THE REVIEW OF SYMBOLIC LOGIC
Volume 3, Number 2, June 2010

A BISIMULATION CHARACTERIZATION THEOREM FOR HYBRID LOGIC WITH THE CURRENT-STATE BINDER

IAN HODKINSON and HICHAM TAHIRI
Imperial College London

Abstract. We prove that every first-order formula that is invariant under quasi-injective bisimulations is equivalent to a formula of the hybrid logic $\mathscr{H}_{\{\downarrow\}}$. Our proof uses a variation of the usual unravelling technique. We also briefly survey related results, and show in a standard way that it is undecidable whether a first-order formula is invariant under quasi-injective bisimulations.

§1. Introduction. Standard modal logics (even Propositional Dynamic Logic or those with Until) have a limitation: they lack a mechanism for referring to specific worlds in the model.

In 1960, Arthur Prior introduced what would become hybrid logic. He used nominals (propositional variables true at only one state) to name worlds. He also introduced hybrid operators, which allow access to a named world or binding of a variable to a world. Hybrid logic already has some computing applications, such as the problem of query and constraint evaluation for semi-structured data: see Franceschet & de Rijke (2006).

Hybrid logic uses a syntax and a semantics close to the modal ones. However, given its ability to talk about a particular state, the expressivity of hybrid logic lies somewhere between modal logic and first-order logic. The exact place depends on which hybrid operators we allow among the most usual ones:

- $@$ moves the evaluation point to a bound state
- $\downarrow$ binds a variable to the current state
- $\exists$ binds a variable to some state in the model.

Therefore, there are many hybrid logics. They are written as $\mathscr{H}_S$, where $S \subseteq \mathscr{O} = \{@, \downarrow, \exists\}$. In this paper, we focus on $\mathscr{H}_{\{\downarrow\}}$, in which the only operator allowed is $\downarrow$.

In his doctoral thesis in the 1970s, published as van Benthem (1985), van Benthem provided an essential bridge between modal and first-order logic. His correspondence theorem (see van Benthem, 1985, theorem 3.9) states that a first-order formula is equivalent to a modal one precisely when it is invariant under (modal) bisimulations. Analogous theorems, using stronger notions of bisimulation, have been proved for $\mathscr{H}_{\{@\}}$ by ten Cate (2005), and for $\mathscr{H}_{\{\downarrow,@\}}$ by Areces *et al.* (2001). The simple cases of $\mathscr{H}_{\{\exists,@\}}$ and $\mathscr{H}_{\mathscr{O}}$ are briefly studied in Section 5 below.

Blackburn & Seligman (1998) introduced the notion of *quasi-injective bisimulation* and showed in Blackburn & Seligman (1998, theorem 1) that in the case without nominals, if a first-order formula is equivalent to a $\mathscr{H}_{\{\downarrow\}}$-sentence then it is invariant under quasi-injective bisimulations. We show in Section 3 of this paper that the converse also

Received: September 16, 2008
AMS 2000 subject classification: 03B45.

doi:10.1017/S1755020309990426

holds. So in this case, invariance under quasi-injective bisimulations is a necessary and sufficient condition to transpose van Benthem's theorem to $\mathscr{H}_{\{\downarrow\}}$.

In Section 4 we show that there is no algorithm to decide whether a first-order formula is invariant under quasi-injective bisimulations. The proof is the same as for modal bisimulations (see van Benthem, 1996, remark 4.19). In Section 5, we briefly consider characterization results for other hybrid logics.

§2. Definitions.

2.1. Syntax and semantics. In first-order logic, we can give names to the elements of the structure, either by having constants in the signature, or during the evaluation of a formula using variables and the binder ∃. Hybrid logics are extensions of modal logic where we name worlds, using one of those solutions or both at the same time. The constants are called *nominals*. The variables are called *world variables*, the binders are ↓, ∃, and we might also want to use @ (the retrieval operator).

The syntax and semantics that we will now introduce are taken from Areces *et al.* (2001).

DEFINITION 2.1. *The set of hybrid operators* $\mathscr{O}$
$\mathscr{O} = \{@, \downarrow, \exists\}$ *is the set of hybrid logic operators.*

DEFINITION 2.2. *Syntax of* $\mathscr{H}_S$
A hybrid signature is a triple of the form ⟨PROP, NOM, WVAR⟩, *where* PROP *is a set of* propositional variables, NOM *a set of* nominals, *and* WVAR $= \{x_0, x_1, \ldots\}$ *a countably infinite set of* world variables. *We assume that these sets are pairwise disjoint. We call* WSYM = NOM∪WVAR *the set of* world symbols, *and* ATOM = PROP∪NOM∪WVAR *the set of* atoms.

The formulas of the hybrid language $\mathscr{H}_{\mathscr{O}}$ *over this signature are:*

$$\varphi ::= \bot \mid a \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \Diamond\varphi \mid \downarrow x\varphi \mid \exists x\varphi \mid @_s\varphi,$$

where $a \in$ ATOM, $x \in$ WVAR, and $s \in$ WSYM. *We will also use* $\mathscr{H}_{\mathscr{O}}$ *to denote the set of such formulas (where the signature is given by context). We define* $\mathscr{H}_S$ *for any* $S \subseteq \mathscr{O}$ *by allowing the use of only the operators in* S.

An occurrence of a world variable x in a hybrid formula is said to be *bound* if it is within the scope of some $\downarrow x$ or $\exists x$, and *free* otherwise. @ does not bind. A *sentence* is a hybrid formula without any free occurrences of world variables.

DEFINITION 2.3. *Semantics of hybrid logics*
A (hybrid) model $\mathfrak{M}$ *for the signature* ⟨PROP, NOM, WVAR⟩ *is a triple of the form* (M, R, V), *where* (M, R) *is a Kripke frame and the valuation* V : PROP ∪ NOM $\longrightarrow$ $\mathscr{P}(M)$ *is such that for all nominals* $i \in$ NOM, $V(i)$ *is a singleton. Elements of* M *are called* worlds of $\mathfrak{M}$.

An assignment g *for* $\mathfrak{M}$ *is a mapping* g : WVAR $\longrightarrow M$. *Given an assignment* g, *a world variable* x, *and* $m \in M$, *we define* g^x_m *(an* x*-variant of* g*) by* $g^x_m(x) = m$ *and* $g^x_m(y) = g(y)$ *for* $y \neq x$.

Let $\mathfrak{M} = (M, R, V)$ *be a model,* $w \in M$, *and* g *an assignment for* $\mathfrak{M}$. *For any atom* a, *write* $[V, g](a)$ *for* $\{g(a)\}$ *if* a *is a world variable, and* $V(a)$ *otherwise. The semantics is then:*

$$\mathfrak{M}, g, w \models \bot \qquad \textit{never}$$

$$\mathfrak{M}, g, w \models a \quad \textit{iff} \quad w \in [V, g](a), \ a \in \text{ATOM}$$

$\mathfrak{M}, g, w \models \neg\varphi$	*iff*	*not* $\mathfrak{M}, g, w \models \varphi$
$\mathfrak{M}, g, w \models \varphi_1 \vee \varphi_2$	*iff*	$\mathfrak{M}, g, w \models \varphi_1$ *or* $\mathfrak{M}, g, w \models \varphi_2$
$\mathfrak{M}, g, w \models \Diamond\varphi$	*iff*	$\mathfrak{M}, g, v \models \varphi$ *for some* $v \in M$ *with* wRv
$\mathfrak{M}, g, w \models \downarrow x\varphi$	*iff*	$\mathfrak{M}, g^x_w, w \models \varphi,\ x \in \text{WVAR}$
$\mathfrak{M}, g, w \models \exists x\varphi$	*iff*	$\mathfrak{M}, g^x_v, w \models \varphi$ *for some* $v \in M,\ x \in \text{WVAR}$
$\mathfrak{M}, g, w \models @_s\varphi$	*iff*	$\mathfrak{M}, g, v \models \varphi$ *where* $[V, g](s) = \{v\},\ s \in \text{WSYM}$.

For a sentence φ, the choice of assignment is irrelevant, so we just write $\mathfrak{M}, w \models \varphi$.

Hybrid logics can express more things than modal logic. For example, in $\mathscr{H}_{\{\downarrow\}}$, we can express the reflexivity of a point with the formula $\downarrow x\Diamond x$, while we cannot in modal logic.

2.2. Standard translation. In a standard way, we can define a link called the *standard translation* between modal logic or hybrid logic and the wider logical world, namely first-order logic. This correspondence is a main topic of Section 3.

2.2.1. The correspondence language. We can define the corresponding first-order language to our hybrid language:

DEFINITION 2.4. *The corresponding language*
Let $\sigma = \langle$PROP, NOM, WVAR$\rangle$ be a hybrid signature. We let L_σ be the signature with a unary predicate P for each propositional variable $p \in$ PROP, a constant I for each nominal $i \in$ NOM, and a binary relation symbol R to denote the accessibility relation in models.

We will use the first-order language with equality in the signature L_σ. We write $\alpha(x)$ to denote a first-order formula with at most one free variable, x.

We will keep using the notation $\mathfrak{M}$, but from now on we will sometimes consider $\mathfrak{M}$ as a hybrid model, and sometimes construe it as a first-order L-structure in the obvious way.

2.2.2. Standard translation Let x be a first-order variable. The standard translation of a hybrid sentence φ is a first-order L-formula $ST_x(\varphi)$ with at most one free variable, x, such that for any model $\mathfrak{M}$ and world w of $\mathfrak{M}$, we have $\mathfrak{M}, w \models \varphi \iff \mathfrak{M} \models ST_x(\varphi)[w]$. See Blackburn & Seligman (1998); Areces *et al.* (2001) for its proper recursive definition.

2.2.3. Which L-formulas are equivalent to $\mathscr{H}_S$-formulas? The question arises of the range of the standard translation. In fact we are not interested in this set itself, but in its closure under logical equivalence. This represents the expressive power of the hybrid language. We will try to characterize it by a bisimulation-invariance property. In the next section, we do this for $S = \{\downarrow\}$. In Section 5, we will consider some other cases.

§3. Bisimulation characterization theorem for $\mathscr{H}_{\{\downarrow\}}$.

We begin this section by recalling from Blackburn & Seligman (1998) the appropriate type of bisimulation for $\mathscr{H}_{\{\downarrow\}}$, and a preservation theorem for it. We will then prove the main theorem of this paper. Before that, we will need two tools. The first one is a new kind of unravelling that we will call $\downarrow$*-unravelling*. The second one is the relation between a pebble game and satisfaction of first-order formulas.

However it is important to notice that the theorem we are giving, and for that matter the preservation theorem in Blackburn & Seligman (1998), covers only the case of sentences without nominals. The general case remains open. So until the end of Section 3, we consider a hybrid signature $\sigma = \langle \text{PROP}, \text{NOM}, \text{WVAR} \rangle$ without nominals: that is, $\text{NOM} = \emptyset$. We write the first-order signature L_σ (Definition 2.4) as L.

3.1. Quasi-injective bisimulations.

DEFINITION 3.1. *Accessibility*
In a model $\mathfrak{M} = (M, R, V)$, we say that the world y is accessible from *the world x iff there are $n \in \mathbb{N}$ with $n > 0$, and a sequence of worlds $x = x_0, x_1, \ldots, x_n = y$ such that $x_i R x_{i+1}$ for each i with $0 \leq i < n$. We write $x \to y$ in this case.*

DEFINITION 3.2. *A* pointed model *is a pair $(\mathfrak{M}, m)$, where $\mathfrak{M} = (M, R, V)$ is a hybrid model and $m \in M$. A* rooted model *is a pointed model $(\mathfrak{M}, r)$ such that r is a* root *of $\mathfrak{M}$: that is, $M = \{r\} \cup \{m \in M : r \to m\}$.*

Recall that for a pointed model $(\mathfrak{M}, m)$, the *submodel of $\mathfrak{M}$ generated by m* is the submodel of $\mathfrak{M}$ whose set of worlds is $\{m\} \cup \{m' \in M : m \to m'\}$.

DEFINITION 3.3. *Quasi-injective bisimulation: ↓-bisimulation (see Blackburn & Seligman, 1998, definition 3)*
Let $(\mathfrak{M}, m)$ and $(\mathfrak{M}', m')$ be pointed models, with $\mathfrak{M} = (M, R, V)$ and $\mathfrak{M}' = (M', R', V')$. A relation $B \subseteq M \times M'$ is said to be a ↓-bisimulation from $(\mathfrak{M}, m)$ to $(\mathfrak{M}', m')$ *(also called* quasi-injective bisimulation*) iff it is a bisimulation:*

- *$(m, m') \in B$,*
- *if $(u, u') \in B$ then $u \in V(a) \iff u' \in V'(a)$ for all $a \in \text{PROP} \cup \text{NOM}$,*
- *if $(u, u') \in B$ and uRv, then there is v' with $u'R'v'$ and $(v, v') \in B$,*
- *if $(u, u') \in B$ and $u'R'v'$, then there is v with uRv and $(v, v') \in B$,*

and moreover,

- *for all $u, v \in M$, $u' \in M'$, if $(u, u') \in B$ and $(v, u') \in B$ and $u \neq v$ then u and v are mutually inaccessible (i.e., neither $u \to v$ nor $v \to u$ holds),*
- *for all $u \in M$, $u', v' \in M'$, if $(u, u') \in B$ and $(u, v') \in B$ and $u' \neq v'$ then u' and v' are mutually inaccessible.*

A functional ↓-bisimulation from $(\mathfrak{M}, m)$ to $(\mathfrak{M}', m')$ *is a function $\pi : M \to M'$ that is a ↓-bisimulation—that is, $\{(x, \pi(x)) : x \in M\}$ is a ↓-bisimulation. We say that $(\mathfrak{M}, m)$ is* ↓-bisimilar *to $(\mathfrak{M}', m')$ if there exists a ↓-bisimulation from $(\mathfrak{M}, m)$ to $(\mathfrak{M}', m')$.*

Blackburn & Seligman (1998, lemma 1) prove a preservation result for $\mathscr{H}_{\{\downarrow\}}$-formulas without nominals under ↓-bisimulations. The proof is easily checked to cover the following very slight strengthening of the original lemma, to use a particular set of free variables.

LEMMA 3.4. *Preservation of $\mathscr{H}_{\{\downarrow\}}$-formulas by ↓-bisimulations*
Let B be a quasi-injective bisimulation from $(\mathfrak{M}, m)$ to $(\mathfrak{N}, n)$, let g_1, g_2 be assignments for $\mathfrak{M}, \mathfrak{N}$, respectively, and let X be a set of world variables, such that for all $x \in X$:

1. *$g_1(x)Bv$ iff $v = g_2(x)$, for all v in the submodel of $\mathfrak{N}$ generated by n, and*
2. *$uBg_2(x)$ iff $u = g_1(x)$, for all u in the submodel of $\mathfrak{M}$ generated by m.*

Then $\mathfrak{M}, g_1, m \models \varphi$ iff $\mathfrak{N}, g_2, n \models \varphi$, for all $\varphi \in \mathscr{H}_{\{\downarrow\}}$ whose free variables all lie in X.

Taking $X = \emptyset$, we obtain the result of Blackburn & Seligman (1998, theorem 1) that $\mathscr{H}_{\{\downarrow\}}$-sentences are preserved by ↓-bisimulations.

DEFINITION 3.5. *Invariance under $\downarrow$-bisimulations*

An L-formula $\varphi(x)$ is said to be invariant under $\downarrow$-bisimulations *if whenever* $(\mathfrak{M}, m)$ *and* $(\mathfrak{N}, n)$ *are $\downarrow$-bisimilar, we have $\mathfrak{M} \models \varphi[m]$ iff $\mathfrak{N} \models \varphi[n]$.*

So any L-formula equivalent to the standard translation of an $\mathscr{H}_{\{\downarrow\}}$-sentence is invariant under $\downarrow$-bisimulations. We will be proving a converse to this theorem. In doing so, we will be applying Lemma 3.4 for functional $\downarrow$-bisimulations, so we now derive a corollary covering that case.

COROLLARY 3.6. *Preservation of $\mathscr{H}_{\{\downarrow\}}$-formulas by functional $\downarrow$-bisimulations*

Let π be a functional $\downarrow$-bisimulation from $(\mathfrak{M}, m)$ to $(\mathfrak{N}, n)$, let g_1, g_2 be assignments for $\mathfrak{M}, \mathfrak{N}$, respectively, and let X be a set of world variables, such that for every $x \in X$, we have $\pi(g_1(x)) = g_2(x)$, and for every u in the submodel of $\mathfrak{M}$ generated by m, if $\pi(u) = g_2(x)$ then $u = g_1(x)$. Then $\mathfrak{M}, g_1, m \models \varphi$ iff $\mathfrak{N}, g_2, n \models \varphi$, for all $\varphi \in \mathscr{H}_{\{\downarrow\}}$ whose free variables all lie in X.

Proof. Simply check that this hypothesis implies the one of Lemma 3.4. □

3.2. $\downarrow$-unravelling. The standard unravelling technique whose idea can be found in Blackburn *et al.* (2001) involves taking the tree consisting of all paths from the root. This is used with $\mathscr{H}_{\{\downarrow\}}$ in Blackburn & Seligman (1998) for strict partial orders. But for arbitrary models, it unravels too much. First, nominals must remain true in exactly one world: this is why it helps if there are no nominals. Second, we cannot unravel a cycle into a tree, as $\mathscr{H}_{\{\downarrow\}}$-formulas like $\downarrow x \Diamond^n x$ are not preserved. We will try in fact to preserve the following relation during our unravelling process.

DEFINITION 3.7. *$\simeq_{\mathfrak{M}}$-equivalence*

Given a model $\mathfrak{M} = (M, R, V)$, the equivalence relation $\simeq_{\mathfrak{M}}$ is defined on M by: $x \simeq_{\mathfrak{M}} y$ iff $x = y$ or x is accessible from y and y accessible from x.

A $\simeq_{\mathfrak{M}}$-class will be called a component of $\mathfrak{M}$.

Here is how we modify the standard unravelling. We do want to follow paths through the model, but on such a path, we forget all parts that lie inside a single component. We only keep the edges going from one component to another. We also incorporate numbers $n_0, n_1, \dots$, that serve to copy each component infinitely many times.

DEFINITION 3.8. *$\downarrow$-unravelling*

The $\downarrow$-unravelling *of a pointed model $(\mathfrak{M}, r)$, where $\mathfrak{M} = (M, R, V)$, is a rooted model $(\mathfrak{M}^*, r^*)$ with $\mathfrak{M}^* = (M^*, R^*, V^*)$. The elements of M^* are all finite sequences of the form $(e_0, n_0, \dots, e_{l-1}, n_{l-1}, x)$, where*

- *$l < \omega$ (l is the* number of parameters *of $(e_0, n_0, \dots, e_{l-1}, n_{l-1}, x)$)*
- *$n_0, \dots, n_{l-1} < \omega$*

and there exist components $E_0, \dots, E_l \subseteq M$ such that

- *the E_i are pairwise distinct*
- *$r \in E_0$*
- *for each $i < l$, e_i is an R-edge in $\mathfrak{M}$ from E_i to E_{i+1} (i.e., $e_i = (a, b)$ for some $a \in E_i$, $b \in E_{i+1}$ with aRb)*
- *$x \in E_l$.*

The E_i are necessarily unique.

We define R^* by: $(e_0, n_0, \dots, e_{l-1}, n_{l-1}, x) R^* (d_0, m_0, \dots, d_{l'-1}, m_{l'-1}, y)$ iff xRy and one of the following holds:

1. $(d_0, m_0, \ldots, d_{l'-1}, m_{l'-1}) = (e_0, n_0, \ldots, e_{l-1}, n_{l-1})$,
2. $(d_0, m_0, \ldots, d_{l'-1}) = (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, (x, y))$.

We define $r^* = (r) \in M^*$. *This is a root of* $\mathfrak{M}^*$.

Finally we put $V^*(p) = \{(e_0, n_0, \ldots, e_{l-1}, n_{l-1}, x) : x \in V(p)\}$, *for each propositional variable* p.

There is an example of this unravelling in Figure 1, with three atoms, each made true at only one world in the original model. The components are delimited by dashed lines in each case.

However there would be no point in defining a new rooted model from $(\mathfrak{M}, r)$ if the two did not share some logical properties. In fact, as in the modal case, where the unravelling and the original are bisimilar, a pointed model is $\downarrow$-bisimilar to its $\downarrow$-unravelling, using the following function:

DEFINITION 3.9. *The projection* π

Given a pointed model $(\mathfrak{M}, r)$ *and its* $\downarrow$*-unravelling* $(\mathfrak{M}^*, r^*)$, *the projection* π *is defined by:*

$$\begin{array}{rccc} \pi : & \mathfrak{M}^* & \longrightarrow & \mathfrak{M} \\ & (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, x) & \longmapsto & x \end{array}$$

PROPOSITION 3.10. $\downarrow$*-bisimulation between a model and its* $\downarrow$*-unravelling*
π *is a functional* $\downarrow$*-bisimulation from* $(\mathfrak{M}^*, r^*)$ *to* $(\mathfrak{M}, r)$.

Proof. Clearly, $\pi(r^*) = r$. The conservation of atoms comes from the definition of V^*.

Assume that uR^*v, where $u = (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, x)$ and $v = (d_0, m_0, \ldots, d_{l'-1}, m_{l'-1}, y)$. By definition of R^*, xRy, so $\pi(u)R\pi(v)$.

On the other side, suppose that $u = (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, x) \in M^*$ and xRy. We seek $v \in M^*$ with uR^*v and $\pi(v) = y$. If $x \simeq_{\mathfrak{M}} y$, then we take $v = (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, y)$. If not, we take $v = (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, (x, y), 0, y)$.

Finally, since π is a function, we just have to check the 'back' direction of quasi-injectivity. Let $u, v \in M^*$ be distinct states with $\pi(u) = \pi(v) = x$, and assume for contradiction that there is an R^*-path from u to v. Let $u = (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, x)$. By definition of R^*, we have $v = (e_0, n_0, \ldots, e_{l-1}, n_{l-1}, e_l, n_l, \ldots, e_{l-1+k}, n_{l-1+k}, x)$

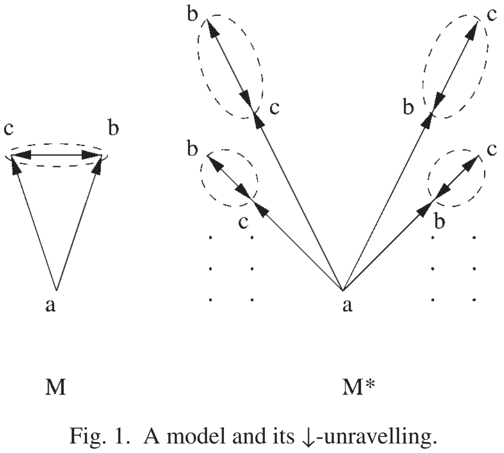

Fig. 1. A model and its $\downarrow$-unravelling.

for some $k > 0$. Let $e_i = (e_i^0, e_i^1)$. For each $i < k - 1$, we have $e^1_{l+i} \simeq_{\mathfrak{M}} e^0_{l+i+1}$. Therefore $x \simeq_{\mathfrak{M}} e_l^0 R e_l^1 \simeq_{\mathfrak{M}} e^0_{l+1} R e^1_{l+1} \ldots e^1_{l-1+k} \simeq_{\mathfrak{M}} x$, and so there is a path from x to x itself. So all the points on this path are in the same component. This contradicts the fact that e_l^0 and e_l^1 are not in the same component. □

Another interesting property:

PROPOSITION 3.11. *Components of* $\mathfrak{M}^*$
The component of $(e_0, n_0, \ldots, e_{l-1}, n_{l-1}, x)$ *is* $\{(e_0, n_0, \ldots, e_{l-1}, n_{l-1}, y) \in M^* \mid x \simeq_{\mathfrak{M}} y\}$.

Proof. We just have to see that the number of parameters of the states keeps increasing along a path, and is stable iff we remain in the same component. □

DEFINITION 3.12. *Bridge*
For a component $C = \{(e_0, n_0, \ldots, (e^0_{l-1}, e^1_{l-1}), n_{l-1}, y) \in M^* \mid e^1_{l-1} \simeq_{\mathfrak{M}} y\}$ *with* $l > 0$, *we call* $a = (e_0, n_0, \ldots, (e^0_{l-1}, e^1_{l-1}), n_{l-1}, e^1_{l-1})$ *the* end of the bridge to C, *and* $b = (e_0, n_0, \ldots, e_{l-2}, n_{l-2}, e^0_{l-1})$ *the* beginning of the bridge to C. *We call* (b, a) *the* bridge to C.

We also define the end of the bridge to the component of the root to be the root itself. There is no beginning of the bridge in this case.

By definition of R^*, an R^*-edge in $\mathfrak{M}^*$ is either a bridge to a component or a link between two worlds in the same component.

Another graph result shows that all the paths from the root to a specific point are forced to go through a certain edge, namely the bridge to its component.

PROPOSITION 3.13. *For any point* $u \in M^*$, *for any path from the root* r^* *to* u, *the path has to go through the end of the bridge to the component of* u.

Proof. The case where u is in the same component as the root is obvious. Now assume that $u = (e_0, n_0, \ldots, (e^0_{l-1}, e^1_{l-1}), n_{l-1}, x)$, with $l > 0$. Assume $u_0 R^* u_1 R^* \ldots R^* u_k$ is a path from the root to u. Since the number of parameters increases along a path, and since we remain in the same component as long as we do not increase its value, there is an i such that the number of parameters of u_i is $l - 1$ and the number of parameters of u_{i+1} is l, with u_{i+1} in the same component as $u_k = u$. By definition of R^*, (u_i, u_{i+1}) is a bridge to the component of u. □

In fact it also proves that the beginning of the bridge has to be visited in the path (except in the case where it is not even defined), so the bridge to a component is also a bridge in the sense of the graph theory vocabulary (an edge whose removal disconnects a graph).

3.3. Games. The game formulated by Andrzej Ehrenfeucht known as the Ehrenfeucht–Fraïssé game or sometimes as the back-and-forth game $G(A, B)$ takes place on two structures A and B with the same signature $\mathscr{L}$. It is described essentially in Hodges (1993, p. 95). There are two players, the male $\forall$ and the female $\exists$. There are ω rounds numbered $0, 1, 2, \ldots$. Each round of the game begins by $\forall$ choosing an element in one of the two structures. In response, $\exists$ chooses an element in the other structure. A pair of such chosen elements is marked by two pebbles. At the start of round n, we have pebbles located at $a_0, \ldots, a_{n-1}$ in A and $b_0, \ldots, b_{n-1}$ in B, where the index corresponds to the round when the element was chosen. This position is a win for $\forall$ iff some atomic $\mathscr{L}$-formula with at most n variables is satisfied by $(a_0, \ldots, a_{n-1})$ and not by $(b_0, \ldots, b_{n-1})$, or vice versa. $\exists$ wins if she has not lost in any round, after an infinite number of rounds.

This game is related to the satisfaction of formulas of the infinitary logic $\mathscr{L}_{\infty\omega}$ (see Karp, 1963; Hodges, 1993, p. 97 & corollary 3.5.3, for more details):

THEOREM 3.14. *Relation between $G(A, B)$ and $\mathscr{L}_{\infty\omega}$*
$\exists$ has a winning strategy in the game $G(A, B)$ iff A, B agree on all sentences of $\mathscr{L}_{\infty\omega}$. This implies that they agree on all first-order sentences.

We introduce another game that is equivalent in the case of rooted Kripke models $(\mathfrak{M}, m)$ and $(\mathfrak{N}, n)$. First we need to introduce another signature.

DEFINITION 3.15. *Signature L'.*
The signature L' is obtained by adding a new constant ∇ to our signature L. A rooted model $(\mathfrak{M}, m)$ is regarded as an L'-structure by regarding $\mathfrak{M}$ as an L-structure as previously explained, and interpreting ∇ as m.

DEFINITION 3.16. *Successor game $G'((\mathfrak{M}, m), (\mathfrak{N}, n))$*
If $(\mathfrak{M}, m)$ and $(\mathfrak{N}, n)$ are two rooted models, $G'((\mathfrak{M}, m), (\mathfrak{N}, n))$ is a game with the same winning conditions as in $G((\mathfrak{M}, m), (\mathfrak{N}, n))$. However, now:
In round 0, $\forall$ has to choose in $\{m, n\}$. At each round $k > 0$, assuming he played x_{k-1} at round $k - 1$, $\forall$ has to choose the next state in $\{m, n\} \cup \{x \mid x_{k-1}Rx\}$ where R is the accessibility relation of $\mathfrak{M}$ or $\mathfrak{N}$, depending on where is x_{k-1}.

As we show now, because of the accessibility of every world from the root, this game is equivalent to the original one:

THEOREM 3.17. *$\exists$ has a winning strategy in $G'((\mathfrak{M}, m), (\mathfrak{N}, n))$ iff she has a winning strategy in $G((\mathfrak{M}, m), (\mathfrak{N}, n))$.*

Proof. The direction from right to left is obvious. Now assume that $\exists$ has a winning strategy in $G'((\mathfrak{M}, m), (\mathfrak{N}, n))$. Here is the winning strategy she would use in $G((\mathfrak{M}, m), (\mathfrak{N}, n))$. She would have a personal secret board where she would play $G'((\mathfrak{M}, m), (\mathfrak{N}, n))$ alone. In the course of $G((\mathfrak{M}, m), (\mathfrak{N}, n))$, whenever $\forall$ puts a pebble in position x in $\mathfrak{M}$ for example, $\exists$ chooses a path from the root to x. On her own board, she plays as $\forall$ the succession of positions on this path, answering by using her winning strategy. She copies the location of her final $\exists$-pebble to the original board. Since the pebbles present on the original board are a subset of those on the hidden one, it is obvious that this is a winning strategy. □

Obviously, if $\exists$ has played y_k in round k of $G'((\mathfrak{M}, m), (\mathfrak{N}, n))$, in round $k + 1$ she will have to play in $\{m, n\} \cup \{y \mid y_k Ry\}$ (otherwise she would lose directly). That is why in this game, at any round, for any pebble there is a path made of pebbles from the root to this pebble, unless the position is a win for $\forall$ of course.

In the case we are playing on two $\downarrow$-unravellings, we conclude using Proposition 3.13 that if we are not in a win position for $\forall$, then for any pebble in the game, there is a pebble on the end of the bridge to the component where this pebble is, and at the beginning of this bridge (when it is defined).

3.4. Bisimulation characterization theorem for $\mathscr{H}_{\{\downarrow\}}$.

In order to state the characterization theorem, we need the notion of hybrid equivalence:

DEFINITION 3.18. *The S-hybrid equivalence $\equiv_S$*
For $S \subseteq \mathscr{O}$ and two pointed models $(\mathfrak{M}, m)$, $(\mathfrak{N}, n)$, we write $(\mathfrak{M}, m) \equiv_S (\mathfrak{N}, n)$ if $\mathfrak{M}, m \models \varphi \iff \mathfrak{N},n \models \varphi$ for every sentence φ in $\mathscr{H}_S$.

The idea of the following lemma is well known and is nearly identical to Blackburn *et al.*, pp. 103–104), where it is used to prove the bisimulation characterization theorem in the modal case. So we only sketch the proof.

LEMMA 3.19. *For any $S \subseteq \mathscr{O}$, an L-formula $\alpha(x)$ is equivalent to the standard translation of a sentence of $\mathscr{H}_S$ iff for any couple of ω-saturated pointed models $(\mathfrak{M}, m)$ and $(\mathfrak{N}, n)$ such that*

- $(\mathfrak{M}, m) \equiv_S (\mathfrak{N}, n)$,
- $\mathfrak{M} \models \alpha[m]$,

we have $\mathfrak{N} \models \alpha[n]$.

Proof. The direction from left to right is obvious. Now, assume the right side of the lemma. Let

$$C_S(\alpha) = \{ST_x(\varphi) \mid \varphi \text{ an } \mathscr{H}_S\text{-sentence}, \alpha(x) \models ST_x(\varphi)\}.$$

If $C_S(\alpha) \models \alpha(x)$ then by compactness, $\alpha(x)$ is equivalent to the standard translation of an $\mathscr{H}_S$-sentence. Therefore, we just have to show that $C_S(\alpha) \models \alpha(x)$. Assume $\mathfrak{N} \models C_S(\alpha)[n]$; we need to show that $\mathfrak{N} \models \alpha[n]$. We may assume that $\mathfrak{N}$ is ω-saturated. Let

$$T = \{\alpha(x)\} \cup \{ST_x(\varphi) \mid \varphi \text{ an } \mathscr{H}_S\text{-sentence}, \mathfrak{N}, n \models \varphi\}.$$

Again by compactness, T is consistent. Let $(\mathfrak{M}, m)$ be an ω-saturated structure such that $\mathfrak{M} \models T[m]$. It is easy to see that $(\mathfrak{M}, m) \equiv_S (\mathfrak{N}, n)$. Now, using our hypothesis on $(\mathfrak{M}, m)$ and $(\mathfrak{N}, n)$, we obtain $\mathfrak{N} \models \alpha[n]$ as required. □

We come now to our main result. We repeat that the proof assumes that the set NOM of nominals in the signature is empty.

THEOREM 3.20. *Let $\varphi(x)$ be an L-formula. Then φ is equivalent to the standard translation of an $\mathscr{H}_S$-sentence iff it is invariant by $\downarrow$-bisimulations.*

Proof. The direction from left to right comes from Lemma 3.4.

Now, for the other direction, using Lemma 3.19, we just have to show that for any L-formula $\varphi(x)$ invariant under $\downarrow$-bisimulations, if $(\mathfrak{M}, m)$ and $(\mathfrak{N}, n)$ are two ω-saturated pointed models such that $(\mathfrak{M}, m) \equiv_{\{\downarrow\}} (\mathfrak{N}, n)$ and $\mathfrak{M} \models \varphi[m]$, then $\mathfrak{N} \models \varphi[n]$. We will call R the relation in both $\mathfrak{M}$ and $\mathfrak{N}$ to simplify notations. Their $\downarrow$-unravellings will be $(\mathfrak{M}^*, m^*)$ and $(\mathfrak{N}^*, n^*)$, respectively. Again, we will use the same notation, R^*, for their accessibility relations, and π for the two projections onto the original models.

We are going to show that $\exists$ has a winning strategy in $G'((\mathfrak{M}^*, m^*), (\mathfrak{N}^*, n^*))$. After that: since by Proposition 3.10, $(\mathfrak{M}^*, m^*)$ and $(\mathfrak{M}, m)$ are $\downarrow$-bisimilar, and since φ is invariant by $\downarrow$-bisimulations, we obtain $\mathfrak{M}^* \models \varphi[m^*]$. Therefore $\mathfrak{M}^* \models \varphi(\nabla)$. $(\mathfrak{M}^*, m^*)$ and $(\mathfrak{N}^*, n^*)$ agree on first-order sentences of L', so $\mathfrak{N}^* \models \varphi(\nabla)$. Hence, $\mathfrak{N}^* \models \varphi[n^*]$. Finally, again because of the $\downarrow$-bisimilarity between a rooted model and its $\downarrow$-unravelling, $\mathfrak{N} \models \varphi[n]$.

Here is $\exists$'s winning strategy. In the initial round, $\forall$ is forced to play one root. Let the answer be the other root.

Suppose at the start of round t of the game, where $t > 0$, the pebbles in play in $\mathfrak{M}^*$ are on elements $a_0, \dots, a_{t-1}$ of $\mathfrak{M}^*$ and there are corresponding pebbles on elements $b_0, \dots, b_{t-1}$ in $\mathfrak{N}^*$. We call f_{t-1} and g_{t-1} the assignments to the variables $\{x_i : i < t\}$ such that for every $i < t$, $f_{t-1}(x_i) = a_i$ and $g_{t-1}(x_i) = b_i$.

We assume inductively that:

1. $a_i R^* a_j \iff b_i R^* b_j$, for all $i, j < t$.
2. $a_i \simeq_{\mathfrak{M}^*} a_j \iff b_i \simeq_{\mathfrak{N}^*} b_j$, for all $i, j < t$.
3. For $i, j < t$, put $i \sim j$ iff $a_i \simeq_{\mathfrak{M}^*} a_j$. If $\{p_1, \dots, p_s\}$ is any $\sim$-equivalence class, then for every $\mathscr{H}_{\{\downarrow\}}$-formula $\alpha(x_{p_1} \dots x_{p_s})$ in which at most these world variables occur free, we have for each $j = 1, \dots, s$:

$$\mathfrak{M}^*, f_{t-1}, a_{p_j} \models \alpha(x_{p_1} \dots x_{p_s}) \iff \mathfrak{N}^*, g_{t-1}, b_{p_j} \models \alpha(x_{p_1} \dots x_{p_s}).$$

We check that the properties hold for $t = 1$. Note that $a_0 = m^*$ and $b_0 = n^*$. Let $\alpha(x_0)$ be any $\mathscr{H}_{\{\downarrow\}}$-formula. Let h, j be assignments for $\mathfrak{M}, \mathfrak{N}$ (respectively) with $h(x_0) = m$ and $j(x_0) = n$. By Proposition 3.10, $\pi : \mathfrak{M}^* \to \mathfrak{M}$ is a functional $\downarrow$-bisimulation. Applying Corollary 3.6 with $X = \{x_0\}$, we obtain $\mathfrak{M}^*, f_0, a_0 \models \alpha$ iff $\mathfrak{M}, h, m \models \alpha$. The latter is clearly iff $\mathfrak{M}, m \models \downarrow x_0 \alpha$, iff $\mathfrak{N}, n \models \downarrow x_0 \alpha$ (since $(\mathfrak{M}, m)$ and $(\mathfrak{N}, n)$ agree on $\mathscr{H}_{\{\downarrow\}}$-sentences), iff $\mathfrak{N}, j, n \models \alpha$, iff $\mathfrak{N}^*, g_0, b_0 \models \alpha$ (by Corollary 3.6 again). So Point 3 holds. Point 2 is trivial, and Point 1 follows from Point 3 using the formula $\alpha = \Diamond x_0$.

Let $\forall$ play round t by choosing without loss of generality an element $a_t \in \mathfrak{M}^*$. (If he chooses in $\mathfrak{N}^*$ the situation is handled similarly.) $\exists$ must select some b_t in $\mathfrak{N}^*$ as her response. She must either choose a successor of a_{t-1} or one of the two roots. Once she has done this, since a_t, b_t have been defined, f_t, g_t will be defined as well.

If $a_t = a_u$ for some $u < t$ (e.g., because $\forall$ chooses a_t to be the root m^*), then $\exists$ responds by choosing b_u. This answer is well defined. Indeed, assume that $a_t = a_u = a_v$ with $u, v < t$. Then because of Point 2, $b_v \simeq_{\mathfrak{N}^*} b_u$, so $u \sim v$. Because of Point 3, since $\mathfrak{M}^*, f_{t-1}, a_u \models x_v$, we have $\mathfrak{N}^*, g_{t-1}, b_u \models x_v$, which means $b_u = b_v$.

Let us show now that this choice of b_u satisfies the inductive hypothesis. The first two points are obvious. For the third point, let $\{p_1, \dots, p_s\}$ be the $\sim$-class containing u. Then for any formula $\alpha(x_{p_1} \dots x_{p_s}, x_t)$, in which we may assume no variable occurs both free and bound, and any $j \in \{p_1, \dots, p_s, t\}$, we have $\mathfrak{M}^*, f_t, a_j \models \alpha(x_{p_1} \dots x_{p_s}, x_t)$ iff $\mathfrak{M}^*, f_{t-1}, a_j \models \alpha(x_{p_1} \dots x_{p_s}, x_u)$, which inductively is iff $\mathfrak{N}^*, g_{t-1}, b_j \models \alpha(x_{p_1} \dots x_{p_s}, x_u)$, iff $\mathfrak{N}^*, g_t, b_j \models \alpha(x_{p_1} \dots x_{p_s}, x_t)$.

From now on, assume that $a_t \notin \{a_0 \dots a_{t-1}\}$, so that $R^*(a_{t-1}, a_t)$, there is no pebble on a_t, and

$$\begin{aligned} a_{t-1} &= (e_0, m_0, \dots, e_{l-1}, m_{l-1}, x), \\ b_{t-1} &= (e'_0, n_0, \dots, e'_{l'-1}, n_{l'-1}, y). \end{aligned}$$

Let $\{p_1, \dots, p_s\}$ be the $\sim$-class containing $t-1$, with $p_s = t-1$. Let

$$\Delta = \{\alpha(x_{p_1}, \dots, x_{p_s}, x_t) \in \mathscr{H}_{\{\downarrow\}} \mid \mathfrak{M}^*, f_t, a_t \models \alpha\}.$$

Take any finite $\Delta_0 \subseteq \Delta$, and let $\bigwedge \Delta_0 = \alpha(x_{p_1}, \dots, x_{p_s}, x_t)$. Then we have $\mathfrak{M}^*, f_{t-1}, a_{t-1} \models \Diamond \downarrow x_t\, \alpha(x_{p_1}, \dots x_{p_s}, x_t)$. Inductively, $\mathfrak{N}^*, g_{t-1}, b_{t-1} \models \Diamond \downarrow x_t\, \alpha(x_{p_1} \dots x_{p_s}, x_t)$. We have

$$y = \pi(b_{t-1}).$$

By $\downarrow$-bisimulation (formally, by applying Corollary 3.6 to the pointed models $(\mathfrak{N}^*, b_{t-1})$ and $(\mathfrak{N}, y)$, with $X = \{x_{p_1}, \dots, x_{p_s}\}$), we obtain $\mathfrak{N}, h_{t-1}, y \models \Diamond \downarrow x_t\, \alpha(x_{p_1} \dots x_{p_s}, x_t)$, where h_{t-1} is an assignment for $\mathfrak{N}$ satisfying $h_{t-1}(x_k) = \pi(b_k)$ for $k < t$. This holds for all Δ_0. By ω-saturation of $\mathfrak{N}$, there is β in $\mathfrak{N}$ such that $yR\beta$ and $\mathfrak{N}, h_t, \beta \models \Delta$, where h_t is an x_t-variant of h_{t-1} such that $h_t(x_t) = \beta$. Therefore, for every $\mathscr{H}_{\{\downarrow\}}$-formula

$\alpha(x_{p_1}, \dots, x_{p_s}, x_t)$, we have

$$\mathfrak{M}^*, f_t, a_t \models \alpha \iff \alpha \in \Delta \iff \mathfrak{N}, h_t, \beta \models \alpha. \tag{1}$$

The back condition comes from the fact that in proving the contraposition, the negated formula is in Δ.

We then have two cases where the choice of b_t is going to be different. In both of them, we will ensure that $\pi(b_t) = \beta$, and that for each $x \in \{x_{p_1}, \dots, x_{p_s}, x_t\}$ and each u in the submodel of $\mathfrak{N}^*$ generated by b_t, if $\pi(u) = h_t(x)$ then $u = g_t(x)$. So by $\downarrow$-bisimulation (Corollary 3.6), for every $\mathscr{H}_{\{\downarrow\}}$-formula $\alpha(x_{p_1}, \dots, x_{p_s}, x_t)$ we will have

$$\mathfrak{N}^*, g_t, b_t \models \alpha \iff \mathfrak{N}, h_t, \beta \models \alpha. \tag{2}$$

- First case: $y \not\simeq_{\mathfrak{N}} \beta$. Let $b_t = (e'_0, n_0, \dots, e'_{l'-1}, n_{l'-1}, (y, \beta), q, \beta)$, with q the least integer not chosen by any other pebble before, in the sense that it is not one of the integer parameters of any pebbles. b_t is a well-defined element of $\mathfrak{N}^*$. Let us show that $(a_0, \dots, a_t)$ and $(b_0, \dots, b_t)$ verify the induction hypothesis.
  First a_t is also the end of a bridge. Otherwise there would be a path from a_t to a_{t-1}, so $\Diamond^k x_{t-1} \in \Delta$ for some integer k, which by (1) would also lead β to be in the same component as y.
  Because of the choice of q, there is no pebble on b_t. Since both b_t and a_t are ends of bridges and there are no pebbles on them, there cannot be any other pebble in their components. This proves Point 2.
  The result for Point 3 is simple, using the subset of Δ whose formulas only use x_t as a free world variable.
  For Point 1, because of the result for Point 2, we only have to investigate four cases: $a_i R^* a_t$ with $i < t$, $a_t R^* a_t$, $b_i R^* b_t$, and $b_t R^* b_t$. If $a_i R^* a_t$, then because there are no other pebbles in the component of a_t, we have $a_i = a_{t-1}$ so $b_i R^* b_t$. The exact same reasoning holds for $b_i R^* b_t$. $a_t R^* a_t$ iff $\Diamond x_t \in \Delta$ iff $b_t R^* b_t$.
- If $y \simeq_{\mathfrak{N}} \beta$, let $b_t = (e'_0, n_0, \dots, e'_{l'-1}, n_{l'-1}, \beta)$, a well-defined element of $\mathfrak{N}^*$. We have $b_t \simeq_{\mathfrak{N}^*} b_{t-1}$, so there is a path of length q for some q from b_t to b_{t-1}. By $\downarrow$-bisimulation, $\mathfrak{N}, h_t, \beta \models \Diamond^q x_{t-1}$. It is then also satisfied by a_t, so $a_t \simeq_{\mathfrak{M}^*} a_{t-1}$, so by using the induction hypothesis, the second point is proved.
  Now there can be no pebble on b_t. For otherwise, since $b_t \simeq_{\mathfrak{N}^*} b_{t-1}$, we must have $b_t \in \{b_{p_1}, \dots, b_{p_s}\}$. Say, $b_t = b_{p_i}$. So $x_{p_i} \in \Delta$. Therefore, $a_t = a_{p_i}$, contrary to our assumption that there is no pebble on a_t.
  We now verify the first point. If $a_t R^* a_j$ for $j \leq t$, then a_j is in the same component as a_t: otherwise, since a_t would be the beginning of a bridge to a component where there is a pebble, there would be a pebble on it, contrary to assumption. Therefore, $\Diamond x_j \in \Delta$, so $b_t R^* b_j$. We have the same thing if $b_t R^* b_j$. If $a_j R^* a_t$ for $j < t$, then a_j has also to be in the same component as a_t. Otherwise a_t would be the end of a bridge to a component where there is already a pebble (namely a_{t-1}, because we know $a_{t-1} \simeq_{\mathfrak{M}^*} a_t$), so there should be a pebble on a_t. We know $a_j \neq a_t$, so $a_t \rightarrow a_j$. Assume we can go from a_t to a_j in q steps. Therefore, the formula $\Diamond^q (x_j \wedge \Diamond x_t)$ is in Δ. It is then also true at b_t, so $b_j R^* b_t$. We prove the same thing in the case of $b_j R^* b_t$ by using the negated formula. This proves the first point.

For the last point, assume for example that $\mathfrak{M}^*, f_t, a_{p_j} \models \alpha$, where $\alpha \in \mathscr{H}_{\{\downarrow\}}$ is a formula with only $x_{p_1} \dots x_{p_s}, x_t$ as free world variables. As previously, there is a q such that $\Diamond^q(x_{p_j} \wedge \alpha) \in \Delta$, so it is also true at b_t, and $\mathfrak{N}^*, g_t, b_{p_j} \models \alpha$. The case where $\mathfrak{M}^*, g_t, a_t \models \alpha$ is even simpler, using $\alpha \in \Delta$ itself. Using $\neg\alpha$, we prove again the equivalence.

It is clear that given the inductive hypothesis and the evident facts that $\mathfrak{M}^* \models a_0 = \nabla$ and $\mathfrak{N}^* \models b_0 = \nabla$, we have $\mathfrak{M}^* \models \psi(a_0, \dots, a_t) \iff \mathfrak{N}^* \models \psi(b_0, \dots, b_t)$ for all $t < \omega$ and all atomic L'-formulas $\psi(x_0, \dots, x_t)$. Therefore, this is a winning strategy for $\exists$. □

However one important question remains: is the result still valid for formulas with nominals?

§4. Undecidability of invariance by bisimulation. Given a first-order L-formula $\varphi(x)$, the existence of an equivalent $\mathscr{H}_{\{\downarrow\}}$-formula depends on whether $\varphi(x)$ is invariant by quasi-injective bisimulations. However, we are going to see that this property is not decidable.

THEOREM 4.1. *Undecidability of invariance under bisimulations*
It is not decidable whether a particular first-order L-formula is invariant under quasi-injective bisimulations.

Proof. van Benthem (1996, remark 4.19) proves that it is undecidable whether a first-order formula is invariant under modal bisimulations. The same proof works for quasi-injective bisimulations, as we will now see.

First, let $\varphi_N(x) = \exists y(yRx)$. Note that this is not invariant under quasi-injective bisimulations. Indeed, let $M = \{a\}$, $M' = \{0, 1\}$, $R' = \{(0, 1)\}$, $\mathfrak{M} = (M, \emptyset, V)$, and $\mathfrak{M}' = (M', R', V)$, where V is the valuation on an empty domain (no propositional variables). Then $(\mathfrak{M}, a)$ and $(\mathfrak{M}', 1)$ are quasi-injective bisimilar, while one satisfies $\exists y(yRx)$ at 1 and the other one does not at a.

Let σ^P be the relativization to P of σ (where P is a unary predicate not in L) defined inductively by:

$$\begin{aligned} \alpha(\bar{x})^P &= \alpha(\bar{x}), \alpha \text{ atomic} \\ (\sigma_1 \vee \sigma_2)^P &= \sigma_1^P \vee \sigma_2^P \\ (\neg\sigma)^P &= \neg(\sigma^P) \\ (\exists y\sigma)^P &= \exists y(P(y) \wedge \sigma^P). \end{aligned}$$

Let f be the map defined on the set of first-order L-sentences by:

$$f(\sigma) = \varphi_N(x) \vee (\exists x P(x) \to \sigma^P).$$

We are going to prove that σ is valid iff $f(\sigma)$ is quasi-injective bisimilar invariant. After that, since it is well known that validity of first-order L-sentences is undecidable, we will have finished.

First assume that σ is valid. Then $\exists x P(x) \to \sigma^P$ is also valid, which means $f(\sigma)$ is valid. Therefore, $f(\sigma)$ has the same truth value in any model, so trivially is invariant under quasi-injective bisimulations.

On the other hand, if σ is not valid, take $\mathfrak{L}$ such that $\mathfrak{L} \models \neg\sigma$. Let $\mathfrak{N}$ (resp. $\mathfrak{N}'$) be the disjoint union of $\mathfrak{L}$ and $\mathfrak{M}$ (resp. $\mathfrak{M}'$), where the new predicate P is made true in the domain of $\mathfrak{L}$.

Since $\mathfrak{M}' \models \varphi_N[1]$, we have $\mathfrak{N}' \models \varphi_N[1]$, and therefore $\mathfrak{N}' \models f(\sigma)[1]$. However, $\mathfrak{N} \not\models \varphi_N[a]$, and $\mathfrak{N} \not\models \exists x P(x) \rightarrow \sigma^P$ (otherwise σ would hold in $\mathfrak{L}$). Therefore $\mathfrak{N} \not\models f(\sigma)[a]$. Since $(\mathfrak{N}, a)$ and $(\mathfrak{N}', 1)$ are quasi-injective bisimilar, $f(\sigma)$ is not invariant under quasi-injective bisimulations. □

This theorem can be easily extended to the case of any bisimulation mentioned in the next section.

§5. Characterization theorem for each $\mathscr{H}_S$. As we said in the Introduction, characterization theorems have been proved in the case of $\mathscr{H}_{\{@\}}$ (see ten Cate, 2005) and of $\mathscr{H}_{\{\downarrow,@\}}$ (see Areces *et al.*, 2001). Each uses another kind of bisimulation, respectively $\mathscr{H}(@)$-bisimulation and ω-bisimulation.

There are four subsets of $\mathscr{O}$ that still need to be studied: $\{\exists\}$, $\{\exists, \downarrow\}$, $\{\exists, @\}$, and $\{\exists, \downarrow, @\}$.

First we can see that when the $\exists$ is available, $\downarrow$ cannot express anything new. Indeed, we can translate any $\downarrow x\varphi$ by $\exists x(x \wedge \varphi)$.

So we have to study two sets: $\{\exists\}$ and $\{\exists, @\}$. The first one is quite complex and would need further research. We leave it as an open question to prove a bisimulation characterization theorem for $\mathscr{H}_{\{\exists\}}$. The expressivity of the second one is the same as that of first-order logic. One direction is obvious, it has to be a fragment of first-order logic. For the other direction, let F be the recursive function defined by:

$$\begin{array}{lcll}
F(x) & = & x & \\
F(I_j) & = & i_j & \\
F(\bot) & = & \bot & \\
F(P_j(t)) & = & @_{F(t)} p_j & \text{for a term } t \\
F(\neg\varphi) & = & \neg F(\varphi) & \\
F(\varphi_1 \vee \varphi_2) & = & F(\varphi_1) \vee F(\varphi_2) & \\
F(\exists x\varphi) & = & \exists x F(\varphi) & \\
F(tRu) & = & @_{F(t)} \Diamond F(u) & \text{for terms } t, u \\
F(t = u) & = & @_{F(t)} F(u) & \text{for terms } t, u
\end{array}$$

and let the standard back translation of a first-order formula $\varphi(x)$ with one free variable x be:

$$SBT(\varphi(x)) = \exists x(x \wedge F(\varphi(x))).$$

It is quite clear that a first-order formula with one free variable and its standard back translation have the same truth value.

In Table 1 there is a summary of all the subsets of $\mathscr{O}$, and the conditions of application of the bisimulation characterization theorem.

Table 1. *The sets of hybrid operators and the conditions*

Set of hybrid operators	Condition of application (invariance by ...)
$\emptyset$	Bisimulation
$\{\downarrow\}$ (without nominals)	Quasi-injective bisimulation
$\{@\}$	$\mathscr{H}(@)$-bisimulation
$\{\exists\}$	Unknown
$\{\downarrow, \exists\}$	Unknown, but the same as for $\{\exists\}$
$\{\downarrow, @\}$	ω-bisimulation
$\{@, \exists\}$	First-order logic, so no condition
$\mathscr{O}$	First-order logic, so no condition

§6. Conclusion.

In this paper we tried to characterize the domain of expressivity of $\mathscr{H}_{\{\downarrow\}}$-formulas within the world of first-order logic. We were successful in using the quasi-injective bisimulation as defined in Blackburn & Seligman (1998) so as to give a new characterization theorem of $\mathscr{H}_{\{\downarrow\}}$ formulas, even though it is not complete: we only handle formulas without nominals. The general case with nominals remains open.

Finally, we adapted easily the usual proof to show the undecidability of invariance under bisimulation of a first-order formula to the case of quasi-injective bisimulation.

§7. Acknowledgment. The authors would like to thank Louis Paternault and the referee for their reading, comments, and suggestions.

BIBLIOGRAPHY

Areces, C., Blackburn, P., & Marx, M. (2001). Hybrid logics: Characterization, interpolation, and complexity. *Journal of Symbolic Logic*, **66**, 977–1010.

Blackburn, P., de Rijke, M., & Venema, Y. (2001). *Modal logic*. Tracts in Theoretical Computer Science. Cambridge, UK: Cambridge University Press.

Blackburn, P., & Seligman, J. (1998). What are hybrid languages? In Kracht, M., de Rijke, M., Wansing, H., and Zakharyaschev, M., editors. *Advances in Modal Logic*. Stanford, CA: CSLI Publications, pp. 41–62.

Franceschet, M., & de Rijke, M. (2006). Model checking hybrid logics (with an application to semistructured data). *Journal of Applied Logic*, **4**(3), 279–304.

Hodges, W. (1993). *Model Theory*, Volume 42 of *Encyclopedia of Mathematics and Its Applications*. Cambridge, UK: Cambridge University Press.

Karp, C. (1963). Finite-quantifier equivalence. In Addison, J., Henkin, L., and Tarski, A., editors. *The Theory of Models*. Amsterdam: North-Holland Publishing, pp. 407–412.

ten Cate, B. (2005). Model theory for extended modal languages. PhD Thesis, University of Amsterdam. ILLC Dissertation Series DS-2005-01.

van Benthem, J. (1985). *Modal Logic and Classical Logic*. Napoli, Italy: Bibliopolis.

van Benthem, J. (1996). *Exploring Logical Dynamics*. Studies in Logic, Language and Information. Stanford, CA: CSLI Publications & FoLLI.

DEPARTMENT OF COMPUTING
IMPERIAL COLLEGE
LONDON SW7 2AZ
UK
E-mail: imh@doc.ic.ac.uk, hicham.tahiri@centraliens.net