example is provided by cloud based applications that elastically react to client demand levels, for example by allocating new server units to meet higher rates of service requests. Modern cars offer a second example: in each of them hundreds of electronic control units must operate in different modes, depending on the current situation—such as driving on a highway or in town where different speed regulations are applied. Switching between these modes is a typical example of a dynamic reconfiguration. Actually, reconfigurability [SC11], together with related issues like self-adaptation or context-awarness, became a main research topic, in the triple perspective of foundations, methods and technologies.

Correspondence and offprint requests to: A. Madeira, e-mail: amadeira@inesctec.pt

Specifications of this sort of systems, as discussed in [MFMB11], should be able to make assertions both about the transition dynamics and, locally, about each particular configuration. This leads to the adoption of hybrid logic [AtC06, Bra10], which adds to the modal description of transition structures the ability to refer to specific states, as the specification *lingua franca* for reconfigurable systems.

An elementary example to be discussed later in the paper (see Example 5.3) is that of a storing system equipped with a *read* operation which retrieves the *first* or the *last* element stored depending on the current execution mode. Reconfiguration between such modes is achieved by a control event, *shift*. The properties of each mode are specified equationally, whereas switching between them is encoded as a modality. Nominals provide a unique way to refer to each execution mode and its properties. Therefore, hybridised (equational) logic provides a suitable framework to develop the overall specification.

However, because specific problems may require specific logics to describe their configurations (e.g., equational, first-order, fuzzy, etc.), our approach is rooted on very general grounds. Instead of choosing a particular version of hybrid logic, we play with *hybridised* logics. The latter are the result of hybridising [MMDB11] whatever logic is found suitable for expressing and reasoning about the requirements at the configuration (static) level. This process, *hybridisation*, was characterised in [MMDB11, DM14] as well as in [Mad13]. To be completely general, it is framed in the context of the theory of institutions of J. Goguen and R. Burstall [GB92, Dia08], each logic (base and hybridised) treated abstractly as an *institution*. This is later taken as the *base* logic on top of which the characteristic features of hybrid logic, both at the level of syntax (i.e. modalities, nominals, etc.) and of the semantics (i.e. possible worlds), are developed.

In this context, the quest for suitable notions of *equivalence* and *refinement* between models of hybridised logic specifications becomes fundamental to the development of a design methodology for reconfigurable systems. Such is the purpose of the present paper. Its contributions are characterisations of bisimilarity and of two notions of refinement for (models of) specifications in hybridised logics. As discussed below, this requires a form of *elementary equivalence* [Hod97] between bisimilar states, as a generic formulation of the usual informal requirement that *truth remains invariant*. Clearly what *elementary equivalent* means in each case boils down to the way the satisfaction relation is defined for the base logic used in local configurations.

The choice of similarity and bisimilarity to base refinement and equivalence of (models of) reconfigurable systems seems quite standard as a fine grained approach to observational methods for systems comparison. The notion of bisimulation and the associated conductive proof method, which is now pervasive in Computer Science, originated in concurrency theory due to the seminal work of David Park [Par81] and Robin Milner in the quest for an appropriate definition of observational equivalence for communicating processes as understood in CCS [Mil89]. But the concept also arose independently in modal logic as a refinement of notions of homomorphism between algebraic models—see [San09] for an extensive historical account.

Contributions and organisation This paper extends preliminary work on refinement in hybridised institutions [MMB13] along three main directions: (1) the proof of a Hennessy–Milner result for hybridised logics, (2) the characterisation of two dual notions of refinement, forward and backward, and (3) a discussion on refinement of specifications. From a wider perspective, it is part of a broader research line on *logics for software reconfigurabilty* documented in [MMDB11, DM14] (for the hybridisation process), and [MFMB11, MNMB13, MMDB11, MMB13] (for the associated design methodology).

The paper is organised as follows: Sect. 2 recalls institutions as abstract characterisations of logics and provides a brief, and simplified, overview of the hybridisation method proposed in [MMDB11, DM14]. This forms the context for the paper's contribution. Then, Sect. 3 introduces a general notion of bisimulation for hybridised logics and Sect. 4 proves a Hennessy–Milner like theorem. Section 5 introduces notions of forward and backward refinement and discusses preservation of logic satisfaction under them. This discussion is extended to the specification level in Sect. 6. Finally, Sect. 7 concludes and points out directions for further research.

2. Background

2.1. Institutions

An *institution* is a category theoretic formalisation of a logical system, encompassing syntax, semantics and satisfaction. The concept was put forward by Goguen and Burstall, in the end of the seventies, in order to *"formalise the formal notion of logical systems"*, in response to the *"population explosion among the logical systems used in Computing Science"* [GB92].

The universal character of institutions proved effective and resilient as witnessed by the wide number of logics formalised in this framework. Examples range from the usual logics in classical mathematical logic (propositional, equational, first order, etc.), to the ones underlying specification and programming languages or used for describing particular systems from different domains. Well-known examples include *probabilistic logics* [BKI05], *quantum logics* [CMSS06], *hidden and observational logics* [BD94, BH06], *coalgebraic logics* [CÔ6], as well as logics for reasoning about *process algebras* [MR06], *functional* [ST12, SM09] and *imperative programing languages* [ST12].

The theory of institutions (see [Dia08] for an extensive account) was motivated by the need to abstract from the particular details of each individual logic and characterise generic issues, such as satisfaction and combination of logics, in very general terms. In Computer Science, this lead to the development of a solid *institution-independent specification theory*, on which structuring and parameterisation mechanisms, required to scale up software specification methods, are defined 'once and for all', irrespective of the concrete logic used in each application domain [Tar03]. The definition is recalled below (e.g., [GB92, Dia08]) and illustrated with a few examples to which we return later in the paper.

Definition 2.1 (*Institution*) An *institution*

$$I = \left(\mathrm{Sign}^I, \mathrm{Sen}^I, \mathrm{Mod}^I, (\models^I_\Sigma)_{\Sigma \in |\mathrm{Sign}^I|}\right)$$

consists of

- a category Sign^I whose objects are called *signatures* and arrows *signature* morphisms;
- a functor $\mathrm{Sen}^I : \mathrm{Sign}^I \to \mathbb{S}et$ giving for each signature a set whose elements are called *sentences* over that signature;
- a functor $\mathrm{Mod}^I : (\mathrm{Sign}^I)^{op} \to \mathbb{C}AT$, giving for each signature Σ a category whose objects are called Σ-*models*, and whose arrows are called Σ-*(model) homomorphisms*; each arrow $\varphi : \Sigma \to \Sigma' \in \mathrm{Sign}^I$, (i.e., $\varphi : \Sigma' \to \Sigma \in (\mathrm{Sign}^I)^{op}$) is mapped into a functor $\mathrm{Mod}^I(\varphi) : \mathrm{Mod}^I(\Sigma') \to \mathrm{Mod}^I(\Sigma)$ called a *reduct functor*, whose effect is to cast a model of Σ' as a model of Σ; when $M = \mathrm{Mod}^I(\varphi)(M')$ we say that M is the φ-reduct of M' and that M is an φ-expansion of M;
- a relation $\models^I_\Sigma \subseteq |\mathrm{Mod}^I(\Sigma)| \times \mathrm{Sen}^I(\Sigma)$ for each $\Sigma \in |\mathrm{Sign}^I|$, called the *satisfaction relation*,

such that for each morphism $\varphi : \Sigma \to \Sigma' \in \mathrm{Sign}^I$, the satisfaction condition

$$M' \models^I_{\Sigma'} \mathrm{Sen}^I(\varphi)(\rho) \text{ iff } \mathrm{Mod}^I(\varphi)(M') \models^I_\Sigma \rho \tag{1}$$

holds for each $M' \in |\mathrm{Mod}^I(\Sigma')|$ and $\rho \in \mathrm{Sen}^I(\Sigma)$. Graphically,

$$\begin{array}{ccc} \Sigma & & \mathrm{Mod}^I(\Sigma) \xrightarrow{\quad\models^I_\Sigma\quad} \mathrm{Sen}^I(\Sigma) \\ {\scriptstyle\varphi}\downarrow & & \uparrow{\scriptstyle \mathrm{Mod}^I(\varphi)} \qquad\qquad \downarrow{\scriptstyle \mathrm{Sen}^I(\varphi)} \\ \Sigma' & & \mathrm{Mod}^I(\Sigma') \xrightarrow[\quad\models^I_{\Sigma'}\quad]{} \mathrm{Sen}^I(\Sigma') \end{array}$$

Example 2.1 (The trivial institution *TRIV*) The simplest institution one can think of is *TRIV*. Its category of signatures, Sign^{TRIV}, is the *final category*, i.e., the category whose class of objects is the singleton set $\{*\}$ and morphisms reduce to the identity $1_*(*) = *$. Functor Sen^{TRIV} sends object $*$ into the empty set $\emptyset$ and morphism 1_* into the empty function. The models functor, Mod^{TRIV}, sends the signature $*$ to the final category. Since the set of sentences is empty, the satisfaction condition holds trivially.

Example 2.2 (Propositional Logic *PL*) A signature $Prop \in |\ \mathrm{Sign}^{PL}\ |$ in the institution *PL* is a set of symbols, called propositional variables, and a signature morphism is just a function $\varphi : Prop \to Prop'$. Therefore, Sign^{PL} coincides with the category $\mathbb{S}et$.

Functor Mod maps each signature $Prop$ to the category $\mathrm{Mod}^{PL}(Prop)$ and each signature morphism φ to the reduct functor $\mathrm{Mod}^{PL}(\varphi)$. Objects of $\mathrm{Mod}^{PL}(Prop)$ are functions $M : Prop \to \{\top, \bot\}$ and its morphisms functions $h : Prop \to Prop$ such that $M(p) = M'(h(p))$. Given a signature morphism $\varphi : Prop \to Prop'$, the reduct of a model $M' \in |\ \mathrm{Mod}^{PL}(Prop')\ |$, say $M = \mathrm{Mod}^{PL}(\varphi)(M')$, is defined, for each $p \in Prop$, as $M(p) = M'(\varphi(p))$.

Functor Sen^{PL} maps each signature $Prop$ to the set of propositional sentences $\mathrm{Sen}^{PL}(Prop)$ and each morphism $\varphi : Prop \to Prop'$ to the sentences' translation $\mathrm{Sen}^{PL}(\varphi) : \mathrm{Sen}^{PL}(Prop) \to \mathrm{Sen}^{PL}(Prop')$. The set $\mathrm{Sen}^{PL}(Prop)$ is the usual set of propositional formulas defined by the grammar

$$\rho ::= p \mid \rho \vee \rho \mid \rho \wedge \rho \mid \rho \Rightarrow \rho \mid \neg\rho$$

for $p \in Prop$. The translation of a sentence $\mathrm{Sen}^{PL}(\varphi)(\rho)$ is obtained by replacing each proposition of ρ by the respective φ-image.

Finally, for each $Prop \in \mathrm{Sen}^{PL}$, the satisfaction relation $\models^{PL}_{Prop}$ is defined as usual:

- $M \models^{PL}_{Prop} p$ iff $M(p) = \top$, for any $p \in Prop$,
- $M \models^{PL}_{Prop} \rho \vee \rho'$ iff $M \models^{PL}_{Prop} \rho$ or $M \models^{PL}_{Prop} \rho'$.

and similarly for the other connectives.

Example 2.3 (Equational logic *EQ*) Signatures in the institution *EQ* of equational logic are pairs (S, F) where S is a set of sort symbols and $F = \{F_{\underline{ar} \to s} \mid \underline{ar} \in S^*, s \in S\}$ is a family of sets of operation symbols indexed by arities $\underline{ar}$ (for the arguments) and sorts s (for the results). *Signature morphisms* map both components in a compatible way: they consist of pairs $\varphi = (\varphi^{\mathrm{st}}, \varphi^{\mathrm{op}}) : (S, F) \to (S', F')$, where $\varphi^{\mathrm{st}} : S \to S'$ is a function, and $\varphi^{\mathrm{op}} = \{\varphi^{\mathrm{op}}_{\underline{ar} \to s} : F_{\underline{ar} \to s} \to F'_{\varphi^{\mathrm{st}}(\underline{ar}) \to \varphi^{\mathrm{st}}(s)} \mid \underline{ar} \in S^*, s \in S\}$ is a family of functions mapping operation symbols according to their arities.

A model M for a signature (S, F) is an algebra interpreting each sort symbol s as a carrier set M_s and each operation symbol $\sigma \in F_{\underline{ar}} \to s$ as a function $M_\sigma : M_{\underline{ar}} \to M_s$, where $M_{\underline{ar}}$ is the product of the arguments' carriers. This interpretation is extended to (S, F)-terms $t = \sigma(t_1, \ldots, t_n)$, by $M_{\sigma(t_1,\ldots,t_n)} = M_\sigma(M_{t_1}, \ldots, M_{t_n})$. Model morphisms are homomorphisms of algebras, i.e., S-indexed families of functions $\{h_s : M_s \to M'_s \mid s \in S\}$ such that for any $m \in M_{\underline{ar}}$, and for each $\sigma \in F_{\underline{ar} \to s}$, $h_s(M_\sigma(m)) = M'_\sigma(h_{\underline{ar}}(m))$. For each signature morphism φ, the *reduct* of a model M', say $M = \mathrm{Mod}^{EQ}(\varphi)(M')$ is defined by $(M)_x = M'_{\varphi(x)}$ for each sort and function symbol x from the domain signature of φ. The models functor maps signatures to categories of algebras and signature morphisms to the respective reduct functors.

Sentences are universally quantified equations $(\forall X)t = t'$. Sentence translations along a signature morphism $\varphi : (S, F) \to (S', F')$, i.e., $\mathrm{Sen}^{EQ}(\varphi) : \mathrm{Sen}^{EQ}(S, F) \to \mathrm{Sen}^{EQ}(S', F')$, replace symbols of (S, F) by the respective φ-images in (S', F'). Functor Sen^{EQ} maps each signature to the set of universally quantified equations and each signature morphism to the respective sentences translation.

The satisfaction relation is the usual Tarskian satisfaction defined recursively on the structure of the sentences as follows:

- $M \models_{(S,F)} t = t'$ when $M_t = M_{t'}$,
- $M \models_{(S,F)} (\forall X)\rho$ when $M' \models_{(S,F \uplus X)} \rho$ for any inc-expansion M' of M where $inc : (S, F) \hookrightarrow (S, F \uplus X)$ is the inclusion morphism that enrich (S, F) with the set of variables X.

Example 2.4 (Propositional Fuzzy Logic MVL_L) Multi-valued logics [Got01] generalise classic logics by replacing, as their *truth domain*, the 2-element Boolean algebra by larger sets structured as *complete residuate lattices*. They were originally formalised as institutions in [ACEGG90] (see also [Dia11] for a recent reference).

A *residuate lattice* is a tuple $L = (\mathbf{L}, \leq, \wedge, \vee, \top, \bot, \otimes, \Rightarrow)$, where

- $(\mathbf{L}, \wedge, \vee, \top, \bot)$ is a lattice ordered by $\leq$, with carrier $\mathbf{L}$, with (binary) infimum ($\wedge$) and supremum ($\vee$), and bigest and smallest elements $\top$ and $\bot$;
- $\otimes$ is an associative binary operation such that, for any elements $x, y, z \in L$,
  - $x \otimes \top = \top \otimes x = x$,
  - $y \leq z$ implies that $(x \otimes y) \leq (x \otimes z)$,
  - the following Galois connection holds:

    $y \leq (x \Rightarrow z)$ iff $x \otimes y \leq z$.

A residuate lattice L is complete if any subset $S \subseteq \mathbf{L}$ has infimum and supremum, denoted by $\bigwedge S$ and $\bigvee S$, respectively.

Given a complete residuate lattice L, the institution MVL_L is defined as follows:

- MVL_L-signatures are PL-signatures, i.e., signatures are sets $Prop$ and morphisms are functions $\varphi : Prop \to Prop'$.
- Sentences of MVL_L consist of pairs (ρ, p) where p is an element of L and ρ is defined as a PL-sentence over the set of connectives $\{\Rightarrow, \vee, \top, \bot, \otimes\}$.
- A MVL_L-model M is a function $M : Prop \to L$,
- For any $M \in \mathrm{Mod}^{MVL_L}(Prop)$ and for any $(\rho, p) \in \mathrm{Sen}^{MVL_L}(Prop)$, the satisfaction relation is

  $$M \models^{MVL_L}_{Prop} (\rho, p) \text{ iff } p \leq (M \models \rho),$$

  where $M \models \rho$ is inductively defined as follows:
  - for any proposition $p \in Prop$, $(M \models p) = M(p)$),
  - $(M \models \top) = \top$,
  - $(M \models \bot) = \bot$,
  - $(M \models \rho_1 \star \rho_2) = (M \models \rho_1) \star (M \models \rho_2)$, for $\star \in \{\vee, \Rightarrow, \otimes\}$.

This institution captures many multi-valued logics in the literature. For instance, taking L as the Łukasiewicz arithmetic lattice over the closed interval $[0, 1]$, where $x \otimes y = 1 - max\{0, x + y - 1\}$ (and $x \Rightarrow y = min\{1, 1 - x + y\}$), yields the standard *propositional fuzzy logic*.

2.2. Hybridisation

The *hybridisation* method proposed in [MMDB11, DM14, Mad13], enriches an arbitrary institution $I = (\mathrm{Sign}^I, \mathrm{Sen}^I, \mathrm{Mod}^I, (\models^I_\Sigma)_{\Sigma \in |\mathrm{Sign}^I|})$ with the (modal) hybrid logic features and the corresponding Kripke semantics. The result is still an institution, $\mathcal{H}I$, called the *hybridisation of* I. The construction is revisited in the sequel. This overview is focussed on a simplified version, consisting of the quantifier-free and non-constrained version of the general method. The results in this paper are developed in the context of this simplified version, referred to as the hybridisation process.

The category of $\mathcal{H}I$-signatures. First of all the base signature is enriched with nominals and polyadic modalities. Therefore, the category of *I-hybrid signatures*, denoted by $\mathrm{Sign}^{\mathcal{H}I}$, is defined as the direct (cartesian) product of categories:

$$\mathrm{Sign}^{\mathcal{H}I} = \mathrm{Sign}^I \times \mathrm{Sign}^{REL}.$$

where REL is the sub-institution of (the institution of) single sorted first order logic, without non-constant operation symbols. Thus, signatures are triples $(\Sigma, \mathrm{Nom}, \Lambda)$, where $\Sigma \in |\mathrm{Sign}^I|$ and, in the REL-signature (Nom, Λ), Nom is a set of constants called *nominals* and Λ is a set of relational symbols called *modalities*; Λ_n stands for the set of modalities of arity n. Morphisms $\varphi \in \mathrm{Sign}^{\mathcal{H}I}((\Sigma, \mathrm{Nom}, \Lambda), (\Sigma', \mathrm{Nom}', \Lambda'))$ are triples $\varphi = (\varphi_{\mathrm{Sign}}, \varphi_{\mathrm{Nom}}, \varphi_{\mathrm{MS}})$ where $\varphi_{\mathrm{Sign}} \in \mathrm{Sign}^I(\Sigma, \Sigma')$, $\varphi_{\mathrm{Nom}} : \mathrm{Nom} \to \mathrm{Nom}'$ is a function and $\varphi_{\mathrm{MS}} = (\varphi_n : \Lambda_n \to \Lambda'_n)_{n \in \mathbb{N}}$ a $\mathbb{N}$-family of functions mapping nominals and $n - ary$-modality symbols, respectively.

Functor of the $\mathcal{HI}$-sentences. The second step is to enrich the base sentences accordingly. The sentences of the base institution and the nominals are taken as atoms and composed with the boolean connectives, modalities, and satisfaction operators as follows: $\mathrm{Sen}^{\mathcal{HI}}(\Sigma, \mathrm{Nom}, \Lambda)$ is the least set such that

- $\mathrm{Sen}^{I}(\Sigma) \subseteq \mathrm{Sen}^{\mathcal{HI}}(\Delta)$,
- $\mathrm{Nom} \subseteq \mathrm{Sen}^{\mathcal{HI}}(\Delta)$,
- $\rho \star \rho' \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$ for any $\rho, \rho' \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$ and any $\star \in \{\vee, \wedge, \Rightarrow\}$,
- $\neg\rho \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$, for any $\rho \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$,
- $@_i\rho \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$ for any $\rho \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$ and $i \in \mathrm{Nom}$,
- $[\lambda](\rho_1, \ldots, \rho_n)$, for any $\lambda \in \Lambda_{n+1}$, $\rho_i \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$, $i \in \{1, \ldots, n\}$,
- $\langle\lambda\rangle(\rho_1, \ldots, \rho_n)$, for any $\lambda \in \Lambda_{n+1}$, $\rho_i \in \mathrm{Sen}^{\mathcal{HI}}(\Delta)$, $i \in \{1, \ldots, n\}$.

Given a $\mathcal{HI}$-signature morphism $\varphi = (\varphi_{\mathrm{Sign}}, \varphi_{\mathrm{Nom}}, \varphi_{\mathrm{MS}}) : (\Sigma, \mathrm{Nom}, \Lambda) \to (\Sigma', \mathrm{Nom}', \Lambda')$, the translation of sentences $\mathrm{Sen}^{\mathcal{HI}}(\varphi)$ is defined as follows:

- $\mathrm{Sen}^{\mathcal{HI}}(\varphi)(\rho) = \mathrm{Sen}^{I}(\varphi_{\mathrm{Sign}})(\rho)$ for any $\rho \in \mathrm{Sen}^{I}(\Sigma)$,
- $\mathrm{Sen}^{\mathcal{HI}}(\varphi)(i) = \varphi_{\mathrm{Nom}}(i)$,
- $\mathrm{Sen}^{\mathcal{HI}}(\varphi)(\rho \star \rho') = \mathrm{Sen}^{\mathcal{HI}}(\varphi)(\rho) \star \mathrm{Sen}^{\mathcal{HI}}(\varphi)(\rho')$, $\star \in \{\vee, \wedge, \Rightarrow\}$,
- $\mathrm{Sen}^{\mathcal{HI}}(\varphi)(\neg\rho) = \neg\mathrm{Sen}^{\mathcal{HI}}(\varphi)(\rho)$,
- $\mathrm{Sen}^{\mathcal{HI}}(\varphi)(@_i\rho) = @_{\varphi_{\mathrm{Nom}}(i)}\mathrm{Sen}^{\mathcal{HI}}(\rho)$,
- $\mathrm{Sen}^{\mathcal{HI}}(\varphi)([\lambda](\rho_1, \ldots, \rho_n)) = [\varphi_{\mathrm{MS}}(\lambda)](\mathrm{Sen}^{\mathcal{HI}}(\rho_1), \ldots, \mathrm{Sen}^{\mathcal{HI}}(\rho_n))$,
- $\mathrm{Sen}^{\mathcal{HI}}(\varphi)(\langle\lambda\rangle(\rho_1, \ldots, \rho_n)) = \langle\varphi_{\mathrm{MS}}(\lambda)\rangle(\mathrm{Sen}^{\mathcal{HI}}(\rho_1), \ldots, \mathrm{Sen}^{\mathcal{HI}}(\rho_n))$.

$\mathcal{HI}$-models functor Models of the hybridised logic $\mathcal{H}I$ can be regarded as $(\Lambda$-)relational structures whose worlds are I-models. Formally $(\Sigma, \mathrm{Nom}, \Lambda)$-*models* are pairs (M, W) where

- W is a (Nom, Λ)-model in *REL*, called a hybrid structure,
- M is a function $| W | \to | \mathrm{Mod}^{I}(\Sigma) |$.

In each model (M, W), $\{W_n \mid n \in \mathrm{Nom}\}$ provides interpretations for *nominals* in Nom, whereas relations $\{W_\lambda \mid \lambda \in \Lambda_n, n \in \mathbb{N}\}$ interpret *modalities* Λ. We denote the I-model $M(w)$ simply by M_w. The reduct definition is lifted from the base institution I: the reduct of a Δ'-model (M', W') along a signature morphism $\varphi = (\varphi_{\mathrm{Sign}}, \varphi_{\mathrm{Nom}}, \varphi_{\mathrm{MS}}) : \Delta \to \Delta'$, denoted by $\mathrm{Mod}^{\mathcal{HI}}(\varphi)(M', W')$, is the Δ-model (M, W) such that

- W is the $(\varphi_{\mathrm{Nom}}, \varphi_{\mathrm{MS}})$-reduct of W', i.e.
  - $|W| = |W'|$,
  - for any $n \in \mathrm{Nom}$, $W_n = W'_{\varphi_{\mathrm{Nom}}(n)}$,
  - for any $\lambda \in \Lambda$, $W_\lambda = W'_{\varphi_{\mathrm{MS}}(\lambda)}$,
- for any $w \in | W |$, $M_w = \mathrm{Mod}^{I}(\varphi_{\mathrm{Sign}})(M'_w)$.

The Satisfaction Relation. Let $(\Sigma, \mathrm{Nom}, \Lambda) \in | \mathrm{Sign}^{\mathcal{HI}} |$ and $(M, W) \in | \mathrm{Mod}^{\mathcal{HI}}(\Sigma, \mathrm{Nom}, \Lambda) |$. For any $w \in |W|$ we define:

- $(M, W) \models^{w} \rho$ iff $M_w \models^{I} \rho$, when $\rho \in \mathrm{Sen}^{I}(\Sigma)$,
- $(M, W) \models^{w} i$ iff $W_i = w$; when $i \in \mathrm{Nom}$,
- $(M, W) \models^{w} \rho \vee \rho'$ iff $(M, W) \models^{w} \rho$ or $(M, W) \models^{w} \rho'$,
- $(M, W) \models^{w} \rho \wedge \rho'$ iff $(M, W) \models^{w} \rho$ and $(M, W) \models^{w} \rho'$,
- $(M, W) \models^{w} \rho \Rightarrow \rho'$ iff $(M, W) \models^{w} \rho$ implies that $(M, W) \models^{w} \rho'$,
- $(M, W) \models^{w} \neg\rho$ iff $(M, W) \not\models^{w} \rho$,
- $(M, W) \models^{w} @_j\rho$ iff $(M, W) \models^{W_j} \rho$,
- $(M, W) \models^{w} [\lambda](\xi_1, \ldots, \xi_n)$ iff for any $(w, w_1, \ldots, w_n) \in W_\lambda$ we have that $(M, W) \models^{w_i} \xi_i$ for some $1 \leq i \leq n$,
- $(M, W) \models^{w} \langle\lambda\rangle(\xi_1, \ldots, \xi_n)$ iff there exists $(w, w_1, \ldots, w_n) \in W_\lambda$ such that and $(M, W) \models^{w_i} \xi_i$ for any $1 \leq i \leq n$.

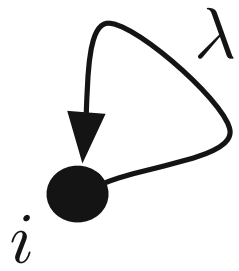

Fig. 1. $\mathcal{H}TRIV$-model

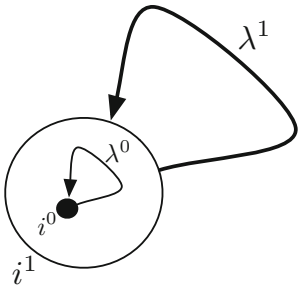

Fig. 2. $\mathcal{H}^2TRIV$-model

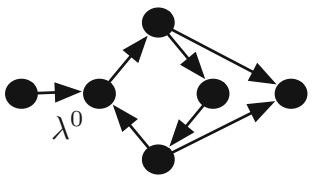

Fig. 3. $\mathcal{H}TRIV$-model

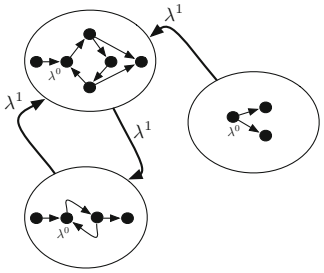

Fig. 4. $\mathcal{H}^2TRIV$-model

We write $(M, W) \models \rho$ iff $(M, W) \models^w \rho$ for any $w \in |W|$.
As expected, $\mathcal{H}I$ is itself an institution satisfying the satisfaction condition:

Theorem 2.1 [MMDB11] *Let* $\Delta = (\Sigma, \text{Nom}, \Lambda)$ *and* $\Delta' = (\Sigma', \text{Nom}', \Lambda')$ *be two* $\mathcal{H}\mathcal{I}$*-signatures and* $\varphi : \Delta \to \Delta'$ *a morphism of signatures. For any* $\rho \in \text{Sen}^{\mathcal{H}I}(\Delta)$, $(M', W') \in |\text{Mod}^C(\Delta')|$, *and* $w \in |W|$,

$$\text{Mod}^{\mathcal{H}I}(\varphi)(M', W') \models^w \rho \text{ iff } (M', W') \models^w \text{Sen}^{\mathcal{H}I}(\varphi)(\rho).$$

Let us illustrate the method by applying it to the trivial institution (twice) as well as to the three other institutions described above.

Example 2.5 ($\mathcal{H}TRIV$ and $\mathcal{H}^2TRIV$) Let us consider the hybridisation of the institution *TRIV* of Example 2.1. The signature category corresponds to

$$\text{Sign}^{TRIV} \times \text{Sign}^{REL} \cong \text{Sign}^{REL}.$$

Since $\text{Sen}^{TRIV}(*) = \emptyset$, $\text{Sen}^{\mathcal{H}TRIV}(*, \text{Nom}, \Lambda)$ is the set of sentences built up from nominals in Nom by the application of modalities in Λ and boolean connectives. This kind of formulas are called *pure hybrid formulas* in [BdRV01, Ind07]. Models of $\text{Mod}^{\mathcal{H}TRIV}(*, \text{Nom}, \Lambda)$ are relational structures (W, M), where M is the constant function $M_w = *$, for any $w \in |W|$ (see Figs. 1, 2).

An interesting institution for the specification of hierarchical state transition systems is obtained through the hybridisation of $\mathcal{H}TRIV$ i.e., the double hybridisation of *TRIV*, which we denote by $\mathcal{H}^2TRIV$. Models of this institution are hybrid structures of hybrid structures (see Fig. 2). Thus $\mathcal{H}^2TRIV$ signatures are triples $((*, \text{Nom}^0, \Lambda^0), \text{Nom}^1, \Lambda^1)$ with Nom^0, Λ^0 and Nom^1, Λ^1 denoting the nominals and the modalities of the first and second layer of hybridisation, respectively. In order to prevent ambiguities, we tag the symbols of each hybrid signature, as well as the connectives and satisfaction operator introduced in each hybridisation, with 0 for the first layer, and with 1 for the second one. For instance, the expression $@_{j^1} k^0 \wedge^1 [\lambda^1](\rho_1, \ldots, \rho_n)$ is a sentence of $\mathcal{H}^2TRIV$ where the symbols k and j represent nominals of the first and second level of hybridisation, respectively. Our tagging convention is extended also to $\mathcal{H}^2TRIV$ models: a $(P, \text{Nom}^0, \text{Nom}^1)$-model is denoted by (M^1, W^1) where, for any $w \in |W^1|$, the models M^1_w are denoted by (W^0_w, M^0_w) (Figs. 3, 4).

Example 2.6 ($\mathcal{H}PL$) The hybridisation of the propositional logic institution PL is an institution where signatures are triples $(Prop, \text{Nom}, \Lambda)$ and sentences are generated by

$$\rho ::= \rho_0 \mid i \mid @_i\rho \mid \rho \odot \rho \mid \neg\rho \mid \langle\lambda\rangle(\rho, \ldots, \rho) \mid [\lambda](\rho, \ldots, \rho) \tag{2}$$

where $\rho_0 \in \text{Sen}^{PL}(Prop)$, $i \in \text{Nom}$, $\lambda \in \Lambda_n$ and $\odot = \{\vee, \wedge, \Rightarrow\}$. Note that there is a double level of connectives in the sentences: one coming from base PL-sentences and another introduced by the hybridisation process. However, they "semantically collapse" in the sense that the semantic interpretation of boolean connectives in both levels is the same, and, hence, no distinction between them needs to be done. (see [DM14] for details). A $(Prop, \text{Nom}, \Lambda)$-model is a pair (M, W), where W is a transition structure with a set of worlds $|W|$. Constants W_i, $i \in \text{Nom}$, stand for the named worlds and $(n + 1)$-ary relations W_λ, $\lambda \in \Lambda_n$, are the accessibility relations characterising the structure. For each world $w \in| W |$, $M(w)$ is a (local) PL-model assigning propositions in $Prop$ to the world w.

Restricting the signatures to those with just a single unary modality (i.e., where $\Lambda_2 = \{\lambda\}$ and $\Lambda_n = \emptyset$ for $n \neq 2$), results in the usual institution for classical hybrid propositional logic [Bra10].

Example 2.7 ($\mathcal{H}MVL_L$) The institution obtained through the hybridisation of MVL_L, for a fixed L, is similar to $\mathcal{H}PL$ defined above, but for two aspects,

- sentences are defined as in (2) but considering MVL $Prop$-sentences (ρ_0, p) as atomic;
- a function, associated to each world $w \in| W |$, assigning to each proposition its value in L.

It is interesting to note that expressivity increases even if one restricts to the case of a (one-world) standard semantics. For instance, differently from the base case where each sentence is tagged by a L-value, one may now deal with more structured expressions involving several L-values, as in, for example, $(\rho, p) \wedge (\rho', p')$.

Example 2.8 ($\mathcal{H}EQ$) Signatures of $\mathcal{H}EQ$ are triples $((S, F), \text{Nom}, \Lambda)$ and sentences are defined as in (2) but taking (S, F)-equations $(\forall X)t = t'$ as atomic base sentences. Models are hybrid structures with a (local)-(S, F)-algebra per world. This institution is a suitable framework to specify reconfigurable systems in a "configurations-as-worlds" perspective: distinct configurations are modelled by distinct algebras; and reconfigurations are expressed by transitions (cf. [MFMB11, Mad13]). Clearly, in this sort of specifications configurations can be specified equationally, based on EQ-signatures, with an initial algebra interpretation. Nominals identify the "relevant" configurations and reconfigurations amount to state transitions. Therefore, one resorts to local equations (i.e. equations tagged by satisfaction operators $@_i(\forall X)t = t'$) to specify local properties of named configurations; to global equations, (i.e. non tagged equations) to specify global properties, i.e. properties true in any state; and, finally, to modal features to specify the reconfigurability dynamics.

3. Bisimulation for hybridised logics

Having briefly reviewed what an institution is and how, through a systematic process, one may build upon an arbitrary logic both modalities and nominals to explicitly refer to states in a specification, we may now focus on the paper's specific contribution. Our starting point is a method to specify reconfigurable software as transition systems whose states represent particular configurations. Each state can endow an algebra, a relation structure or even another, local transition system. Such two-staged specifications are common in the Software Engineering practice (see, e.g., Gurevich's Abstract State Machines [BS03]).

The originality of our method lies in its genericity: whatever logic is found useful to specify each concrete configuration, a method is offered to compute its hybrid counterpart. In this setting, within the next three sections, we look for suitable notions of equivalence and refinement for this kind of specifications. Naturally, such notions should also be parametric on the base logic used, i.e. on the language in which the specification of each concrete configuration is written. The price to pay is, of course, some extra notation and the use of a generic framework—that of *institutions*—in which concepts can be formulated and results proved once and for all.

As the external layer of a reconfigurable system specification is that of a transition system, it is natural to resort to suitable formulations of *bisimilarity* and *similarity* to capture equivalence and refinement, respectively. The precise characterisation of such notions at the high level of abstraction chosen, is, in fact, the paper's main contribution.

Intuitively a bisimulation relates worlds which exhibit the "same" (observable) information and preserve this property along transitions. Thus, to define a general notion of bisimulation over Kripke structures whose states are models of whatever base logic was chosen for expressing specifications, we have to make precise what the "same" information actually means. For example, if the system's configurations are specified by *equations*, establish that two such configurations are bisimilar will certainly require that each specification generates the same variety. Actually, in this case they are essentially the same data type. In the more general setting of this paper the base logic I is a parameter and we have to deal with its hybridised version $\mathcal{H}I$.

Our proposal is, thus, to resort to the broader notion of *elementary equivalence* (see e.g. [Hod97]), and add to the definition of bisimulation the requirement that local configurations, i.e. local I-models related by a bisimulation have to be *elementarily equivalent*. Two models $M, M' \in \mathrm{Mod}(\Sigma)$ are elementarily equivalent if they satisfy the same sentences, as formalised in Definition 3.1 below.

In certain cases, as detailed below, it is convenient to restrict this equivalence by considering only a specific subset of sentences. For instance, we may want to identify FOL-models with elementarily equivalent algebraic reducts. As an illustration consider two models N_{odd} and N_{even} over the natural numbers, both with the operation $+$, one with a predicate $even$ and the other with a predicate odd. Clearly they are not elementarily equivalent if we consider the entire set of sentences. However, $N_{odd} \equiv^{\mathrm{S}} N_{even}$, for a subfunctor S of the sentences functor defined without making use of predicates. Another example, in hybrid Kripke semantics, is to consider models elementarily equivalent only at the frames level, which can be achieved by restricting the sentences to the so-called pure formulas (i.e. sentences without propositional variables). This can be done by parameterising the definition of elementary equivalence (and, consequently, those of bisimulation and refinement) with a subfunctor S of the sentences' functor in order to capture the 'right' set of sentences, as proposed in [MMB13]. Doing this, however, is equivalent to restrict the base institution I to an institution defined as I but replacing Sen^I by S. In the sequel we stick to this alternative to simplify notation.

Definition 3.1 Let $M, M' \in \mathrm{Mod}^I(\Sigma)$ be two models. M and M' are elementarily equivalent, in symbols $M \equiv M'$, if for any $\rho \in \mathrm{Sen}^I(\Sigma)$

$$M \models^I \rho \text{ iff } M' \models^I \rho. \tag{3}$$

Under the institution theory *motto—truth is invariant under change of notation*—we write $M \equiv_\varphi M'$ whenever $M \equiv \mathrm{Mod}^I(\varphi)(M')$ for a given $\varphi \in \mathrm{Sign}^I(\Sigma, \Sigma')$, $M \in \mathrm{Mod}^I(\Sigma)$ and $M' \in \mathrm{Mod}^I(\Sigma')$. Then M and M' are said to be φ-elementarily equivalent. If only the left to right implication of (3) holds, we write $M \gg_\varphi M'$.

Resorting to the satisfaction condition in I, the following characterisation of φ-elementary equivalence pops out:

Corollary 3.1 *$M \equiv_\varphi M'$ iff, for any $\rho \in \mathrm{Sen}^I(\Sigma)$, $M \models^I_\Sigma \rho \Leftrightarrow M' \models^I_{\Sigma'} \mathrm{Sen}^I(\varphi)(\rho)$.*

Note the role of φ above: as a signature morphism it captures the possible *change of notation* from a specification to another. For example it may cater for a renaming of propositions, as in Example 3.1, or signature components, as in Example 3.2. However, its pertinence becomes clearer in refinement situations, as discussed in the next section. At that level it may accommodate a number of forms of interface enrichment or adaptation (e.g. through the introduction of auxilliar operations).

Let us now define bisimulation in this general setting.

Definition 3.2 Let $\mathcal{H}I$ be the hybridisation of the institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. A *φ-bisimulation between models* $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ *and* $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$ is a non-empty relation $\mathrm{B}_\varphi \subseteq |W| \times |W'|$ such that

(i) for any $w \mathrm{B}_\varphi w'$, $M_w \equiv_{\varphi_{\mathrm{Sign}}} M'_{w'}$,

(ii) for any $w \mathrm{B}_\varphi w'$, and for any $i \in \mathrm{Nom}$, $W_i = w$ iff $W'_{\varphi_{\mathrm{Nom}}(i)} = w'$,

(iii) for any $i \in \mathrm{Nom}$, $W_i \mathrm{B}_\varphi W'_{\varphi_{\mathrm{Nom}}(i)}$,

(iv) For any $\lambda \in \Lambda_n$, if $(w, w_1, \ldots, w_n) \in W_\lambda$ and $w \mathrm{B}_\varphi w'$, then for each $k \in \{1, \ldots, n\}$ there is a $w'_k \in |W'|$ such that $w_k \mathrm{B}_\varphi w'_k$ and $(w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$ (*zig*-condition),

(v) For any $\lambda \in \Lambda_n$ if $(w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$ and $w \mathrm{B}_\varphi w'$, then for each $k \in \{1, \ldots, n\}$ there is a $w_k \in |W|$, such that $w_k \mathrm{B}_\varphi w'_k$ and $(w, w_1, \ldots, w_n) \in W_\lambda$ (*zag*-condition).

Note that clause (i) enforces local models of bisimilar states to be elementary equivalent. Clauses (ii) and (iii) deal with nominals: named bisimilar states are identified by the same nominal (ii) and all of them are in the bisimulation (iii). Finally, clauses (iv) and (v) correspond to the usual zig-zag conditions. As usual, a *bisimilarity* relation can be defined as the greatest bisimulation whose existence is guaranteed by Lemma 3.1 below. Therefore, we say that (M, W) and (M', W') are φ-*bisimilar*, and write $(M, W) \rightleftharpoons_\varphi (M', W')$, if there is a φ-bisimulation B_φ between them. Whenever φ is the identity we simply talk of a bisimulation B and the bisimilarity relation $\rightleftharpoons$.

Lemma 3.1 *Let $\mathcal{H}I$ be the hybridisation of the institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. The set of φ-bisimulations between models $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ and $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$ is closed under union.*

Proof. Let $B^0_\varphi, B^1_\varphi \subseteq\mid W \mid \times \mid W' \mid$ be two φ-bisimulations between models $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ and $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$. Their union $B_\varphi = B^0_\varphi \cup B^1_\varphi$ is also a φ-bisimulation because

1. Clearly, all points named by nominals are related by B_φ as they are either by B^0_φ or B^1_φ. Moreover, for any pair (w, w') such that $wB_\varphi w'$ either $wB^0_\varphi w'$ or $wB^1_\varphi w'$. As both B^0_φ and B^1_φ are φ-bisimulations, clauses (i), (ii) and (iii) in Definition 3.2 hold for B_φ.
2. A similar argument applies to both (*zig*) and (*zag*) conditions. For clause (iv) let $(w, w_1, \ldots, w_n) \in W_\lambda$ and $wB_\varphi w'$. Then, either $wB^0_\varphi w'$ or $wB^1_\varphi w'$. Then, for each $k \in \{1, \ldots, n\}$ there is a $w'_k \in\mid W' \mid$ such that $w_k B^0_\varphi w'_k$ or $w_k B^1_\varphi w'_k$, i.e., $w_k B_\varphi w'_k$, and $(w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$. The (*zag*) condition is proved similarly. □

Consider, now, the relational composition of bisimulations.

Lemma 3.2 *Let $\mathcal{H}I$ be the hybridisation of the institution I, $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta'')$ and $\psi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta'', \Delta')$ two signature morphisms. Consider a φ-bisimulation B_φ between models $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ and $(M'', W'') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta'')$ and a ψ-bisimulation B_ψ between models $(M'', W'') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta'')$ and $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$. Then $B_\psi . B_\varphi$ is a $(\psi.\varphi)$-bisimulation between models (M, W) and (M', W').*

Proof. Let $wB_\psi . B_\varphi w'$. Therefore, there is a $w'' \in\mid W'' \mid$ such that $wB_\varphi w''$ and $w''B_\psi w'$. Then, for any $i \in \mathrm{Nom}$, $W_i = w$ iff $W''_{\varphi_{\mathrm{Nom}}(i)} = w''$ iff $W'_{\psi_{\mathrm{Nom}}(i)} = w'$, which proves clause (ii) in Definition 3.2. Clauses (i) and (iii) follow from similar arguments, considering, for the former, that elementary equivalence is an equivalence relation. To establish (iv) suppose that $(w, w_1, \ldots, w_n) \in W_\lambda$. As B_φ is a φ-bisimulation, for each $k \in \{1, \ldots, n\}$ there is w''_k such that $w_k B_\varphi w''_k$ and $(w'', w''_1, \ldots, w''_n) \in W''_\lambda$. As B_ψ is a ψ-bisimulation, there is also a w'_k such that $w''_k B_\psi w'_k$ and $(w', w'_1, \ldots, w'_n) \in W'_\lambda$, which establishes the (*zig*)-condition for relation $B_\psi . B_\varphi$. The (*zag*)-condition, (v), is shown similarly. □

Clearly,

Corollary 3.2 *$\rightleftharpoons$ is an equivalence relation.*

Proof. If no change of signature is involved, this follows from Lemma 3.2 for φ, ψ the identity, together with the observation that the identity relation and the converse of a id-bisimulation are themselves id-bisimulations (for the latter resort to the (*zig*) and (*zag*) conditions interchangeably). □

Theorem 3.1 *Let $\mathcal{H}I$ be the hybridisation of the institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. Let $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$. Then,*

$$\mathrm{Mod}^{\mathcal{H}I}(\varphi)(M', W') \rightleftharpoons_\varphi (M', W')$$

witnessed by the identity relation.

Proof. All the conditions in Definition 3.2 follow from the definition of reduct of $\mathcal{H}I$. □

Example 3.1 (Bisimulation in $\mathcal{H}PL$) Let us instantiate Definition 3.2 for the $\mathcal{H}PL$ case (cf. Example 2.2). More precisely, a sub-institution of $\mathcal{H}PL$ with $\Lambda_2 = \{\lambda\}$ and $\Lambda_n = \emptyset$ for $n \neq 2$. A bisimulation B is such that $(M, W)\mathrm{B}(M', W')$, for any two models $(M, W), (M', W') \in\mid \mathrm{Mod}^{\mathcal{H}PL}(P, \mathrm{Nom}, \{\lambda\}) \mid$, if

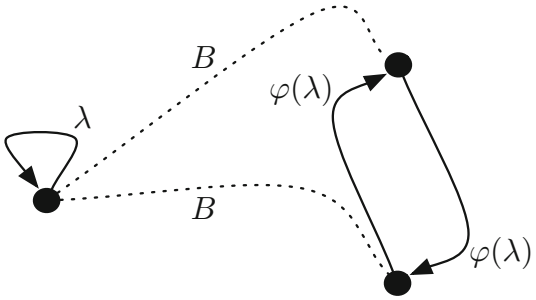

Fig. 5. $\mathcal{H}TRIV$-Bisimulation

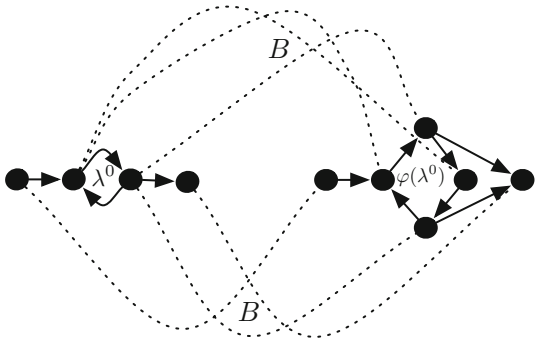

Fig. 6. $\mathcal{H}TRIV$-Bisimulation

(i) $M_w \equiv M'_{w'}$, i.e., bisimilar states satisfy the same sentences,
(ii) for any $i \in \mathrm{Nom}$, $w \mathrm{B} w'$, $w = W_i$ iff $w' = W'_i$,
(iii) for any $i \in \mathrm{Nom}$, $W_i \mathrm{B}\, W'_i$,
(iv) for any $(w, w_1) \in W_\lambda$ with $w \mathrm{B} w'$, there is a $w'_1 \in |W'|$ such that $w_1 \mathrm{B} w'_1$ and $(w', w'_1) \in W'_\lambda$,
(v) for any $(w', w'_1) \in W'_\lambda$ with $w \mathrm{B} w'$, there is a $w_1 \in |W|$ such that $w_1 \mathrm{B} w'_1$ and $(w, w_1) \in W_\lambda$.

Note that condition (i) is equivalent to say that bisimilar states are assigned the same set of propositions (for any $p \in P$, $M_w(p) = \top$ iff $M'_{w'}(p) = \top$). As expected, this definition corresponds exactly to standard bisimulation for propositional hybrid logic (see, e.g. [tC05, Defn. 4.1.1]).

The definition of bisimulation computed in the previous example can also capture the case of propositional modal logic: just consider pure modal signatures (i.e. with an empty set of nominals), as condition (i) is trivially satisfied. Moreover, instantiating Theorem 4.1 below, we get the classical result about preservation of modal truth by bisimulation.

Example 3.2 (Bisimulation for $\mathcal{H}EQ$) Consider now the instantiation of 3.2 for $\mathcal{H}EQ$ (cf. Ex 2.8). All one has to do is to replace condition (ii) in Defn 3.2 by its instantiation for algebras: two algebras are elementarily equivalent if the respective generated varieties coincide [Grä79].

Example 3.3 (Bisimulation in $\mathcal{H}TRIV$ and $\mathcal{H}^2TRIV$) Let us play the same game for $\mathcal{H}TRIV$. Since there are no sentences in $\mathrm{Sen}^{TRIV}(*)$, property (i) trivially holds. Hence bisimulations for $\mathcal{H}TRIV$ consist of standard bisimulations in labeled transition systems with the additional assumptions on named states [clauses (ii) and (iii) in Definition 3.2]. Two examples are depicted in Figs. 5 and 6.

Finally, consider bisimulations in $\mathcal{H}^2TRIV$. At the local level, according to the forthcoming Theorem 4.2 it is enough to have a total and surjective bisimulation to guarantee elementary equivalence in condition (i). Therefore, bisimulation in $\mathcal{H}^2TRIV$ follows from hierachical bisimulation between structured transition systems. An example is depicted in Fig. 7 where B^0 and B^1 are the bisimulations at the local and global levels, respectively. Another example is illustrated in Fig. 8.

4. A Hennessy–Milner theorem

This section discusses the relationship between bisimulation and logical equivalence in the context of hybridised logics. The following result establishes that (local)-hybrid satisfaction is invariant under φ-bisimulations:

Theorem 4.1 *Let $\mathcal{H}I$ be the hybridisation of the institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. Let $\mathrm{B}_\varphi \subseteq |W| \times |W'|$ be a φ-bisimulation. Then, for any $w \mathrm{B}_\varphi w'$ and for any $\rho \in \mathrm{Sen}^{\mathcal{H}I}(\Delta)$,*

$$(M, W) \models^w \rho \text{ iff } (M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho). \tag{4}$$

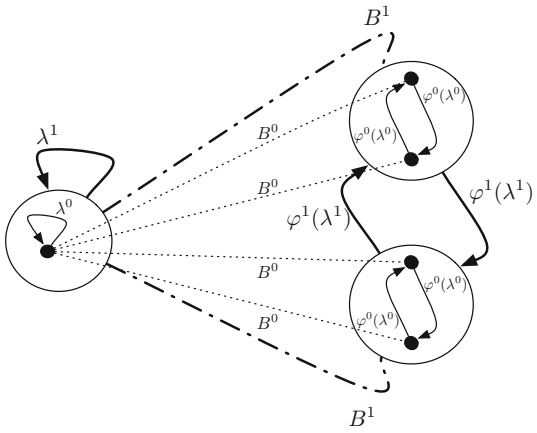

Fig. 7. $\mathcal{H}^2TRIV$-Bisimulation

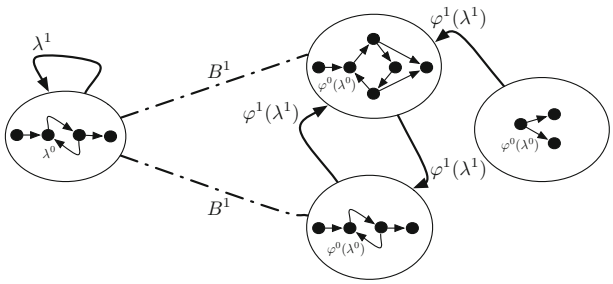

Fig. 8. $\mathcal{H}^2TRIV$-Bisimulation

Proof. The proof is by induction on the structure of the sentences.

1. $\rho = i$ for some $i \in \mathrm{Nom}$:

$$
\begin{aligned}
& (M, W) \models^w i \\
\Leftrightarrow \quad & \{ \text{ definition of } \models^w \} \\
& W_i = w \\
\Leftrightarrow \quad & \{ \text{ clause (ii) of Definition 3.2 } \} \\
& W'_{\varphi(i)} = w' \\
\Leftrightarrow \quad & \{ \text{ definition of } \models^{w'} \} \\
& (M', W') \models^{w'} \varphi_{\mathrm{Nom}}(i) \\
\Leftrightarrow \quad & \{ \text{ definition of } \mathrm{Sen}^{\mathcal{H}I}(\varphi) \} \\
& (M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(i)
\end{aligned}
$$

2. $\rho \in \mathrm{Sen}^I(\Sigma)$:

$$
\begin{aligned}
& (M, W) \models^w \rho \\
\Leftrightarrow \quad & \{ \text{ definition of } \models^w \} \\
& M_w \models^I \rho \\
\Leftrightarrow \quad & \{ \text{ by hypothesis } M_w \equiv_{\varphi_{\mathrm{Sign}}} M'_{w'} \text{ and Corollary 3.1} \} \\
& M'_{w'} \models \mathrm{Sen}^I(\varphi_{\mathrm{Sign}})(\rho) \\
\Leftrightarrow \quad & \{ \text{ definition of } \models^{w'} \} \\
& (M', W') \models^{w'} \mathrm{Sen}^I(\varphi_{\mathrm{Sign}})(\rho) \\
\Leftrightarrow \quad & \{ \text{ definition of } \mathrm{Sen}^{\mathcal{H}I}(\varphi) \} \\
& (M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)
\end{aligned}
$$

3. $\rho = \xi \vee \xi'$ for some $\xi, \xi' \in \mathrm{Sen}^{\mathcal{H}I}(\Delta)$:

$$
\begin{aligned}
& (M, W) \models^w \xi \vee \xi' \\
\Leftrightarrow \quad & \{ \text{ definition of } \models^w \} \\
& (M, W) \models^w \xi \text{ or } (M, W) \models^w \xi'
\end{aligned}
$$

$\Leftrightarrow$ { induction hypothesis }

$(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi)$ or
$(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi')$

$\Leftrightarrow$ { definitionof $\models^{w}$ }

$(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi \vee \xi')$

The proofs for cases $\rho = \xi \wedge \xi'$, $\rho = \xi \Rightarrow \xi'$, $\rho = \neg\xi$, etc. are analogous.

4. $\rho = [\lambda](\xi_1, \ldots, \xi_n)$ for some $\xi_1, \ldots, \xi_n \in \mathrm{Sen}^{\mathcal{H}I}(\Delta)$, $\lambda \in \Lambda_{n+1}$:

$(M, W) \models^{w} [\lambda](\xi_1, \ldots, \xi_n)$

$\Leftrightarrow$ { definition of $\models^{w}$ }

for any $(w, w_1, \ldots, w_n) \in W_\lambda$ there is some $k \in \{1, \ldots, n\}$
such that $(M, W) \models^{w_k} \xi_k$

$\Leftrightarrow$ { * }

for any $(w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$ there is some
$p \in \{1, \ldots, n\}$ such that $(M', W') \models^{w'_p} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_p)$

$\Leftrightarrow$ { definition of $\models^{w'}$ }

$(M', W') \models^{w'} [\varphi_{\mathrm{MS}}(\lambda)](\mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_1), \ldots, \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_n))$

$\Leftrightarrow$ { definition of $\mathrm{Sen}^{\mathcal{H}I}(\varphi)$ }

$(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)([\lambda](\xi_1, \ldots, \xi_n))$

For the step marked with * we proceed as follows. Assuming $(w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$ with $w \mathrm{B}_\varphi w'$, we have by clause (v) of Definition 3.2 that there are w_k, with $k \in \{1, \ldots, n\}$, such that $(w, w_1, \ldots, w_n) \in W_\lambda$. By hypothesis, $(M, W) \models^{w_p} \xi_p$ for some $p \in \{1, \ldots, n\}$. Moreover, by the induction hypothesis, $(M', W') \models^{w'_p} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_p)$. Clause (iv) of Definition 3.2 entails the converse implication. The proof for sentences with shape $\rho = \langle\lambda\rangle(\xi_1, \ldots, \xi_n)$ is analogous.

5. $\rho = @_i\xi$ for some $\xi \in \mathrm{Sen}^{\mathcal{H}I}(\Delta)$ and $i \in \mathrm{Nom}$:

$(M, W) \models^{w} @_i\xi$

$\Leftrightarrow$ { definition of $\models^{w}$ }

$(M, W) \models^{W_i} \xi$

$\Leftrightarrow$ { induction hypothesis and clause (iii) of Definition 3.2 }

$(M', W') \models^{W'_{\varphi_{\mathrm{Nom}}(i)}} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi)$

$\Leftrightarrow$ { definition of $\models^{w}$ }

$(M', W') \models^{w} @_{\varphi_{\mathrm{Nom}}(i)}\mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi)$

$\Leftrightarrow$ { definition of $\mathrm{Sen}^{\mathcal{H}I}(\varphi)$ }

$(M', W') \models^{w} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(@_i\xi)$

□

As in the standard modal case the converse of Theorem 4.1 does not hold in general, i.e., logical equivalence is not a bisimulation. Such is the case, however, for image-finite Kripke models, as well known from the plain case of modal logic [BVB07]. A model (M, W) is *image-finite* if for each state $w \in W$ and each relation W_λ, $\lambda \in \Lambda$, the set $\{(w_1, \ldots, w') : (w, w_1, \ldots, w') \in W_\lambda\}$ is finite. No condition is imposed on the number of relations present or the cardinality of W.

We are, thus, prepared to state and prove the following Hennessy–Milner like theorem:

Theorem 4.2 *Let $\mathcal{H}I$ be the hybridisation of the institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. Let (M, W) and (M', W') be two* image-finite Δ *and Δ'-models, respectively. Then, for every $w \in W$ and $w' \in W'$, the following conditions are equivalent:*

(i) $(M, W) \models^{w} \rho$ *iff* $(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$*, for any formula ρ,*
(ii) *There is a φ-bisimulation* $\mathrm{B}_{\varphi} \subseteq\mid W \mid \times \mid W' \mid$ *such that* $w \mathrm{B}_{\varphi} w'$.

Proof. We have just to prove that (i) implies (ii). Let us prove that

$$Z := \left\{(w, w') \in W \times W' : \text{ for any } \rho, (M, W) \models^{w} \rho \text{ iff } (M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)\right\}$$

is a bisimulation.

The atomic conditions trivially hold. For the (*zig*) condition, let $\lambda \in \Lambda$ be, without loss of generality, a binary modality symbol. Assume that wZw' and let $u \in W$ such that $wW_{\lambda}u$. To obtain a contradiction, suppose that there is no $u' \in W'$ with $w'W'_{\lambda}u'$ and uZu'. As in the standard case the image-finite condition makes $S' = \{u' : w'W'_{\lambda}u'\}$ finite. Moreover, S' cannot be empty since in such a case $(M, W) \models^{w} [\lambda]\neg(@_i i)$ [equivalently, $(M, W) \models^{w} \neg\langle\lambda\rangle(@_i i)$], which is incompatible with the fact that $(M, W) \models^{w} \langle\lambda\rangle(@_i i)$, which holds because $wW_{\lambda}u$.

By assumption, for every $v \in S'$ there is a formula ψ_v such that $(M, W) \models^{u} \psi_v$ and it is false that $(M', W') \models^{v} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\psi_v)$. Consider now the conjunction

$$\psi = \bigwedge_{v \in S'} \psi_v$$

of all of these formulas. Then, on the one hand, $(M, W) \models^{w} \langle\lambda\rangle\psi$. On the other, however, for all $v \in S'$, it is false that $(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\langle\lambda\rangle\psi)$. This contradicts the fact that wZw'.

The (*zag*) condition is shown in a similar way. □

5. Forward and backward refinement

Consider again a reconfigurable system described by a set of configurations and a transition structure entailing changes from one to another. If equivalence of such systems corresponds to a notion of bisimilarity in which bisimilar configurations are enforced to be elementary equivalent, a *refinement* relation corresponds to *similarity*. This can be defined in two different ways. One of them entails preservation of transitions from the abstract to the concrete model; the other proceeds dually.

5.1. Forward refinement

Forward refinement means that behaviours (on the system's global dynamics) valid in the abstract model are also allowed in the concrete one, which, however, may exhibit further behaviour. On the other hand, at each local configuration, the original properties are preserved along local refinement. We call this *forward* refinement.

Definition 5.1 Let $\mathcal{H}I$ be the hybridisation of an institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. A *forward φ-refinement relation between models* $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ *and* $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$ is a non-empty relation $\mathrm{R}_{\varphi} \subseteq\mid W \mid \times \mid W' \mid$ such that, for any $w\mathrm{R}_{\varphi}w'$,

(i) $M_w \gg_{\varphi} M'_{w'}$,
(ii) for any $i \in \mathrm{Nom}$, if $W_i = w$ then $W'_{\varphi_{\mathrm{Nom}}(i)} = w'$,
(iii) for any $i \in \mathrm{Nom}$, $W_i \,\mathrm{R}_{\varphi}\, W'_{\varphi_{\mathrm{Nom}}(i)}$,
(iv) for any $\lambda \in \Lambda_n$, if $(w, w_1, \ldots, w_n) \in W_{\lambda}$ then for each $k \in \{1, \ldots, n\}$ there is a $w'_k \in\mid W' \mid$ such that $w_k \mathrm{R}_{\varphi} w'_k$ and $(w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$.

We say that (M', W') is a *forward φ-refinement* of (M, W), in symbols $(M, W) \rightharpoonup_{\varphi} (M', W')$, if there is a forward φ-refinement between them. When φ is the identity we denote it simply by $\rightharpoonup$.

The relevant question is whether (forward) refinement preserves (hybrid) satisfaction. Actually, this is not the case. Note that in the proof of Theorem 4.1 preservation of hybrid satisfaction of sentences of the form $[\lambda](\xi_1, \ldots, \xi_n)$ is entailed by conditions (iv) and (v) of Definition 3.2, but the latter is not considered in a (forward) refinement situation. Boxed formulas are, as a matter of fact, not preserved. As a simple counter-example, define a R_{φ}-refinement from a Δ-hybrid model (M, W) with $| W | = \{w\}$ and $W_{\lambda} = \emptyset$, for $\lambda \in \Lambda_n$, to any other Δ'-hybrid model (M', W') such that $\mathrm{Mod}^{\mathcal{H}I}(\varphi_{\mathrm{Sign}})(M')_{w'} = M_w$ for some $w' \in | W' |$. Any sentence $[\lambda](\xi_1, \ldots, \xi_n)$, which trivially holds in the world w of (M, W), may fail to be satisfied in the R_{φ}-related world w' of (M', W'). Negative sentences $\neg\xi$, are also in general not preserved through refinement because, only the (*zig*) condition being enforced, non satisfaction in one direction does not imply non satisfaction in the other.

Definition 5.2 (*Positive existential sentences*) The positive existential sentences of a signature $\Delta \in | \mathrm{Sign}^{\mathcal{H}I} |$ are given by the subfunctor $\mathrm{Sen}^{\mathcal{H}I}_{\diamond} \subseteq \mathrm{Sen}^{\mathcal{H}I}$ defined inductively for each signature Δ as $\mathrm{Sen}^{\mathcal{H}I}(\Delta)$, but excluding both negation and boxed formulas. For each signature morphism $\varphi : \Delta \to \Delta'$, $\mathrm{Sen}^{\mathcal{H}I}_{\diamond}(\varphi)$ is the restriction of $\mathrm{Sen}^{\mathcal{H}I}(\varphi)$ to $\mathrm{Sen}^{\mathcal{H}I}_{\diamond}(\Delta)$.

Theorem 5.1 *Let $\mathcal{H}I$ be the hybridisation of an institution I, $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism, R_{φ} a φ-refinement relation and $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ and $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$ two models such that (M', W') is a forward refinement of (M, W) witnessed by relation R_{φ}. Then, for any $w \mathrm{R}_{\varphi} w'$ and $\rho \in \mathrm{Sen}^{\mathcal{H}I}_{\diamond}(\Delta)$,*

$$(M, W) \models^{w} \rho \text{ implies that } (M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho).$$

Proof. The proof is by induction on the structure of the existential positive sentences and comes directly from the proof of Theorem 4.1, taking the left to right implication. What remains to be proved is the case $\rho = \langle\lambda\rangle(\xi_1, \ldots, \xi_n)$. Thus,

$$\begin{array}{ll}
& (M, W) \models^{w} \langle\lambda\rangle(\xi_1, \ldots, \xi_n) \\
\Leftrightarrow & \quad \{ \text{ definition of } \models^{w} \} \\
& \text{there exists } (w, w_1, \ldots, w_n) \in W_{\lambda} \\
& \text{such that } (M, W) \models^{w_k} \xi_k \text{ for any } k \in \{1, \ldots, n\} \\
\Rightarrow & \quad \{ \text{By (iii) and (iv) (the (\textit{zig}) condition) and the induction hypothesis.} \} \\
& \text{there exists } (w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)} \\
& \text{such that } (M', W') \models^{w'_k} \xi_k \text{ for any } k \in \{1, \ldots, n\} \\
\Leftrightarrow & \quad \{ \text{ definition of } \models^{w'} \} \\
& (M', W') \models^{w'} \langle\varphi_{\mathrm{MS}}(\lambda)\rangle(\mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_1), \ldots, \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_n)) \\
\Leftrightarrow & \quad \{ \text{definition of } \mathrm{Sen}^{\mathcal{H}I}(\varphi) \} \\
& (M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\langle\lambda\rangle(\xi_1, \ldots, \xi_n))
\end{array}$$

□

The following examples illustrate refinement situations in this setting.

Example 5.1 (Refinement in $\mathcal{H}PL$) Forward refinement notion in $\mathcal{H}PL$ consists of the standard notion of simulation in Kripke structures. Theorem 5.1 generalises the well known preservation result of positive sentences by simulation (see [BdRV01] for the modal standard case). In this case $\mathrm{Sen}^{\mathcal{H}PL}_{\diamond}(\Delta)$ consists exactly in the restriction of $\mathrm{Sen}^{\mathcal{H}}PL(\Delta)$ to all the sentences without occurrences of negations and boxes.

Example 5.2 (Refinement in $\mathcal{H}MVL_L$) Figure 9 presents an example of a refinement in multi-valued logic based on the lattice L_4 (on the left of Fig. 9). Let $MVL^*_{L_4}$ be the institution obtained from MVL_{L_4} by restricting the functor of the sentences to the subfunctor S defined by $\mathrm{S}(LProp) = \{(p, l),\ p \in LProp \text{ and } l \in L_4\}$. Consider now the hybridisation $\mathcal{H}MVL^*_{L_4}$ of $MVL^*_{L_4}$.

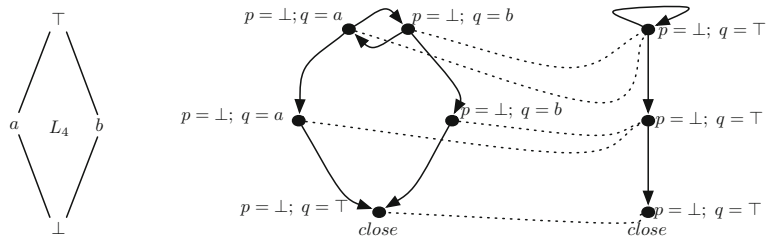

Fig. 9. Forward refinement in $\mathcal{H}MVL_L$

Conditions (ii) and (iii) are obviously satisfied. In what concerns the verification of condition (i) for which $(p, l) \in \mathrm{S}(LProp)$, $M_w \models^{MVL^*_{L_4}}_{LProp} (p, l) \Rightarrow M'_{w'} \models^{MVL^*_{L_4}}_{LProp} (p, l)$, it is sufficient to see that, $(M_w \models p) \leq (M'_{w'} \models p)$, $p \in LProp$.

Example 5.3 (Refinement in $\mathcal{H}EQ$) Consider a store system abstractly modelled as the initial algebra A with signature $((S, F), \Gamma)$ where $S = \{mem, elem\}$, $F_{\to mem} = \{new\}$, $F_{\to elem} = \{0\}$ $F_{mem \times elem \to mem} = \{write\}$, $F_{mem \to mem} = \{del\}$ and $F_{\underline{\mathrm{ar}} \to s} = \emptyset$ otherwise, and where Γ is the following set of equations:

$del(new) = new,$
$del(write(m, e)) = m.$

Suppose one intends to refine this structure by adding a *read* function configurable in two different modes: in one of them it reads the first element in the store, in the other the last. Reconfiguration between the two execution modes is enforced by an external control event *shift*. Note that this abstract model can be seen as the $\big((S, F), \emptyset, \{\mathrm{shift}\}\big)$-hybrid model $\mathcal{M} = (M, W)$, taking $|\ W\ | = \{\star\}$, $W_{\mathrm{shift}} = id$ and $M_\star = A$ (see Fig. 10). Then, we take the inclusion morphism $\varphi_{\mathrm{Sign}} : (S, F) \hookrightarrow (S, F')$ where F' extends F with $F'_{mem \to elem} = \{read\}$. For the envisaged refinement let us consider model $\mathcal{M}' = (M', W')$ where $W' = \{s_1, s_2\}$ and $W'_{\mathrm{shift}} = \{(s_1, s_2), (s_2, s_1)\}$ and where M'_{s_1} and M'_{s_2} are respectively, two algebras satisfying the equations

$read(new) = 0,$
$del(new) = new,$
$del(write(m, e)) = m,$
$read(write(m, e)) = e,$

and

$read(new) = 0,$
$del(new) = new,$
$del(write(m, e)) = m,$
$read(write(write(m, e), e')) = read(write(m, e)),$
$read(write(new, e)) = e$

respectively.

It is not difficult to see that $R = \{(\star, s_1), (\star, s_2)\}$ is a φ-refinement relation: conditions (ii) and (iii) are trivially fulfilled; the initiality of (the algebra) M_* entails the condition (i): as is well known (e.g. [EM85]) properties valid in the initial model of a set of equation are the ones valid in all the models of the respective variety. This includes the models $\mathrm{Mod}(\varphi)(M_{s_1})$ and $\mathrm{Mod}(\varphi)(M_{s_2})$).

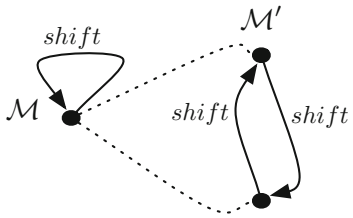

Fig. 10. Forward refinement in $\mathcal{HEQ}$.

5.2. Backward refinement

Forward refinement simulates the abstract model behaviour by the concrete one, i.e. the refined model allows all behaviours specified at the abstract level. A dual notion goes in the opposite direction, enforcing all concrete behaviours to be allowed in the abstract model. Actually this notion is more common in the literature: it constrains the concrete, refined model to exhibit only behaviours allowed in its specification. Formally this leads to a notion of *backward* refinement by replacing condition (iv) in Definition 5.1 by the (*zag*) condition:

(iv) For any $\lambda \in \Lambda_n$, if $(w', w'_1) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$ then for each $k \in \{1, \ldots, n\}$ there is a $w_k \in |\ W\ |$ such that $w_k \mathrm{R}_\varphi w'_k$ and $(w, w_1, \ldots, w_n) \in W_\lambda$.

leading to

Definition 5.3 Let $\mathcal{H}I$ be the hybridisation of an institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. A *backward φ-refinement relation between models* $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ *and* $(M'\, W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$ is a non-empty relation $\mathrm{R}_\varphi \subseteq |\ W\ | \times |\ W'\ |$ such that, for any $w \mathrm{R}_\varphi w'$,

(i) $M_w \gg_\varphi M'_{w'}$,

(ii) for any $i \in \mathrm{Nom}$, if $W_i = w$ then $W'_{\varphi_{\mathrm{Nom}}(i)} = w'$,

(iii) for any $i \in \mathrm{Nom}$, $W_i \,\mathrm{R}_\varphi\, W'_{\varphi_{\mathrm{Nom}}(i)}$,

(iv) For any $\lambda \in \Lambda_n$, if $(w', w'_1) \in W'_{\varphi_{\mathrm{MS}}(\lambda)}$ then for each $k \in \{1, \ldots, n\}$ there is a $w_k \in |\ W\ |$ such that $w_k \mathrm{R}_\varphi w'_k$ and $(w, w_1, \ldots, w_n) \in W_\lambda$.

We say that (M', W') is a *backward φ-refinement* of (M, W), in symbols $(M, W) \leftharpoonup_\varphi (M', W')$, if there is a backward φ-refinement between them. Again $\leftharpoonup_\varphi$ is abbreviated to $\leftharpoonup$ whenever φ is the identity.

Note that existential ('diamond') sentences are no longer preserved through backward refinement: effective transitions at the abstract level can be backward-refined into a non-transition at the concrete level. Universal ('boxed') sentences, however, are preserved, leading to a re-phrasing of Theorem 5.1 for positive, universal sentences, collected in $\mathrm{Sen}^{\mathcal{H}I}_\Box(\Delta)$. Formally,

Definition 5.4 (*Positive universal sentences*) The positive universal sentences of a signature $\Delta \in |\ \mathrm{Sign}^{\mathcal{H}I}\ |$ are given by the subfunctor $\mathrm{Sen}^{\mathcal{H}I}_\Box \subseteq \mathrm{Sen}^{\mathcal{H}I}$ defined inductively for each signature Δ as $\mathrm{Sen}^{\mathcal{H}I}(\Delta)$, but excluding both negation and $\Diamond$-formulas. For each signature morphism $\varphi : \Delta \to \Delta'$, $\mathrm{Sen}^{\mathcal{H}I}_\Box(\varphi)$ is the restriction of $\mathrm{Sen}^{\mathcal{H}I}(\varphi)$ to $\mathrm{Sen}^{\mathcal{H}I}_\Box(\Delta)$.

Theorem 5.2 *Let $\mathcal{H}I$ be the hybridisation of an institution I, $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism, R_φ a backward φ-refinement relation and $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ and $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$ two models such that (M', W') is a backward φ-refinement of (M, W) witnessed by relation R_φ. Then, for any $w \mathrm{R}_\varphi w'$ and $\rho \in \mathrm{Sen}^{\mathcal{H}I}_\Box(\Delta)$,*

$(M, W) \models^w \rho$ *implies that* $(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$.

Proof. The crucial step in the proof is the preservation of 'boxed' formulas $\rho = [\lambda](\xi_1, \ldots, \xi_n)$, as follows:

$$
\begin{array}{ll}
& (M, W) \models^{w} [\lambda](\xi_1, \ldots, \xi_n) \\
\Leftrightarrow & \quad \{ \text{ definition of } \models^{w} \} \\
& \text{for all } (w, w_1, \ldots, w_n) \in W_\lambda,\ (M, W) \models^{w_k} \xi_k, \quad \text{for any } k \in \{1, \ldots, n\} \\
\Rightarrow & \quad \{ (\star) \} \\
& \text{for all } (w', w'_1, \ldots, w'_n) \in W'_{\varphi_{\mathrm{MS}}(\lambda)},\ (M', W') \models^{w'_k} \xi_k, \quad \text{for any } k \in \{1, \ldots, n\} \\
\Leftrightarrow & \quad \{ \text{ definition of } \models^{w'} \} \\
& (M', W') \models^{w'} [\varphi_{\mathrm{MS}}(\lambda)](\mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_1), \ldots, \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\xi_n)) \\
\Leftrightarrow & \quad \{ \text{ definition of } \mathrm{Sen}^{\mathcal{H}I}(\varphi) \} \\
& (M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)([\lambda](\xi_1, \ldots, \xi_n))
\end{array}
$$

The proof step marked with ($\star$) above is justified as follows: the (*zag*) condition guarantees that if there is a set of transitions from w in the abstract model, a subset (possibly empty) of corresponding transitions is also present in the concrete model from state w'. Actually, this is an equivalence step, with the implication from right to left being just a direct consequence of the (*zag*) condition. □

Of course the restriction to *positive* sentences is also enforced here. If such was not the case the whole argument would collapse as existential sentences could be built from universal ones and vice-versa.

Therefore, we end up with two notions of refinement defined in terms of which transitions are globally preserved and in which direction. If one regards 'boxed' properties as a sort of (elementary) *safety* requirements, one could state that backward refinement preserves safety. Dually, regarding existential sentences as (elementary) *liveness* requirements, forward refinement preserves liveness. It comes to no surprise that the more common notion of refinement, that of backward refinement, preserves safety.

6. Refinement of specifications

Until now we have been seeking for suitable notions of equivalence and refinement between models of specifications in hybridised institutions. We shall now turn to the *specifications* themselves, in the sense the word has in the tradition of *property oriented* specification methods (see [ST12] for a recent overview).

A specification is a collection of properties a system is supposed to obey, i.e. a theory in a suitable institution. Its semantics is the class of models satisfying such a theory. Formally, a (non-structured) specification in a institution I consists of a pair (Δ, E), where $\Delta \in \mathrm{Sign}^I$ and $E \subseteq \mathrm{Sen}^I(\Delta)$. Its (loose) semantics is given by

- its signature $Sig[SP] = \Delta$, for some $\Delta \in |\, \mathrm{Sign}^I \,|$,
- its class of models $[|\ SP\ |] = \{M \in |\, \mathrm{Mod}^I(\Delta) \,|\colon M \models^I_\Delta E\}$.

Conceptually, $[|\ SP\ |]$ can be understood as the class of admissible implementations for the system and, the implementation of SP, as one of these models chosen to realise the system. The construction of this particular model proceeds by a stepwise refinement process. Formally, we say that SP' refines SP via φ, in symbols, $SP' \rightsquigarrow_\varphi SP$, if

- $\varphi \in \mathrm{Sign}^I(Sig(SP), Sig(SP'))$,
- $[|\ SP'\ |]\,|_\varphi \subseteq [|\ SP\ |]$, where $[|\ SP'\ |]\,|_\varphi = \{\mathrm{Mod}^I(\varphi)(M) \mid M \in [|\ SP\ |]\}$.

Note that this is a straightforward generalisation of the notion of *simple refinement* in algebraic specification e.g. [San99], in which case $Sig[SP] = Sig[SP']$ and φ is the identity. Similarly, two specifications SP and SP' are equivalent up to a signature morphism $\varphi : Sig[SP] \to Sig[SP']$ when $[|\ SP'\ |]\,|_\varphi = [|\ SP\ |]$.

Back to dealing with classes of models, we are also back to the notions of bisimulation and refinement used before. Although in process algebra, where such notions were born, their formulation is essentially local (e.g., two processes are bisimilar if their *initial* states are related by a bisimulation), when reasoning with specifications

a notion of initial state is usually absent. This entails the need for a shift of perspective for"globalising" the preservation results. In particular, the local characterisation established in Theorem 4.1, can be re-framed as follows:

Theorem 6.1 *Let $\mathcal{H}I$ be the hybridisation of institution I and $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism. Let $\mathrm{B}_\varphi \subseteq\mid W \mid \times \mid W' \mid$ be a total and surjective φ-bisimulation. Then,*

$$(M, W) \models^{\mathcal{H}I} \rho \text{ iff } (M', W') \models^{\mathcal{H}I} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho) \tag{5}$$

Proof. Let us suppose $(M, W) \models^{\mathcal{H}I} \rho$, i.e. that for any $w \in\mid W \mid$, $(M, W) \models^{w} \rho$. Since B_φ is surjective, for any $w' \in\mid W' \mid$ there is a $w \in\mid W \mid$ such that $w \mathrm{B}_\varphi w'$. Since $(M' W) \models^{w} \rho$, by Theorem 4.1, $(M', W') \models^{w'} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$. Hence $(M', W') \models^{\mathcal{H}I} \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$. The converse implication is proved similarly using resorting to the totality of B_φ. □

A similar global characterisation of preservation results for both forward and backward refinements arises as a corollary of Theorem 5.1 and its backward counterpart explained in Sect. 5.2.

Corollary 6.1 *Let $\mathcal{H}I$ be the hybridisation of an institution I, $\varphi \in \mathrm{Sign}^{\mathcal{H}I}(\Delta, \Delta')$ a signature morphism, $(M, W) \in \mathrm{Mod}^{\mathcal{H}I}(\Delta)$ and $(M', W') \in \mathrm{Mod}^{\mathcal{H}I}(\Delta')$ two $\mathcal{H}\mathcal{I}$-models and $\mathrm{R}_\varphi :\mid W \mid \times \mid W' \mid$ a relation.*

1. *if R_φ is a surjective forward φ-refinement relation, we have that for any $\rho \in \mathrm{Sen}^{\mathcal{H}I}_{\diamond}(\Delta)$,*

   $(M, W) \models^{\mathcal{H}I} \rho$ *implies that* $(M', W') \models \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$.

2. *if R_φ is a total backward φ-refinement relation, we have that for any $\rho \in \mathrm{Sen}^{\mathcal{H}I}_{\Box}(\Delta)$,*

   $(M, W) \models^{\mathcal{H}I} \rho$ *implies that* $(M', W') \models \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$.

The following results relate specification refinement ($\rightsquigarrow$) with bisimulation and with refinement of specification models as previously introduced.

Theorem 6.2 *Let $SP = (\Delta, E)$ and $SP' = (\Delta, E')$ be two specifications. Then, the following statements are equivalent:*

1. $SP \rightsquigarrow_\varphi SP'$,
2. *for any $(M', W') \in [\mid SP' \mid]$, there is a $(M, W) \in [\mid SP \mid]$ such that $(M, W) \rightleftharpoons_\varphi (M', W')$ witnessed by a total and surjective bisimulation.*

Proof. **1** ⇒ **2** By assumption, that for any $(M', W') \in [\mid SP' \mid]$, $\mathrm{Mod}^{\mathcal{H}I}(\varphi)(M', W') \in [\mid SP \mid]$. By Theorem 3.1, there is a model $(M, W) \in [\mid SP \mid](= \mathrm{Mod}^{\mathcal{H}I}(\varphi)(M', W'))$ such that $(M, W) \rightleftharpoons_\varphi (M', W')$ witnessed by the identity relation, a total and surjective bisimulation.

2 ⇒ **1** Let us consider a model $(M', W') \in [\mid SP' \mid]$. By hypothesis there is a $(M, W) \in [\mid SP \mid]$ such that $(M, W) \rightleftharpoons_\varphi (M', W')$. Hence by Corollary 6.1, for any $\rho \in \mathrm{Sen}^{\mathcal{H}I}(\Delta)$, $(M, W) \models \rho$ iff $(M', W') \models \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$. In particular, $(M', W') \models \mathrm{Sen}^{\mathcal{H}I}(\varphi)(E)$. By Satisfaction Condition we have $\mathrm{Mod}^{\mathcal{H}I}(\varphi)(W', M') \models E$, i.e., $\mathrm{Mod}^{\mathcal{H}I}(\varphi)(M', W') \in [\mid SP \mid]$. Therefore $SP \rightsquigarrow_\varphi SP'$. □

Theorem 6.3 *Let $SP = (\Delta, E)$ and $SP' = (\Delta, E')$ be two specifications with $E \subseteq \mathrm{Sen}^{\mathcal{H}I}_{\diamond}(\Delta)$. Then, the following statements are equivalent:*

1. $SP \rightsquigarrow_\varphi SP'$,
2. *for any $(M', W') \in [\mid SP' \mid]$, there is a $(M, W) \in [\mid SP \mid]$ such that $(M, W) \rightharpoonup_\varphi (M', W')$ witnessed by a surjective refinement relation.*

Proof. **1.** ⇒ **2.** This implication is proved analogously to the implication 1 ⇒ 2 in Theorem 6.2 using the fact that $(M, W) \rightleftharpoons_\varphi (M', W')$ implies $(M, W) \rightharpoonup_\varphi (M', W')$ and also $(M, W) \leftharpoonup_\varphi (M', W')$.

2. ⇒ **1.** Let us consider a model $(M', W') \in [\mid SP' \mid]$. By hypothesis there is a $(M, W) \in [\mid SP \mid]$ such that $(M, W) \rightharpoonup_\varphi (M', W')$. Hence by item 1. of Corollary 6.1, for any $\rho \in \mathrm{Sen}^{\mathcal{H}I}_{\diamond}(\Delta)$, $(M, W) \models \rho$ implies that $(M', W') \models \mathrm{Sen}^{\mathcal{H}I}(\varphi)(\rho)$. In particular, $(M', W') \models \mathrm{Sen}^{\mathcal{H}I}(\varphi)(E)$. The Satisfaction Condition entails $\mathrm{Mod}^{\mathcal{H}I}(\varphi)(W', M') \models E$, i.e., $\mathrm{Mod}^{\mathcal{H}I}(\varphi)(M', W') \in [\mid SP \mid]$. Therefore $SP \rightsquigarrow_\varphi SP'$. □

Theorem 6.4 *Let* $SP = (\Delta, E)$ *and* $SP' = (\Delta, E')$ *be two specifications with* $E \subseteq \mathrm{Sen}_{\Box}^{\mathcal{H}I}(\Delta)$. *Then, the following statements are equivalent:*

1. $SP \rightsquigarrow_{\varphi} SP'$,
2. *for any* $(M', W') \in [\!|\ SP'\ |\!]$, *there is a* $(M, W) \in [\!|\ SP\ |\!]$ *such that* $(M, W) \leftharpoonup_{\varphi} (M', W')$ *witnessed by a total refinement relation.*

Proof. The proof is analogous to the one of Theorem 6.3 but using, in the implication $2 \Rightarrow 1$, item 2. of Corollary 6.1. □

7. Conclusions

This paper introduced notions of equivalence and refinement for models of hybrid specifications, i.e., specifications formalised in hybridised versions of logics used to describe systems' possible configurations. The definition is parametric on precisely the base logic relevant for each application.

From an engineering point of view, the characterisation of suitable, generic notions of equivalence and refinement is fundamental to a software design methodology to deal with systems' reconfigurability in a rigorous way. Such a methodology was introduced in [MFMB11], and provided with effective, computer-based proof support through the recent implementation [NMMB13] of the hybridisation method in the HETS platform [MML07].

Current work on this topic includes the study of typical constructions on Kripke structures (e.g. bounded morphism images, substructures and disjoint unions) and their characterisation under bisimilarity and refinement. Whether the complexity of each hybridised logic can be computed from the complexity of the corresponding base logic remains a somehow lateral, but challenging research topic.

Acknowledgements

This work is funded by ERDF—European Regional Development Fund, through the COMPETE Programme, and by National Funds through FCT within project FCOMP-01-0124-FEDER-028923 and by project NORTE-07-0124-FEDER-000060, co-financed by the North Portugal Regional Operational Programme (ON.2), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF). The work had also partial financial assistance by the project PEst-OE/MAT/UI4106/2014 at CIDMA,FCOMP-01-0124-FEDER-037281 at INESC TEC and the Marie Curie project FP7-PEOPLE-2012-IRSES (GetFun).

References

[ACEGG90] Agusti-Cullell J, Esteva F, Garcia P, Godo L (1990) Formalizing multiple-valued logics as institutions. In: Bouchon-Meunier B, Yager RR, Zadeh LA (eds) 3rd International conference on information processing and management of uncertainty in knowledge-based systems (IPMU 90, Paris, France, July 2–6, 1990). Lecture notes in computer science, vol 521. Springer, pp 269–278

[AtC06] Areces C, ten Cate B (2006) Hybrid logics. In: Blackburn P, Wolter F, van Benthem J (eds) Handbook of modal logics. Elsevier, Amsterdam, pp 821–868

[BD94] Burstall R, Diaconescu R (1994) Hiding and behaviour: an institutional approach. In: Roscoe W (ed) A classical mind: essays in honour of C.A.R. Hoare. Prentice-Hall, Hertfordshire, pp 75–92

[BdRV01] Blackburn P, de Rijke M, Venema Y (2001) Modal logic. Number 53 in Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, Cambridge

[BH06] Bidoit M, Hennicker R (2006) Constructor-based observational logic. J Logic Algebr Progr 67(1–2):3–51

[BKI05] Beierle C, Kern-Isberner G (2005) Looking at probabilistic conditionals from an institutional point of view. In: Kern-Isberner G, Rödder W, Kulmann F (eds) Conditionals, information, and inference (revised selected papers of WCII 2002, Hagen, Germany, May 13–15, 2002). Lecture notes in computer science, vol 3301. Springer, pp 162–179

[Bra10] Brauner T (2010) Hybrid logic and its proof-theory. Applied logic series, Springer, Netherlands

[BS03] Börger E, Stärk R (2003) Abstract state machines: a method for high-level system design and analysis. Springe, Berlin

[BVB07] Blackburn P, Van Benthem J (2007) Modal logic: a semantic perspective. In: Blackburn P, Wolter F, van Benthem J (eds) Handbook of modal logic, studies in logic and practical reasoning, vol 3. Elsevier, Amsterdam, pp 1–82

[Cô6] Cîrstea C (2006) An institution of modal logics for coalgebras. J Logic Algebr Progr 67(1–2):87–113
[CMSS06] Caleiro C, Mateus P, Sernadas A, Sernadas C (2006) Quantum institutions. In: Futatsugi K, Jouannaud J-P, Meseguer J (eds) Algebra, meaning, and computation, essays dedicated to Joseph A. Goguen on the occasion of his 65th birthday. Lecture notes in computer science, vol 4060. Springer, pp 50–64
[Dia08] Diaconescu R (2008) Institution-independent model theory. studies in universal logic. Birkhäuser, Basel
[Dia11] Diaconescu R (2011) On quasi-varieties of multiple valued logic models. Math Log Q 57(2):194–203
[DM14] Diaconescu R, Madeira A (2014) Encoding hybridized institutions into first order logic. Math Struct Comput Sci. doi:10.1017/S0960129514000383
[EM85] Ehrig H, Mahr B (1985) Fundamentals of algebraic specification 1: equations and initial semantics. Monographs in theoretical computer science, an EATCS Series. Springer, Berlin
[GB92] Goguen JA, Burstall RM (1992) Institutions: abstract model theory for specification and programming. J ACM 39(1):95–146
[Got01] Gottwald S (2001) A treatise on many-valued logics. studies in logic and computation, vol 9. Research Studies Press, Baldock
[Grä79] Grätzer G (1979) Universal algebra. Springer, New York, Heidelberg, Berlin
[Hod97] Hodges W (1997) A shorter model theory. Cambridge University Press, Cambridge
[Ind07] Indrzejczak A (2007) Modal hybrid logic. Logic Log Philos 16:147–257
[Mad13] Madeira A (2013) Foundations and techniques for software reconfigurability. Ph.D. thesis, Universidades do Minho, Aveiro and Porto (Joint MAP-i Doctoral Programme)
[MFMB11] Madeira A, Faria JM, Martins MA, Barbosa LS (2011) Hybrid specification of reactive systems: an institutional approach. In: Barthe G, Pardo A, Schneider G (eds) Software engineering and formal methods (SEFM 2011, Montevideo, Uruguay, November 14–18, 2011). Lecture notes in computer science, vol 7041. Springer, pp 269–285
[Mil89] Milner R (1989) Communication and concurrency. series in computer science. Prentice-Hall, Englewood Cliffs
[MMB13] Madeira A, Martins MA, Barbosa LS (2013) Bisimilarity and refinement for hybrid(ised) logics. In: Derrick J, Boiten EA, Reeves S (eds) Refine-Proceedings 16th international refinement workshop. Electronic proceedings in theoretical computer science, vol 115, pp 84–98
[MMDB11] Martins MA, Madeira A, Diaconescu R, Barbosa LS (2011) Hybridization of institutions. In: Corradini A, KlIn B, Cîrstea C (eds) Algebra and coalgebra in computer science (CALCO 2011, Winchester, UK, August 30–September 2, 2011). Lecture notes in computer science, vol 6859. Springer, pp 283–297
[MML07] Mossakowski T, Maeder C, Lüttich K (2007) The heterogeneous tool set, Hets. In: Grumberg O, Huth M (eds) Tools and algorithms for the construction and analysis of systems (TACAS 2007-Braga, Portugal, March 24–April 1, 2007). Lecture notes in computer science, vol 4424. Springer, pp 519–522
[MNMB13] Madeira A, Neves R, Martins MA, Barbosa LS (2013) When even the interface evolves. In: Wang H, Banach R (eds) Proceedings of TASE (7th IEEE symposium on theoretical aspects of software engineering, Birmingham, July, 2013). IEEE Computer Society, pp 79–82
[MR06] Mossakowski T, Roggenbach M (2006) Structured CSP—a process algebra as an institution. In: Fiadeiro JL, Schobbens P-Y (eds) Recent trends in algebraic development techniques (revised selected papers of WADT 2006, La Roche en Ardenne, Belgium, June 1–3, 2006). Lecture notes in computer science, vol 4409. Springer, pp 92–110
[NMMB13] Neves R, Madeira A, Martins MA, Barbosa LS (2013) Hybridisation at work. In: Heckel R, Milius S (eds) Algebra and coalgebra in computer science—5th international conference, CALCO 2013, Warsaw, Poland, September 3–6, 2013. Proceedings, Lecture notes in computer science, vol 8089, Springer, pp 340–345
[Par81] Park D (1981) Concurrency and automata on infinite sequences. In: Deussen P (ed) Theoretical computer science (5th GI-conference, Karlsruhe, Germany, March 23–25, 1981). Lecture notes in computer science, vol 104. Springer, pp 167–183
[San99] Sannella D (1999) Algebraic specification and program development by stepwise refinement. In: Bossi A (ed) Logic-based program synthesis and transformation. Lecture notes in computer science, vol 1817. Springer, Venezia, Italy, pp 1–9
[San09] Sangiorgi D (2009) On the origins of bisimulation and coinduction. ACM Trans Progr Lang Syst 31(4):1–41. doi:10.1145/1516507.1516510
[SC11] Szepesia R, Ciocarlie H (2011) An overview on software reconfiguration. Theory Appl Math Comput Sci 1:74–79
[SM09] Schröder L, Mossakowski T (2009) HasCasl: integrated higher-order specification and program development. Theor Comput Sci 410(12–13):1217–1260
[ST12] Sannella D, Tarlecki A (2012) Foundations of algebraic specification and formal software development. Monographs on theoretical computer science, an EATCS series. Springer
[Tar03] Tarlecki A (2003) Abstract specification theory: an overview. In: Broy M, Pizka M (eds) Models, algebras, and logics of engineering software. NATO science series, computer and systems sciences, vol 191. IOS Press, pp 43–79
[tC05] ten Cate BD (2005) Model theory for extended modal languages. Ph.D. thesis, Institute for Logic, Language and Computation Universiteit van Amsterdam

Received 5 November 2013
Revised 30 October 2014
Accepted 3 November 2014 by John Derrick, Steve Reeves, and Eerke Boiten
Published online 19 December 2014