

# Completeness Results for Memory Logics

Carlos Areces\*    Santiago Figueira†    Sergio Mera†

## Abstract

Memory logics are a family of modal logics in which standard relational structures are augmented with data structures and additional operations to modify and query these structures. In this paper we present sound and complete axiomatizations for some members of this family. We analyze the use of nominals to achieve completeness, and present one example in which they can be avoided.

## 1 Modal Logics with Memory

Many attempts have been made in recent years to increase modal logic expressivity by adding some notion of *state* to standard relational structures. This is a natural need, since modal logics are used in many different scenarios as tools for modeling behavior.

One example of how this can be achieved comes from epistemic logic with dynamic operators, which allow to express the evolution of knowledge by knowledge-changing actions. Such logics are often called Dynamic Epistemic Logics (DEL) [16], and a large number of DELs has been proposed [9, 13, 14, 15]. These logics differ considerably in expressive power, but the common idea is to express knowledge evolution by accessing and changing the model structure through logic operators. An alternative approach comes from the software verification community, and the use of temporal logic with explicit global clocks which are accessed and controlled through logic operators. The logic XCTL of Harel et al. [10] is an example of this approach. Another example, also from the software verification community, is the extension of temporal logic with a concrete domain (e.g., the natural numbers with some operations like addition, comparison, etc.) which is accessed via the so-called freeze operator [1, 11]. In the extended language, we can model qualitative properties using the temporal operators, and concrete properties –such as weight, temperature, etc.– using the new machinery. To cite yet another example, concrete domains have also been added to description logics, with much the same aims [12].

We would like to take a step back, and analyze some of the basic intuitions that most of the formal languages mentioned above have in common. We want to try to investigate the idea of adding an *explicit state* to a model, and being able to access (and modify) it

---

\*INRIA Nancy Grand Est, France

†Departamento de Computación, FCEyN, UBA, Argentina

via logical operators. And we would like to take this idea in its simplest form, in order to be able to understand it in detail.

We can take a standard relational structure and complement it with a *data structure*, that will keep the state information we want to model. We will also add to the logical language a collection of operations to modify and access the data structure. Formally, given a relational structure  $\langle D, (R_r)_{r \in \mathbf{Rel}}, L \rangle$  where  $D$  is a non empty domain,  $(R_r)_{r \in \mathbf{Rel}}$  is a set of relations over  $D$ , and  $L : \mathbf{Atom} \rightarrow 2^D$  is a labeling function that assigns atomic properties to elements of  $D$ , we extend the structure with a set  $S \subseteq D$ . We can think of  $S$  as a set of states that are ‘known’ to us, and it will represent our current ‘memory’. Even in this simple setting we can define the following operators:

$$\begin{aligned} \langle D, (R_r)_{r \in \mathbf{Rel}}, L, S \rangle, w \models \textcircled{\mathbf{r}}\varphi & \text{ iff } \langle D, (R_r)_{r \in \mathbf{Rel}}, L, S \cup \{w\} \rangle, w \models \varphi \\ \langle D, (R_r)_{r \in \mathbf{Rel}}, L, S \rangle, w \models \textcircled{\mathbf{k}} & \text{ iff } w \in S. \end{aligned}$$

As it is clear from the definition above, the ‘remember’ operator  $\textcircled{\mathbf{r}}$  (a unary modality) just marks the current state as being ‘known’ or ‘already visited’, by storing it in our ‘memory’  $S$ . On the other hand, the zero-ary operator  $\textcircled{\mathbf{k}}$  (for ‘known’) queries  $S$  to check if the current state has already been visited. Notice that the extension of  $S$  is dynamic and it can vary during the evaluation of a formula; while the ‘concrete’ operation we can apply to  $S$  is simple membership.

Other operators can naturally be added, for example:

$$\langle D, (R_r)_{r \in \mathbf{Rel}}, L, S \rangle, w \models \textcircled{\mathbf{e}}\varphi \quad \text{iff} \quad \langle D, (R_r)_{r \in \mathbf{Rel}}, L, \emptyset \rangle, w \models \varphi.$$

I.e., we can use the erase operator  $\textcircled{\mathbf{e}}$  to completely wipe out the memory  $S$ . We have introduced this family of logics, that we called *memory logics*, and investigated its expressive power in [2, 3, 4].

The language we have just described is very flexible, and it can be used to easily characterize model properties. For example if all states in the domain of a model  $\mathcal{M}$  satisfy the formula  $\textcircled{\mathbf{e}}\textcircled{\mathbf{r}}\langle r \rangle \textcircled{\mathbf{k}}$  then the relation  $R_r$  is reflexive (we wipe out the memory, memorize the current point of evaluation and verify that it is accessible). Similarly, if they satisfy  $\textcircled{\mathbf{e}}\textcircled{\mathbf{r}}[r]\langle r \rangle \textcircled{\mathbf{k}}$  then  $R_r$  is symmetric (erase memory, memorise the current point of evaluation and verify that all successors can reach back the memorized point). Actually, using  $\textcircled{\mathbf{e}}$ ,  $\textcircled{\mathbf{r}}$  and  $\textcircled{\mathbf{k}}$  we can express properties similarly as how it is done in the hybrid language  $\mathcal{HL}(\downarrow)$  [5]. But in [3, 4] we have shown that the modal language extended with  $\textcircled{\mathbf{e}}$ ,  $\textcircled{\mathbf{r}}$  and  $\textcircled{\mathbf{k}}$  is strictly less expressive than  $\mathcal{HL}(\downarrow)$ .

In this article we are interested in providing complete axiomatizations for these logics. With this aim in mind, we will include in the language also nominals and the hybrid  $\textcircled{\mathbf{e}}$  operator (see [5, 6] for details on hybrid logics). As discussed in [8], the hybrid machinery can be used to prove general completeness results, and to axiomatize logics which are otherwise difficult to characterize.

The rest of the paper is organized as follows. In the next section we formally introduce the different logics we will investigate. In Sect. 3 we present a sound and complete axiom-

atization for  $\mathcal{HL}(@, \textcircled{r}, \textcircled{k})^1$ , the basic modal logic extended with nominals, @, and the  $\textcircled{r}$  and  $\textcircled{k}$  memory operators. Our axiomatization crucially involves the use of nominals. As discussed in [3], the  $\textcircled{r}$  and  $\textcircled{k}$  operators are very expressive, and even when added to the basic modal language, they already give rise to an undecidable logic. In [3] we introduced a logic including  $\textcircled{r}$  and  $\textcircled{k}$  with additional constraints on how the modal and the memory operators interact. In that work we showed that although this logic is strictly more expressive than  $\mathcal{ML}$ , it turns out to be decidable. We will show a sound and complete axiomatization for this logic in Sect. 4. Finally in Sect. 5 we discuss completeness for a language including the  $\textcircled{e}$  operator. We conclude in Sect. 6 with some final remarks.

## 2 Syntax and Semantics of Memory Logics

In this section we formally introduce the languages mentioned above, together with some basic notation and notions related to completeness.

**Definition 1** (Syntax). *Let  $\text{Prop} = \{p_1, p_2, \dots\}$  (the propositional symbols),  $\text{Nom} = \{n_1, n_2, \dots\}$  (the nominal symbols) and  $\text{Rel} = \{r_1, r_2, \dots\}$  (the relational symbols) be pairwise disjoint, countable infinite sets. Let  $\text{Atom} = \text{Prop} \cup \text{Nom}$ . The set **Forms** of formulas in the signature  $\langle \text{Prop}, \text{Nom}, \text{Rel} \rangle$  is defined as:*

$$\text{Forms} ::= \top \mid p \mid i \mid \textcircled{k} \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \langle r \rangle\varphi \mid @_i\varphi \mid \textcircled{r}\varphi \mid \textcircled{e}\varphi,$$

where  $p \in \text{Prop}$ ,  $i \in \text{Nom}$ ,  $r \in \text{Rel}$  and  $\varphi, \varphi_1, \varphi_2 \in \text{Forms}$ . We use  $[r]\varphi$  as a shorthand for  $\neg\langle r \rangle\neg\varphi$ .

**Definition 2** (Semantics). *Given a signature  $\mathcal{S} = \langle \text{Prop}, \text{Nom}, \text{Rel} \rangle$ , a model for  $\mathcal{S}$  is a tuple  $\langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$ , satisfying the following conditions: (i)  $D \neq \emptyset$ ; (ii) each  $R_r$  is a binary relation on  $D$ ; (iii)  $L : \text{Atom} \rightarrow 2^D$  is a labeling function such that  $L(n)$  is a singleton whenever  $n \in \text{Nom}$ ; and (iv)  $S \subseteq D$ .*

*Given the model  $\mathcal{M} = \langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$  and  $w \in D$ , the semantics for the different operators is defined as follows:*

$$\begin{array}{lll} \mathcal{M}, w \models \top & \text{always} & \\ \mathcal{M}, w \models p & \text{iff} & w \in L(p) \quad p \in \text{Atom} \\ \mathcal{M}, w \models \neg\varphi & \text{iff} & \mathcal{M}, w \not\models \varphi \\ \mathcal{M}, w \models \varphi \wedge \psi & \text{iff} & \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi \\ \mathcal{M}, w \models \langle r \rangle\varphi & \text{iff} & \text{there is } w' \text{ such that } R_r(w, w') \text{ and } \mathcal{M}, w' \models \varphi \\ \mathcal{M}, w \models @_i\varphi & \text{iff} & \mathcal{M}, v \models \varphi \text{ where } L(i) = \{v\} \\ \mathcal{M}, w \models \textcircled{r}\varphi & \text{iff} & \langle D, (R_r)_{r \in \text{Rel}}, L, S \cup \{w\} \rangle, w \models \varphi \\ \mathcal{M}, w \models \textcircled{k} & \text{iff} & w \in S \\ \mathcal{M}, w \models \textcircled{e}\varphi & \text{iff} & \langle D, (R_r)_{r \in \text{Rel}}, L, \emptyset \rangle, w \models \varphi. \end{array}$$

---

<sup>1</sup>Our notational convention for naming logics is the following. We will call  $\mathcal{ML}$  the basic modal logic, with the standard modal operators. We will use  $\mathcal{HL}$  for the modal language extended with only nominals. And we will then list the additional operators included in the language. For example  $\mathcal{HL}(@, \textcircled{r}, \textcircled{k}, \textcircled{e})$  is the modal language extended with nominals, and the @,  $\textcircled{r}$ ,  $\textcircled{k}$  and  $\textcircled{e}$  operators.

Given a model  $\mathcal{M} = \langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$  and states  $w_1, \dots, w_n \in D$ , we will write  $\mathcal{M}[w_1, \dots, w_n]$  for the model  $\langle D, (R_r)_{r \in \text{Rel}}, L, S \cup \{w_1, \dots, w_n\} \rangle$ .

Given a model  $\mathcal{M}$ , we say that  $\varphi$  is valid on  $\mathcal{M}$  and write  $\mathcal{M} \models \varphi$  if for all states  $w$  in the domain of  $\mathcal{M}$  we have that  $\mathcal{M}, w \models \varphi$ .

**Definition 3** (Satisfiability, Validity, Completeness). *Let  $\mathcal{C}$  be a class of models. We say that  $\varphi$  is satisfiable in  $\mathcal{C}$  if there is a model  $\mathcal{M} \in \mathcal{C}$  and a state  $w$  in the domain of  $\mathcal{M}$  such that  $\mathcal{M}, w \models \varphi$ . We say that  $\varphi$  is valid in  $\mathcal{C}$  if  $\neg\varphi$  is not satisfiable in  $\mathcal{C}$ . The notions of satisfiability and validity can be extended to set of formulas in the usual way. For example, we say that a set of formulas  $\Gamma$  is satisfiable in a class of models  $\mathcal{C}$  if there is a model  $\mathcal{M} \in \mathcal{C}$  and a state  $w$  in the domain of  $\mathcal{M}$  such that for all formulas  $\varphi \in \Gamma$  we have  $\mathcal{M}, w \models \varphi$ . We will note  $T(\mathcal{C})$  the set of all valid formulas in  $\mathcal{C}$ .*

Given an axiomatization  $\mathcal{A}$ , a formula  $\varphi$  is a theorem of  $\mathcal{A}$  if it is an axiom in  $\mathcal{A}$ , or it can be obtained by a finite number of applications of inference rules in  $\mathcal{A}$  from axioms of  $\mathcal{A}$ . We write  $T(\mathcal{A})$  for the set of all theorems in  $\mathcal{A}$ .

We say that a formula  $\varphi$  is consistent with respect to an axiomatization  $\mathcal{A}$  (or  $\mathcal{A}$ -consistent) if  $\neg\varphi$  is not a theorem of  $\mathcal{A}$ . The notion of consistency can be extended to a set of formulas  $\Gamma$  by requiring that for no finite subset  $\Gamma^f$ , the formula  $\bigwedge \Gamma^f \rightarrow \neg\top$  be a theorem of  $\mathcal{A}$ .

Given an axiomatization  $\mathcal{A}$  and a class of models  $\mathcal{C}$  we say that  $\mathcal{A}$  is sound for  $\mathcal{C}$  if  $T(\mathcal{A}) \subseteq T(\mathcal{C})$ , and that it is complete for  $\mathcal{C}$  if  $T(\mathcal{C}) \subseteq T(\mathcal{A})$ . Completeness can be equivalently defined in terms of consistency and satisfiability:  $\mathcal{A}$  is complete for  $\mathcal{C}$  if every formula consistent in  $\mathcal{A}$  is satisfiable in  $\mathcal{C}$ .

Finally, we say that an axiomatization  $\mathcal{A}$  is strongly complete with respect to  $\mathcal{C}$ , if every  $\mathcal{A}$ -consistent set of formulas is satisfiable in  $\mathcal{C}$ .

In this article we will present a number of axiomatizations and prove them (strongly) complete with respect to different classes of models. The different logical languages involved will be defined in terms of the operators introduced in Definitions 1 and 2; and we will be interested mainly in the class of all models, and the class  $\{\langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle \mid S = \emptyset\}$  of models with no previously ‘remembered’ states. This last class is a natural choice: in the absence of the  $\textcircled{e}$  operator, evaluating formulas on such models provides additional expressivity, and the intuitive meaning of the *remember* and *known* operators are naturally captured. For example the formula  $\textcircled{r}\langle r \rangle \textcircled{\text{K}}$  characterizes reflexivity of  $R_r$  over this class (that is, let  $\mathcal{M} = \langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$  be an arbitrary model, except that  $S = \emptyset$ , then  $\mathcal{M} \models \textcircled{r}\langle r \rangle \textcircled{\text{K}}$  if and only if  $R_r$  is reflexive). This no longer holds when  $S$  is arbitrary. See [3] for further details.

As we mentioned in the introduction, we will also be interested in a logic in which the behavior of the *remember* operator is highly coupled with the modal transitions to ensure decidability. In this logic, every time we make a modal step, we are constrained to remember the current state. We change the semantic definition of  $\langle r \rangle$  to be:

$$\langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle, w \models \langle r \rangle \varphi \quad \text{iff} \quad \exists w' \in D, R_r(w, w') \text{ and} \\ \langle D, (R_r)_{r \in \text{Rel}}, L, S \cup \{w\} \rangle, w' \models \varphi$$

<b>Axioms:</b>			
<i>CT</i>	All classical tautologies	<i>Intro</i>	$\vdash (i \wedge p) \rightarrow @_i p$
$K_{@}$	$\vdash @_i(p \rightarrow q) \rightarrow @_i p \rightarrow @_i q$	<i>Self-dual<sub>@</sub></i>	$\vdash @_i p \leftrightarrow \neg @_i \neg p$
$K_{[r]}$	$\vdash [r](p \rightarrow q) \rightarrow ([r]p \rightarrow [r]q)$	<i>Ref</i>	$\vdash @_i i$
<i>Sym</i>	$\vdash @_i j \leftrightarrow @_j i$	<i>Nom</i>	$\vdash (@_i j \wedge @_j p) \rightarrow @_i p$
<i>Agree</i>	$\vdash @_j @_i p \leftrightarrow @_i p$	<i>Back</i>	$\vdash \langle r \rangle @_i p \rightarrow @_i p$
<i>Rem</i>	$\vdash @_i(\mathbb{R}\varphi \leftrightarrow \varphi[\mathbb{K}/(\mathbb{K} \vee i)])$		
<b>Rules:</b>			
<i>MP</i>	If $\vdash \varphi$ and $\vdash \varphi \rightarrow \psi$ then $\vdash \psi$	$Gen_{[r]}$	If $\vdash \varphi$ then $\vdash [r]\varphi$
<i>Name</i>	$\vdash j \rightarrow \varphi$ then $\vdash \varphi$ ( $j$ not in $\varphi$ )	$Gen_{@}$	If $\vdash \varphi$ then $\vdash @_i \varphi$
<i>Paste</i>	If $\vdash (@_i \langle r \rangle j \wedge @_j \varphi) \rightarrow \psi$ then $\vdash (@_i \langle r \rangle \varphi) \rightarrow \psi$ ( $j \neq i$ and $j$ is not in $\varphi$ or $\psi$ )		
<i>SortedSub<sub>1</sub></i>	If $\vdash \varphi$ then $\vdash \varphi[p/\psi]$ for any $p \in \mathbf{Prop}$		
<i>SortedSub<sub>2</sub></i>	If $\vdash \varphi$ then $\vdash \varphi[i/j]$ for any $i, j \in \mathbf{Nom}$		
The expression $\varphi[a/b]$ is the result of uniformly replacing all occurrences of $a$ in $\varphi$ by $b$ .			

Figure 1: Axiomatization for  $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$ .

We call this logic  $\mathcal{ML}^-$  ( $\mathcal{HL}^-$  for the hybrid case). As we proved in [4],  $\mathcal{ML}^-(\mathbb{R}, \mathbb{K})$  is decidable and strictly more expressive than  $\mathcal{ML}$ .

### 3 Completeness for $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$

This section is devoted to prove a completeness result for  $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$ . Our axiomatization is shown in Fig. 1. It is an extension of the axiomatization for  $\mathcal{HL}(@)$  presented in [7].

The axiom characterizing the behavior of the memory operator is *Rem*. To show soundness of the axiomatization, we only have to look at this new axiom. Intuitively, the axiom says that, when standing in a state named by  $i$ , the act of remembering the current state is equivalent to increase the extension of  $\mathbb{K}$  with  $i$  throughout the formula. Formally:

**Lemma 4.** *Let  $\mathcal{M}$  be a model and  $w \in \mathcal{M}$  such that  $\mathcal{M}, w \models i$ . Then, for all  $v \in \mathcal{M}$ ,  $\mathcal{M}[w], v \models \varphi$  iff  $\mathcal{M}, v \models \varphi[\mathbb{K}/(\mathbb{K} \vee i)]$ .*

*Proof.* By induction on  $\varphi$ . For the base case, if  $\varphi$  is a proposition symbol or a nominal, then since  $\varphi = \varphi[\mathbb{K}/(\mathbb{K} \vee i)]$  we have  $\mathcal{M}[w], v \models \varphi$  iff  $\mathcal{M}, v \models \varphi$ . For the  $\mathbb{K}$  case we have to prove  $\mathcal{M}[w], v \models \mathbb{K}$  iff  $\mathcal{M}, v \models \mathbb{K} \vee i$ .

$\Rightarrow$ ) Assume that  $\mathcal{M}[w], v \models \mathbb{K}$ . If  $v = w$ , then  $\mathcal{M}, v \models i$ , and therefore  $\mathcal{M}, v \models \mathbb{K} \vee i$ . If  $v \neq w$ , then  $\mathcal{M}, v \models \mathbb{K}$ , and hence  $\mathcal{M}, v \models \mathbb{K} \vee i$ .

$\Leftarrow$ ) Let's assume that  $\mathcal{M}, v \models \mathbb{K} \vee i$ . If  $v = w$ , then  $\mathcal{M}[w], v \models \mathbb{K}$ . On the other hand, if  $v \neq w$ , then we know that  $\mathcal{M}[w], v \models \neg i$ , and therefore  $\mathcal{M}, v \models \mathbb{K}$ . We conclude  $\mathcal{M}[w], v \models \mathbb{K}$ .

The conjunction, negation, diamond,  $@$  and remember cases are straightforward, using the inductive hypothesis and the fact that the replacement operation  $[\mathbb{K}/(\mathbb{K} \vee i)]$  distributes

over  $\wedge$ ,  $\neg$ ,  $\langle r \rangle$ ,  $@$  and  $\textcircled{r}$ . □

**Corollary 5.** *Rem is sound over the class of all models.*

*Proof.* Take an arbitrary model  $\mathcal{M}$  and let  $w \in \mathcal{M}$  be such that  $\mathcal{M}, w \models i$ . By definition  $\mathcal{M}, v \models @_i \textcircled{r} \varphi$  iff  $\mathcal{M}[w], w \models \varphi$ . Applying the previous lemma, this happens iff  $\mathcal{M}, w \models \varphi[\mathbb{K}/(\mathbb{K} \vee i)]$  iff (by definition)  $\mathcal{M}, v \models @_i \varphi[\mathbb{K}/(\mathbb{K} \vee i)]$ . □

It is worth noting that having nominals in the language is a key feature to describe the  $\textcircled{r}/\mathbb{K}$  interaction with modal operators, and the *Rem* axiom strongly uses this feature. The possibility to identify with a nominal the state in which a remember operation is taking place allows us to fully describe the behavior of this interaction.

We now turn to completeness. We will build a Henkin model using named maximal consistent sets (MCSs) for an arbitrary consistent set (see [7] for further details).

**Definition 6.** *An MCS is named if and only if it contains a nominal. We call any nominal belonging to an MCS a name for that MCS. Also, if  $\Gamma$  is an MCS and  $i$  is a nominal, then we call  $\{\varphi \mid @_i \varphi \in \Gamma\}$  a named set yielded by  $\Gamma$ . Furthermore we say that a model is named if every state in the model is the denotation of some nominal (for all  $w \in D$  there is some nominal  $i$  such that  $L(i) = \{w\}$ ).*

The idea behind the construction presented in [7] is that we can extract all the information we need to build a named canonical model from a single MCS. We start by noting that hidden inside any MCS there is a collection of named MCSs with a number of relevant properties:

**Lemma 7.** *Let  $\Gamma$  be an MCS. For every nominal  $i$ , let  $\Delta_i$  be  $\{\varphi \mid @_i \varphi \in \Gamma\}$ . Then, (i) for every nominal  $i$ ,  $\Delta_i$  is an MCS that contains  $i$ ; (ii) for all nominals  $i$  and  $j$ , if  $i \in \Delta_j$ , then  $\Delta_i = \Delta_j$ ; (iii) for all nominals  $i$  and  $j$ ,  $@_i \varphi \in \Delta_j$  iff  $@_i \varphi \in \Gamma$ ; and (iv) if  $i$  is a name for  $\Gamma$  then  $\Gamma = \Delta_i$ .*

*Proof.* We only sketch the proof, the full details can be found in [7]. Claim (i) can be proved using *Ref* (to guarantee that  $i \in \Delta_i$ ), *Gen<sub>@</sub>* and *Self-dual<sub>@</sub>* (to prove that  $\Delta_i$  is an MCS). Claim (ii) is proved using *Sym* and *Nom*, Claim (iii) follows by *Agree*. And Claim (iv) is obtained by *Intro* and *Self-dual<sub>@</sub>*. □

Given a consistent set of formulas  $\Sigma$ , we can always expand it to an MCS  $\Sigma^+$  using the standard Lindenbaum's Lemma. The problem is that nothing guarantees that this MCS will be named. In addition, as we want to extract named MCSs from named sets yielded by  $\Sigma^+$ , we have to ensure that there are enough named MCSs to use as existential witnesses during the construction of the Henkin model. Here is where the *Name* and *Paste* rules are useful. Expanding the language with new nominals, the *Name* rule is going to solve our first problem, and the *Paste* rule solves the second. We call an MCS  $\Gamma$  *pasted* iff  $@_i \langle r \rangle \varphi \in \Gamma$  implies that for some nominal  $j$ ,  $@_i \langle r \rangle j \wedge @_j \varphi \in \Gamma$ . *Name* and *Paste* guarantee that any consistent set of formulas can be extended to a named and pasted MCS.

**Lemma 8** (Extended Lindenbaum Lemma). *Let  $\mathcal{S} = \langle \text{Prop}, \text{Nom}, \text{Rel} \rangle$  be a signature, let  $\text{Nom}'$  be a countably infinite collection of nominals disjoint from  $\text{Nom}$ , and let  $\mathcal{S}'$  be the signature obtained by extending  $\mathcal{S}$  with  $\text{Nom}'$ . Then every  $\mathcal{HL}(\text{@}, \text{\textcircled{r}}, \text{\textcircled{k}})$ -consistent set of formulas in  $\mathcal{S}'$  can be extended to a named and pasted MCS in  $\mathcal{S}'$ .*

*Proof.* Full details can be found in [7]. The proof follows the standard Lindenbaum's construction with the following modifications. Take a consistent set of formulas  $\Sigma$ , and name it by adding a new nominal  $k$  (use *Name* to prove consistency). Using an enumeration of all the formulas, we expand  $\Sigma$  step-by-step with a formula that is consistent with the expanded set at each point. Because we want the final MCS to be pasted, at the  $(m+1)$ -th step, when we are considering  $\Sigma^m$  and the formula  $\varphi_{m+1}$ , if  $\Sigma^m \cup \{\varphi_{m+1}\}$  is inconsistent, we set  $\Sigma^{m+1} = \Sigma^m$ . Else, if  $\varphi_{m+1}$  has the form  $\text{@}_i \langle r \rangle \varphi$ , we set  $\Sigma^{m+1} = \Sigma^m \cup \{\varphi_{m+1}\} \cup \{\text{@}_i \langle r \rangle j \wedge \text{@}_j \langle r \rangle \varphi\}$ , where  $j$  is new (relying on the *Paste* rule for consistency). If  $\varphi_{m+1}$  does not have the form  $\text{@}_i \langle r \rangle \varphi$ , we set  $\Sigma^{m+1} = \Sigma^m \cup \{\varphi_{m+1}\}$  as usual. Finally, we take the infinite union of all the  $\Sigma^i$ .  $\square$

Now we can define the model we need, using the named sets yielded by a named and pasted MCS.

**Definition 9.** *Let  $\Gamma$  be a named and pasted MCS. The named model yielded by  $\Gamma$  is  $\mathcal{M}^\Gamma = (D^\Gamma, (R_r^\Gamma)_{r \in \text{Rel}}, L^\Gamma, S^\Gamma)$ . Here  $D^\Gamma$  is the set of all named sets yielded by  $\Gamma$ ,  $R_r^\Gamma(u, v)$  holds iff for all formulas  $\varphi$ ,  $\varphi \in v$  implies  $\langle r \rangle \varphi \in u$ ,  $L^\Gamma(a) = \{w \in W^\Gamma \mid a \in w\}$  for any atom  $a$ , and  $S^\Gamma = \{w \mid \text{\textcircled{k}} \in w\}$ .*

Note that  $\mathcal{M}^\Gamma$  is a well defined model, since by items (i) and (ii) of Lemma 7,  $L^\Gamma$  assigns to every nominal a singleton subset of  $D^\Gamma$ . Using the fact that  $\Gamma$  is named and pasted, we can prove the following Existence Lemma

**Lemma 10** (Existence Lemma [7]). *Let  $\Gamma$  be a named and pasted MCS, and let  $\mathcal{M} = \langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$  be the named model yielded by  $\Gamma$ . Suppose  $u \in \mathcal{M}$  and  $\langle r \rangle \varphi \in u$ . Then there is a  $v \in \mathcal{M}$  such that  $R_r(u, v)$  and  $\varphi \in v$*

Now we are ready to prove the Truth Lemma that will lead us to the desired completeness result. Before that, to treat the  $\text{\textcircled{r}}$  case properly, we have to redefine the complexity of the formulas, to be able to handle the substitutions made by the *Rem* axiom.

**Definition 11.** *We define the complexity of a formula as  $\text{comp}(\varphi) = 2(k+1)(r+1)(d+1) + v$ , where  $k$ ,  $r$  and  $d$  are the number of occurrences of  $\text{\textcircled{k}}$ ,  $\text{\textcircled{r}}$  and  $\langle r \rangle$  respectively, and  $v$  is the number of occurrences of all the other possible symbols.*

Note that with this definition,  $\text{comp}(\text{\textcircled{r}}\varphi) > \text{comp}(\varphi[\text{\textcircled{k}}/(\text{\textcircled{k}} \vee i)])$ .

**Lemma 12** (Truth Lemma). *Let  $\mathcal{M} = \langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$  be the named model yielded by a named and pasted MCS, and let  $u \in D$ . Then, for all formulas  $\varphi$ ,  $\varphi \in u$  iff  $\mathcal{M}, u \models \varphi$ .*

*Proof.* By Induction on the structure of  $\varphi$ . The atomic, boolean and modal cases are obvious (the Existence Lemma is used for the modal case, and the  $\mathbb{K}$  case follows directly from the definition of  $S^\Gamma$ ). We analyze the satisfaction operators. Suppose  $\mathcal{M}, u \models @_i\psi$ . This happens iff  $\mathcal{M}, \Delta_i \models \psi$  (by items (i) and (ii) of Lemma 7,  $\Delta_i$  is the only MCS containing  $i$ , and hence, by the atomic case of the present lemma, the only state in  $\mathcal{M}$  where  $i$  is true) iff  $\psi \in \Delta_i$  (by inductive hypothesis) iff  $@_i\psi \in \Delta_i$  (using the fact that  $i \in \Delta_i$  together with *Intro* for the left-to-right direction and *Intro* and *Self-dual*<sub>@</sub> for the right-to-left direction) iff  $@_i\psi \in u$  (by *Agree*).

To finish let's analyze the case for  $\mathbb{R}$ . Given  $u \in \mathcal{M}$ , we know that for some nominal  $i$  we have  $u = \Delta_i$ , so by definition,  $\mathcal{M}, u \models i$  and  $i \in u$ . Suppose  $\mathcal{M}, u \models \mathbb{R}\psi$ . This happens iff  $\mathcal{M}, u \models @_i\mathbb{R}\psi$  (because  $\mathcal{M}, u \models i$ ) iff  $\mathcal{M}, u \models @_i\psi[\mathbb{K}/(\mathbb{K} \vee i)]$  (by Corollary 5) iff  $\mathcal{M}, u \models \psi[\mathbb{K}/(\mathbb{K} \vee i)]$  (again because  $\mathcal{M}, u \models i$ ) iff  $\psi[\mathbb{K}/(\mathbb{K} \vee i)] \in u$  (by inductive hypothesis) iff  $@_i\psi[\mathbb{K}/(\mathbb{K} \vee i)] \in u$  (because  $i \in u$ , using *Intro* for the left-to-right direction, and *Self-dual*<sub>@</sub> and *Intro* for the right-to-left direction) iff  $@_i\mathbb{R}\psi \in u$  (by the *Rem* axiom) iff  $\mathbb{R}\psi \in u$  (because  $i \in u$ , applying again *Intro* and *Self-dual*<sub>@</sub>).  $\square$

**Theorem 13** (Completeness for  $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$ ). *Every MCS of formulas in  $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$  is satisfiable in a countable named model.*

*Proof.* Let  $\Sigma$  be a consistent set of formulas from  $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$ . We use the Extended Lindenbaum Lemma to expand it to a named and pasted set  $\Sigma^+$  in an extended countable language. Let  $\mathcal{M}$  be the named model yielded by  $\Sigma^+$ . By item (iv) of Lemma 7, because  $\Sigma^+$  is named,  $\Sigma^+$  is an element in the domain of  $\mathcal{M}$ . By the Truth Lemma,  $\mathcal{M}, \Sigma^+ \models \Sigma$ . The model is countable because each state is named by some nominal in the extended language, and there are only countably many of these.  $\square$

This establishes strong completeness as desired. But in fact, we have done more. Because our Henkin model is *named*, we can prove a more general result.

**Definition 14.** *If a formula  $\varphi$  contains no propositional symbols (that is, its atoms are nominals or  $\mathbb{K}$ ), we say that  $\varphi$  is  $\mathbb{K}$ -pure. Furthermore, if  $\varphi$  is a  $\mathbb{K}$ -pure formula, we say that  $\psi$  is a  $\mathbb{K}$ -pure instance of  $\varphi$  if  $\psi$  is obtained from  $\varphi$  by uniformly substituting nominals for nominals. A formula  $\varphi$  is pure if its atomic subformulas are only nominals.*

The axiomatization we presented in Fig. 1 for  $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$  has the following property: for any set of pure formulas  $\Pi$ , if  $P$  is the logic obtained by adding the formulas in  $\Pi$  as axioms, then  $P$  is complete with respect to the class defined by  $\Pi$ .<sup>2</sup> This result can be extended to  $\mathbb{K}$ -pure formulas for the case of  $\mathcal{HL}_\emptyset(@, \mathbb{R}, \mathbb{K})$ , the logic obtained over the class  $\{\langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle \mid S = \emptyset\}$  of models with no previously remembered states.

We first state a property that will help us achieve the completeness result for pure axioms.

**Lemma 15.** *Let  $\mathcal{M} = \langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$  be a named model.*

<sup>2</sup>These general completeness results are standard in hybrid logics. See [8] for further details.

1. Let  $\varphi$  be a pure formula, and suppose that for all pure instances  $\psi$  of  $\varphi$ ,  $\mathcal{M} \models \psi$ . Then for any  $L'$  and  $S'$ ,  $\langle D, (R_r)_{r \in \text{Rel}}, L', S' \rangle \models \varphi$ .
2. Let  $S = \emptyset$ ,  $\varphi$  be a  $\mathbb{K}$ -pure formula, and suppose that for all  $\mathbb{K}$ -pure instances  $\psi$  of  $\varphi$ ,  $\mathcal{M} \models \psi$ . Then for any  $L'$ ,  $\langle D, (R_r)_{r \in \text{Rel}}, L', S \rangle \models \varphi$ .

*Proof.* We only discuss item 2. Suppose that the hypothesis hold, but for some labeling  $L'$ ,  $\langle D, (R_r)_{r \in \text{Rel}}, L', \emptyset \rangle \not\models \varphi$ . We can take  $\rho$ , a  $\mathbb{K}$ -pure instance of  $\varphi$ , such that  $\rho$  is obtained from  $\varphi$  replacing each nominal  $i$  by  $j$ , where  $L'(i) = L(j)$ . By an induction on the formula complexity, it is easy to see that  $\langle D, (R_r)_{r \in \text{Rel}}, L, \emptyset \rangle \not\models \rho$ . This is a contradiction.  $\square$

With the help of Lemma 15, and since we showed that we can build named models from  $\mathcal{H}\mathcal{L}(@, \mathfrak{r}, \mathbb{K})$ -MCSs, a wide range of strong completeness results can be established.

**Theorem 16** ([7]). *Let  $\Pi$  be a set of pure formulas and let  $\mathcal{A}$  be the axiomatization obtained by adding formulas in  $\Pi$  as axioms to the axiomatization shown in Fig. 1. Then, every  $\mathcal{A}$ -consistent set of formulas is satisfiable in a countable named model in the class defined by  $\Pi$ .*

*Proof.* Given an  $\mathcal{A}$ -consistent set of formulas  $\Omega$ , we can use the Extended Lindenbaum's Lemma to extend it to a named and pasted  $\mathcal{A}$ -MCS  $\Omega^+$ . The named model  $\mathcal{M}^\Omega$  that  $\Omega^+$  gives rise to will satisfy  $\Omega$  at  $\Omega^+$ . In addition, as every formula in  $\Pi$  belongs to every  $\mathcal{A}$ -MCS, we have that  $\mathcal{M}^\Omega \models \Pi$ . Therefore, by Lemma 15,  $\mathcal{M}^\Omega$  is in the class of models defined by  $\Pi$ .  $\square$

To finish this section, we will discuss an extension of the axiomatization presented above, to characterize  $\mathcal{H}\mathcal{L}_\emptyset(@, \mathfrak{r}, \mathbb{K})$ .

**Theorem 17.** *The system obtained by extending the axiomatization in Fig. 1 with the axiom  $(\text{Empty}) \vdash \neg \mathbb{K}$  is sound and strongly complete for  $\mathcal{H}\mathcal{L}_\emptyset(@, \mathfrak{r}, \mathbb{K})$ .*

*Proof.* Soundness of  $\text{Empty}$  is obvious for the class of  $\mathcal{H}\mathcal{L}_\emptyset(@, \mathfrak{r}, \mathbb{K})$ -models. The completeness proof is as the one for  $\mathcal{H}\mathcal{L}(@, \mathfrak{r}, \mathbb{K})$ , but in addition, thanks to  $\text{Empty}$ , all maximal consistent sets  $\Delta_i$  are such that  $\neg \mathbb{K} \in \Delta_i$ . Therefore, the final model yielded by  $\Gamma$ ,  $\mathcal{M}^\Gamma = \langle D^\Gamma, (R_r^\Gamma)_{r \in \text{Rel}}, L^\Gamma, S^\Gamma \rangle$ , is such that  $S^\Gamma = \emptyset$ , and thus, it is a  $\mathcal{H}\mathcal{L}_\emptyset(@, \mathfrak{r}, \mathbb{K})$ -model.  $\square$

**Proposition 18.** *For the case of  $\mathcal{H}\mathcal{L}_\emptyset(@, \mathfrak{r}, \mathbb{K})$ , the result of adding  $\Pi$ , a set of pure formulas, can be extended to a set  $\Pi$  of  $\mathbb{K}$ -pure formulas*

*Proof.* Trivial, using Lemma 15, and the same proof as in Theorem 17.  $\square$

## 4 The Case for $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$

We will present a sound and complete axiomatization for  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$ . In the previous section we mentioned the importance of nominals to describe the interaction between memory and modal operators. In this section we will show that if we restrict ourselves to the logic in which we are constrained to remember the current state every time we make a modal transition, nominals can be avoided. In this logic we can describe the interaction between  $\mathfrak{r}$  and  $\mathfrak{k}$  at a propositional level. This is not a coincidence. Because this logic has the tree model property [3, 4], we can assume that we evaluate  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$ -formulas on trees, and since there are no cycles, the remember operator has no real effect beyond the current state.

Given a formula  $\varphi$ , we define the formula  $\varphi^\sharp$  as the result of replacing all the occurrences of  $\mathfrak{k}$  that are in  $\varphi$  at modal depth zero by  $\top$ . Formally:

$$\begin{aligned} p^\sharp &= p \quad p \in \text{Prop} \\ \mathfrak{k}^\sharp &= \top \\ (\neg\varphi)^\sharp &= \neg\varphi^\sharp \\ (\varphi_1 \wedge \varphi_2)^\sharp &= \varphi_1^\sharp \wedge \varphi_2^\sharp \\ (\mathfrak{r}\varphi)^\sharp &= \mathfrak{r}\varphi^\sharp \\ (\langle r \rangle\varphi)^\sharp &= \langle r \rangle\varphi \end{aligned}$$

**Lemma 19.**  $\mathcal{M}, w \models \mathfrak{r}\varphi$  iff  $\mathcal{M}, w \models \varphi^\sharp$ .

*Proof.* We proceed by induction. The case for  $\mathfrak{k}$ , the propositional symbols and boolean connectives are straightforward. We analyze the other cases. For the case  $\varphi = \mathfrak{r}\psi$ .  $\mathcal{M}, w \models \mathfrak{r}\mathfrak{r}\psi$  iff  $\mathcal{M}, w \models \mathfrak{r}\psi$  iff (by inductive hypothesis)  $\mathcal{M}, w \models \psi^\sharp$  iff  $\mathcal{M}, w \models (\psi^\sharp)^\sharp$  iff (by inductive hypothesis)  $\mathcal{M}, w \models \mathfrak{r}(\psi^\sharp)$  iff  $\mathcal{M}, w \models (\mathfrak{r}\psi)^\sharp$ . For the case  $\varphi = \langle r \rangle\psi$ .  $\mathcal{M}, w \models \mathfrak{r}\langle r \rangle\psi$  iff (by definition)  $\mathcal{M}[w], w \models \langle r \rangle\psi$  iff (by definition of  $\langle r \rangle$ ) there is a  $v \in \mathcal{M}$ ,  $R_r(w, v)$  such that  $\mathcal{M}[w], v \models \psi$  iff (by definition of  $\langle r \rangle$ )  $\mathcal{M}, w \models \langle r \rangle\psi$  iff (by definition of  $\sharp$ )  $\mathcal{M}, w \models (\langle r \rangle\psi)^\sharp$ .  $\square$

We are now ready to present the axiomatization. The axiomatization for  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$  (shown in Fig. 2) is an extension of the axiomatization for the basic modal logic [7], plus the axiom  $\text{Rem}^- \vdash \mathfrak{r}\varphi \leftrightarrow \varphi^\sharp$ .

Soundness of  $\text{Rem}^-$  follows from Lemma 19. We will prove completeness with respect to the class of acyclic models, and therefore for the class of all models. We will use a step-by-step construction. I.e., instead of building the entire canonical model, we will carry out a stepwise selection from MCSs of the canonical model of  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$  as our basic building blocks.<sup>3</sup>

We define  $\mathcal{M}^c = \langle D^c, (R_r^c)_{r \in \text{Rel}}, L^c, S^c \rangle$ , the  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$  canonical model, in the usual sense (see [7] for details). That is,  $D^c$  is the set of all maximal consistent sets of formulas

<sup>3</sup>Alternatively, one can take the standard canonical model and then unravel it to obtain a tree, and therefore acyclic, model.

<b>Axioms:</b>	
$CT$	All classical tautologies
$K_{[r]}$	$\vdash [r](p \rightarrow q) \rightarrow ([r]p \rightarrow [r]q)$
$Rem^-$	$\vdash \textcircled{\mathfrak{F}}\varphi \leftrightarrow \varphi^\sharp$
<b>Rules:</b>	
$MP$	If $\vdash \varphi$ and $\vdash \varphi \rightarrow \psi$ then $\vdash \psi$
$Gen_{[r]}$	If $\vdash \varphi$ then $\vdash [r]\varphi$
$Sub$	If $\vdash \varphi$ then $\vdash \varphi[p/\psi]$ for any $p \in \text{Prop}$

Figure 2: Axiomatization for  $\mathcal{ML}^-(\textcircled{\mathfrak{F}}, \textcircled{\mathfrak{K}})$ .

of  $\mathcal{ML}^-(\textcircled{\mathfrak{F}}, \textcircled{\mathfrak{K}})$ ,  $R_r^c(\Gamma, \Delta)$  iff for all  $\varphi \in \Delta$ ,  $\langle r \rangle \varphi \in \Gamma$ ,  $\Gamma \in L^c(p)$  iff  $p \in \Gamma$  and  $S^c = \{\Gamma \mid \textcircled{\mathfrak{K}} \in \Gamma\}$ .

**Definition 20.** A network  $\mathcal{N} = \langle N, (R_r)_{r \in \text{Rel}}, v \rangle$  is a triple where  $N$  is a countable non-empty set of elements, each  $R_r$  is a binary relation on  $N$ , and  $v$  is a function that maps elements in  $N$  to maximal consistent sets.

We say that a network is coherent if (C1)  $\bigcup_{r \in \text{Rel}} R_r$  defines an acyclic graph and (C2)  $R_r^c(v(s), v(t))$  for all  $s, t \in N$  such that  $R_r(s, t)$ . A network is saturated if whenever  $\langle r \rangle \psi \in v(s)$  for some  $s \in N$ , then there is a  $t \in N$  such that  $R_r(s, t)$  and  $\psi \in v(t)$ .

We want networks to play the role of models, so we have to check that we have imposed the right conditions on a network to achieve this.

**Definition 21.** Let  $\mathcal{N} = \langle N, (R_r)_{r \in \text{Rel}}, v \rangle$  be a network. We define the induced labeling  $L_{\mathcal{N}}(p) = \{s \in N \mid p \in v(s)\}$ , the induced set of remembered states  $S_{\mathcal{N}} = \{s \in N \mid \textcircled{\mathfrak{K}} \in v(s)\}$ , and the induced model  $\mathcal{M}_{\mathcal{N}} = \langle N, (R_r)_{r \in \text{Rel}}, L_{\mathcal{N}}, S_{\mathcal{N}} \rangle$ .  $\mathcal{F}_{\mathcal{N}} = \langle N, (R_r)_{r \in \text{Rel}} \rangle$  is called the underlying frame of  $\mathcal{N}$ .

We are now ready to prove a Truth Lemma.

**Lemma 22** (Truth Lemma). Let  $\mathcal{N} = \langle N, (R_r)_{r \in \text{Rel}}, v \rangle$  be a coherent and saturated network. Then, for all  $\varphi$  and  $s \in N$ ,

$$\mathcal{M}_{\mathcal{N}}, s \models \varphi \text{ iff } \varphi \in v(s).$$

*Proof.* Before we prove this lemma, let us observe the following property: let  $\mathcal{M} = \langle D, (R_r)_{r \in \text{Rel}}, L, S \rangle$  be an acyclic model, and let  $w, v \in D$  be such that  $R_r(w, v)$ . Then for all formulas  $\varphi$ ,  $\mathcal{M}[w], v \models \varphi$  iff  $\mathcal{M}, v \models \varphi$ .

We now proceed by induction on  $\varphi$ . The propositional case, the  $\textcircled{\mathfrak{K}}$  case and the boolean cases are straightforward, given the definition of  $\mathcal{M}_{\mathcal{N}}$ . Let's suppose that  $\mathcal{M}_{\mathcal{N}}, s \models \textcircled{\mathfrak{F}}\psi$ . This happens iff (by Lemma 19)  $\mathcal{M}_{\mathcal{N}}, s \models \psi^\sharp$  iff (by inductive hypothesis)  $\psi^\sharp \in v(s)$  iff (by  $Rem^-$  axiom)  $\textcircled{\mathfrak{F}}\psi \in v(s)$ .

The  $\langle r \rangle$  case: for the left-to-right direction, if  $\mathcal{M}_{\mathcal{N}}, s \models \langle r \rangle \psi$ , then there exists  $t \in N$  such that  $R_r(s, t)$  and  $\mathcal{M}_{\mathcal{N}}[s], t \models \psi$ . Therefore,  $\mathcal{M}_{\mathcal{N}}, t \models \psi$ . By inductive hypothesis,

$\psi \in v(t)$ . Because the network is coherent, and  $R_r(s, t)$ , then  $R_r^c(v(s), v(t))$ , and we conclude  $\langle r \rangle \psi \in v(s)$ . For the other direction, let's suppose that  $\langle r \rangle \psi \in v(s)$ . Because the network is saturated, there is a  $t \in N$  such that  $\psi \in v(t)$  and  $R_r(s, t)$ . By inductive hypothesis,  $\mathcal{M}_{\mathcal{N}}, t \models \psi$ , so  $\mathcal{M}_{\mathcal{N}}[s], t \models \psi$ , and therefore by definition,  $\mathcal{M}_{\mathcal{N}}, s \models \langle r \rangle \psi$ .  $\square$

Summing up, we have reduced the problem of finding a model for an MCS  $\Delta$  to a search for a coherent and saturated network for  $\Delta$ . The idea here is to start with a coherent network and, one step at a time, remove the defects that are preventing the network from being saturated.

**Definition 23.** *Let  $\mathcal{N}$  be a network. We say that  $\mathcal{N}$  has a saturation defect if there is a node  $s \in N$  and a formula  $\langle r \rangle \psi \in v(s)$  such that there is not a  $t \in N$ ,  $R_r(s, t)$  and  $\psi \in v(t)$ .*

Because a coherent network may have saturation defects, we have to say more about what is the meaning of repairing a defect. We are going to *extend* a network with a saturation defect with another where the defect is corrected.

**Definition 24.** *Let  $\mathcal{N}_0 = \langle N_0, R_0, v_0 \rangle$  and  $\mathcal{N}_1 = \langle N_1, R_1, v_1 \rangle$  be two networks. We say that  $\mathcal{N}_1$  extends  $\mathcal{N}_0$  if  $\mathcal{F}_{\mathcal{N}_0}$  is a subframe of  $\mathcal{F}_{\mathcal{N}_1}$  and  $v_0$  agrees with  $v_1$  on  $N_0$ .*

The following lemma states that a saturation defect of a finite coherent network can always be repaired.

**Lemma 25** (Repair Lemma). *Let  $\mathcal{N}$  be a finite and coherent network with a saturation defect. Then there is a network  $\mathcal{N}'$  extending  $\mathcal{N}$  without that defect.*

*Proof.* Because  $\mathcal{N}$  has a a saturation defect, there is a node  $s \in N$  and a formula  $\langle r \rangle \psi \in v(s)$  such that there is not a  $t \in N$ ,  $R_r(s, t)$  and  $\psi \in v(t)$ . We define  $\mathcal{N}'$  as

$$\begin{aligned} N' &= N \cup \{s'\} && \text{with } s' \notin N \\ R'_r &= R_r \cup \{(s, s')\} \\ v' &= v \cup \{(s', \Delta)\} \end{aligned}$$

where  $\Delta$  is an MCS containing  $\psi$  such that  $R_r^c(v(s), \Delta)$  (the existence of such  $\Delta$  can be proved through an Existence Lemma similar to Lemma 10). Clearly,  $\mathcal{N}'$  is a coherent network extending  $\mathcal{N}$  and does not have the previous defect.  $\square$

Now we can prove the desired strong completeness result. We start with a singleton network, and we extend it step by step to a larger network using the Repair Lemma. We obtain the saturated network we want by taking the union of our sequence of networks.

**Theorem 26.** *The axiomatization is strongly complete with respect to the class of  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$  models.*

*Proof.* Let  $S = \{s_i \mid i \in \omega\}$ . Enumerate the potential saturation defects (the set  $S \times \text{Forms}$ ). Given a consistent set  $\Sigma$ , expand it to an MCS  $\Sigma^+$ . The initial network is  $\mathcal{N}^0 = \langle \{s_0\}, \emptyset, (s_0, \Sigma^+) \rangle$ , which is finite and coherent. Given a network  $\mathcal{N}^i$ ,  $i \geq 0$ , where the

minimal saturation defect is  $D$ , we define  $\mathcal{N}^{i+1}$  as the extension of  $\mathcal{N}^i$  (following the Repair Lemma) without that defect. If  $\mathcal{N}^i$  has no saturation defects, then  $\mathcal{N}^{i+1} = \mathcal{N}^i$ . Let  $\mathcal{N}^\omega = \langle N, (R_r)_{r \in \text{Rel}}, v \rangle$  be:

$$N = \bigcup_{n \in \omega} N^n \quad R_r = \bigcup_{n \in \omega} R_r^n \quad v = \bigcup_{n \in \omega} v^n.$$

It is clear that  $\mathcal{N}^\omega$  is saturated. For suppose not; let  $d$  be the minimal saturation defect (with respect to the enumeration) of  $\mathcal{N}^\omega$ , say  $d = d_k$ . By construction, there must be an approximation  $\mathcal{N}^i$  of  $\mathcal{N}^\omega$  of which  $d$  is also a defect. There only can be  $k$  defects that are less than  $d$ , so  $d$  will be repaired before the stage  $k + i$  of the construction. This is a contradiction, so  $\mathcal{N}^\omega$  is a coherent and saturated network, and therefore  $\mathcal{M}_{\mathcal{N}^\omega}, s_0 \models \Sigma$ .  $\square$

## 5 The Erase Operator

In this section we present an axiomatization for  $\mathcal{HL}(@, \textcircled{r}, \textcircled{k}, \textcircled{e})$ , taking as a starting point the axiomatization for  $\mathcal{HL}(@, \textcircled{r}, \textcircled{k})$  presented in Fig. 1. The first thing we should notice is that the *Rem* axiom is no longer sound. For example, take the valid formula  $@_i \textcircled{e}(\textcircled{k} \vee i)$  and use *Rem* to conclude  $@_i \textcircled{r} \textcircled{e} \textcircled{k}$ . This is clearly a contradiction, since after wiping out the memory,  $\textcircled{k}$  cannot be true. Observe that the problem lays in the interaction between  $\textcircled{r}$  and  $\textcircled{e}$ . The replacement operation defined by *Rem* cannot be carried out throughout the whole formula: it should avoid replacements within the scope of an  $\textcircled{e}$ . More formally, for each formula  $\varphi$  and nominal  $i$  we define the formula  $\varphi_i^*$  as follows:

$$\begin{aligned} p_i^* &= p \quad p \in \text{Prop} \cup \text{Nom} \\ \textcircled{k}_i^* &= \textcircled{k} \vee i \\ (\neg \varphi)_i^* &= \neg \varphi_i^* \\ (\varphi_1 \wedge \varphi_2)_i^* &= \varphi_{1i}^* \wedge \varphi_{2i}^* \\ (\textcircled{r} \varphi)_i^* &= \textcircled{r} \varphi_i^* \\ (\langle r \rangle \varphi)_i^* &= \langle r \rangle \varphi_i^* \\ (@_j \varphi)_i^* &= @_j \varphi_i^* \\ (\textcircled{e} \varphi)_i^* &= \textcircled{e} \varphi \end{aligned}$$

Analogously to Lemma 4, we can use  $(\cdot)^*$  to characterize the behavior of the  $\textcircled{r}$  operator and its interaction with the  $\textcircled{e}$  operator.

**Lemma 27.** *Let  $\mathcal{M}$  be a model and  $w \in \mathcal{M}$  such that  $\mathcal{M}, w \models i$ . Then  $\mathcal{M}, w \models \textcircled{r} \varphi$  iff  $\mathcal{M}, w \models \varphi_i^*$ .*

This result naturally suggests an axiom *Rem'* (shown in Fig. 3) that replaces *Rem*. To characterize the  $\textcircled{e}$  operator we should notice first that it behaves globally and that it does not change the evaluation point. This implies that there is no interaction between  $\textcircled{e}$  and  $\neg, \wedge, \langle r \rangle$  and  $@$ . To describe the interaction between  $\textcircled{e}$  and  $\textcircled{r}$  we can again make use of the operation  $(\cdot)^*$ . The detailed axiomatization is in Fig. 3.

<b>Axioms:</b>	
All the axioms from $\mathcal{HL}(@, \mathfrak{r}, \mathfrak{k})$ except <i>Rem</i>	
<i>Rem'</i>	$\vdash @_i(\mathfrak{r}\varphi \leftrightarrow \varphi_i^*)$
<i>Erase<sub>1</sub></i>	$\vdash \mathfrak{e}\neg\mathfrak{k}$
<i>Erase<sub>2</sub></i>	$\vdash \mathfrak{e}s \leftrightarrow s \quad s \in \text{Prop} \cup \text{Nom}$
<i>Erase<sub>3</sub></i>	$\vdash \mathfrak{e}\neg p \leftrightarrow \neg\mathfrak{e}p$
<i>Erase<sub>4</sub></i>	$\vdash \mathfrak{e}(p \wedge q) \leftrightarrow (\mathfrak{e}p \wedge \mathfrak{e}q)$
<i>Erase<sub>5</sub></i>	$\vdash \mathfrak{e}\langle r \rangle p \leftrightarrow \langle r \rangle \mathfrak{e}p$
<i>Erase<sub>6</sub></i>	$\vdash \mathfrak{e}@_i p \leftrightarrow @_i \mathfrak{e}p$
<i>Erase<sub>7</sub></i>	$\vdash @_i(\mathfrak{e}\mathfrak{r}\varphi \leftrightarrow \mathfrak{e}\varphi_i^*)$
<b>Rules:</b>	
All the rules from $\mathcal{HL}(@, \mathfrak{r}, \mathfrak{k})$	

Figure 3: Axiomatization for  $\mathcal{HL}(@, \mathfrak{r}, \mathfrak{k}, \mathfrak{e})$ .

Soundness of this axiomatization is straightforward. The completeness proof uses the same techniques introduced in Sect. 3. The proof of the Truth Lemma is carried out by induction in the structure of the formula, and the new axioms handle the case for  $\mathfrak{e}$  by appropriately reducing the complexity.

## 6 Conclusions and Further Work

In this paper we presented several axiomatizations for some members of the memory logic family. We showed how nominals can be an effective tool to achieve completeness: by allowing to describe the precise interaction between  $\mathfrak{r}$  and  $\mathfrak{k}$  we could give a completeness result for  $\mathcal{HL}(@, \mathfrak{r}, \mathfrak{k})$ . Small variations of this axiomatization leads us to completeness results for other languages, as we showed for  $\mathcal{HL}_\emptyset(@, \mathfrak{r}, \mathfrak{k})$  and  $\mathcal{HL}(@, \mathfrak{r}, \mathfrak{k}, \mathfrak{e})$ . Our intention was to give the basic techniques to characterize memory operators using nominals, and not to exhaustively list all possible languages. Observe that, for example, the logic  $\mathcal{HL}^-(@, \mathfrak{r}, \mathfrak{k})$  can be easily axiomatized by replacing the *Back* axiom presented in Fig. 1 by  $\vdash @_i\langle r \rangle @_j \varphi \rightarrow @_j \varphi[\mathfrak{k}/(\mathfrak{k} \vee i)]$  (and similarly with the *Paste* rule).

We also showed that nominals are not needed when we add constraints on how  $\langle r \rangle$  interacts with  $\mathfrak{r}$ , giving a completeness result for  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$ . The idea behind this result lays in the fact that  $\mathcal{ML}^-(\mathfrak{r}, \mathfrak{k})$  has the tree model property and hence, we can describe the interaction between  $\mathfrak{r}$  and  $\mathfrak{k}$  at a propositional level, independently of the modal operators.

We have not yet found suitable axiomatizations for certain memory logics. Languages without the tree model property, and which do not have nominals seem to be particularly hard to axiomatize. For example, we have not yet been able to devise complete axiomatizations for  $\mathcal{ML}(\mathfrak{r}, \mathfrak{k})$  and  $\mathcal{ML}_\emptyset^-(\mathfrak{r}, \mathfrak{k})$ .

We are also interested in other memory operators, besides the ones presented in this paper. A particularly interesting case is the *forget* operator  $\mathfrak{f}$ , a local version of the erase operator. While the  $\mathfrak{e}$  operator has a global behavior, setting the memory set  $S$  to  $\emptyset$ , we

could conceive a local version which only eliminates the current point of evaluation from  $S$ .

$$\langle D, (R_r)_{r \in \mathbf{Rel}}, L, S \rangle, w \models \mathbb{F}\varphi \text{ iff } \langle D, (R_r)_{r \in \mathbf{Rel}}, L, S \setminus \{w\} \rangle, w \models \varphi$$

We have proved that the logic  $\mathcal{HL}(@, \mathbb{R}, \mathbb{F}, \mathbb{K})$  is strictly more expressive than  $\mathcal{HL}(@, \mathbb{R}, \mathbb{K})$ , but we don't yet have a complete axiomatization. Even having nominals present in the language, we don't know how to characterize the interaction between  $\mathbb{R}$ ,  $\mathbb{F}$  and  $\mathbb{K}$ .

## References

- [1] R. Alur and T. Henzinger. A really temporal logic. In *Journal of the ACM*, pages 164–169. IEEE Computer Society Press, 1989.
- [2] C. Areces. Hybrid logics: The old and the new. In *Proceedings of LogKCA-07*, San Sebastian, Spain, 2007.
- [3] C. Areces, D. Figueira, S. Figueira, and S. Mera. Expressive power and decidability for memory logics. In *Logic, Language, Information and Computation*, volume 5110 of *LNCS*, pages 56–68. Springer Berlin / Heidelberg, 2008.
- [4] C. Areces, D. Figueira, S. Figueira, and S. Mera. Expressive power and decidability for memory logics. *Journal of Computer and System Sciences*, 2008. Submitted. Extended version of [3].
- [5] C. Areces and B. ten Cate. Hybrid logics. In P. Blackburn, F. Wolter, and J. van Benthem, editors, *Handbook of Modal Logics*. Elsevier, 2006.
- [6] P. Blackburn. Representation, reasoning, and relational structures: a hybrid logic manifesto. *Logic Journal of the IGPL*, 8(3):339–625, 2000.
- [7] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.
- [8] P. Blackburn and M. Tzakova. Hybrid completeness. *Logic Journal of the IGPL*, 6(4):625–650, 1998.
- [9] J. Gerbrandy. *Bisimulations on Planet Kripke*. PhD thesis, University of Amsterdam, 1999. ILLC Dissertation series DS-1999-01.
- [10] E. Harel, O. Lichtenstein, and A. Pnueli. Explicit clock temporal logic. In *Proceedings of LICS'90*, pages 402–413, 1990.
- [11] T. Henzinger. Half-order modal logic: How to prove real-time properties. In *Proceedings of the Ninth Annual Symposium on Principles of Distributed Computing*, pages 281–296. ACM Press, 1990.

- [12] C. Lutz. *The complexity of reasoning with concrete domains*. PhD thesis, LuFG Theoretical Computer Science, RWTH Aachen, Germany, 2002.
- [13] J. Plaza. Logics of public communications. In *4th International Symposium on Methodologies for Intelligent Systems*, pages 201–216, 1989.
- [14] J. van Benthem. Logics for information update. In *TARK'01: Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 51–67. Morgan Kaufmann Publishers Inc., 2001.
- [15] J. van Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.
- [16] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*. Kluwer academic publishers, 2007.