Second-order Propositional Modal Logic: Expressiveness and Completeness Results☆

Francesco Belardinelli

Laboratoire IBISC, UEVE & IRIT Toulouse

Wiebe van der Hoek

Department of Computer Science, University of Liverpool

Louwe B. Kuijer

Department of Computer Science, University of Liverpool
LORIA, CNRS/Université de Lorraine

Abstract

In this paper we advance the state-of-the-art on the application of second-order propositional modal logic (SOPML) in the representation of individual and group knowledge, as well as temporal and spatial reasoning. The main theoretical contributions of the paper can be summarized as follows. Firstly, we introduce the language of (multi-modal) SOPML and interpret it on a variety of different classes of Kripke frames according to the features of the accessibility relations and of the algebraic structure of the quantification domain of propositions. We provide axiomatisations for some of these classes, and show that SOPML is unaxiomatisable on the remaining classes. Secondly, we introduce novel notions of (bi)simulations and prove that they indeed preserve the interpretation of formulas in (the universal fragment of) SOPML. We also define (bi)simulation games and show them as powerful as (bi)simulation relations. Then, we apply this formal machinery to study the expressiveness of SOPEL in representing higher-order knowledge, i.e., the knowledge agents have about other agents' knowledge, as well as graph-theoretic notions (e.g., 3-colorability, Hamiltonian paths, etc.) The final outcome is a rich formalism to represent and reason about relevant concepts in artificial intelligence, while still having a model checking problem that is no more computationally expensive than that of the less expressive quantified boolean logic.

Keywords: Modal logic, Knowledge representation, Second-order propositional modal logic, Epistemic logic, Local properties

Email addresses: `belardinelli@ibisc.fr` (Francesco Belardinelli), `wiebe@liverpool.ac.uk` (Wiebe van der Hoek), `louwe.kuijer@liverpool.ac.uk` (Louwe B. Kuijer)

Contents

1. Introduction

Modal logic is nowadays a well-established area in mathematical logic, which has also become one of the most popular formal frameworks in artificial intelligence for knowledge representation and reasoning [9, 31]. This success is due to several reasons, including an expressive and flexible formal language, which enjoys nice computational properties. In particular, at the core of the semantics of modal logic lies the notion of *world*, or *state*. Indeed, this concept is very natural when studying computational notions (a system evolving over time from a previous to a successive state), accounts of agency (states that are preferred, desired, or epistemically possible), and of interaction (states that can be winning, losing, terminal, initial, etc.) Indeed, distributed computing [29], reactive systems [47], multi-agent systems [37], and game theory [36] have all benefited from the application of tools and techniques from modal logic, and this list is by no means exhaustive. Most importantly, the worlds in the models for modal logic are connected by means of indexed relations R_a, for some index a, which model (program) transitions, epistemic or desired alternatives, or the effect of possible moves, where index a can assume a number of readings: a specific program, a dimension of time, say, future or past, an agent, a move, etc. Each accessibility relation R_a in the semantics is then paired with a necessity operator $\Box_a$ in the modal language, where a formula $\Box_a\varphi$ is then true in a world w of a model, if φ is true in every world v that is a-accessible from w (see Definition 9 for a formal definition). Informally, this may be read as: after every execution of a, in each future time along dimension a, in every world considered possible or desired by agent a, or in every world that is the result of performing move a, formula φ holds.

The language of modal logic provides a crisp, variable-free way of expressing a variety of properties of interest. It is also important to realise that there is not just one modal logic: although the well-known normal axiomatisation **K** characterises the class of validities on all models for modal logic, this does not mean that all logics for, say, agency, are the same and correspond to **K**. It only means that they are typically extensions of **K**. As a simple example, the property *(i)* $\Box_a\varphi \to \varphi$ appears reasonable when $\Box_a$ denotes 'agent a knows that . . . ', but is perhaps less desirable when it is read as 'agent a believes that . . . ', as philosophically knowledge is analysed as truthful belief [33]. One of the reasons for the success of modal logic is that in many relevant cases a syntactic scheme corresponds to an additional constraint on the accessibility relation R_a: in the case of *(i)*, reflexivity of R_a is, in a precise sense, sufficient and necessary for its validity.

To appreciate this point, we use a little bit more detail (we assume some familiarity with modal logic, precise definitions are given in Section 2.) As already mentioned, central in the semantics of modal logic is the notion of (Kripke) frame $\mathcal{F}$, which comprises of a set W of worlds and accessibility relations R_a, for indices $a \in I$. We can then define a notion of validity $\vDash$ on frames and formulate the result mentioned above as follows:

$$R_a \text{ is reflexive} \quad \text{iff} \quad \mathcal{F} \vDash \Box_a\varphi \to \varphi, \text{ for all formulas } \varphi \tag{1}$$

Characterisations such as (1) are referred to as *correspondence results* [8], because they establish a correspondence between a first-order property on frames (i.e., reflexivity) and a modal validity (i.e., *(i)*). Another example of correspondence is that between the first-order formula $\forall x \forall y(R_a(x,y) \to R_b(x,y))$ and modal schema $\Box_b\varphi \to \Box_a\varphi$, which intuitively says that, e.g., whatever is achieved by program b, is also achieved by a, or that a knows at least as much as b.

Mathematically elegant and powerful as correspondence theory may be, it also has shortcomings. Firstly, note that in the case of (1), correspondence is defined *globally*, i.e., *(i)* has to be valid throughout the frame. This means that for instance (using a doxastic reading of *(i)*), we cannot model situations in which a's beliefs are true, but b does not know that. Indeed, if the truthfulness of agent a's beliefs is tantamount to the validity of *(i)*, then *(ii)* $K_b(\Box_a\varphi \to \varphi)$ is also a validity, enforcing agent b's knowledge.

Secondly, in (1) quantification appears at the meta-, and therefore the outermost, level. It is therefore impossible to distinguish (and to express in the language of modal logic) the following two situations: in the first, b knows that a has perfect information and is a perfect reasoner, and therefore, b knows *a priori* that whatever a believes must be correct. Informally, this would be represented as $K_b($ for all ϕ $(\Box_a\phi \to \phi))$, which is not a well-formed formula however. In the second situation b has verified, for every φ that a happens to believe, that φ is in fact true. Informally, this would be represented as (for all ϕ, $K_b(\Box_a\phi \to \phi)$).

As observed in [3], by allowing for quantification over propositions – and thus obtaining the language of second-order propositional modal logic (SOPML) – both issues mentioned above can be addressed. The formal definition of $\forall p\psi$ is given in Definition 9, but informally, given a valuation V which tells us in which worlds $V(p)$ the atom p is true, $\forall p\psi$ holds if for every V' that differs from V in at most the set $V(p)$ (i.e., $V'(q) = V(q)$ for all $q \neq p$), the formula ψ holds. As regards to the first example, the SOPML formula $\forall p(\Box_a p \to p) \land \neg K_b\forall p(\Box_a p \to p)$ intuitively expresses that all beliefs of agent a are correct, but b does not know this fact. Moreover, the two different readings in the second example can be represented by formulas $K_b\forall p(\Box_a p \to p)$ and $\forall pK_b(\Box_a p \to p)$, respectively. Readers familiar with the philosophy literature on the topic may recognize the difference between $K_b\forall p(\Box_a p \to p)$ and $\forall pK_b(\Box_a p \to p)$ as the distinction between *de dicto* and *de re* quantification.

Importantly, the truth of $\forall p(\Box_a p \to p)$ at world w enforces the truthfulness of agent a's beliefs in w only, therefore this is a *local* property of the frame, as opposed to the *global* validity of *(i)*. This fact allows agent b to consider (epistemically) possible a different world w' in which *(i)* does not hold.

The aim of this paper is to further the applications of propositional quantification and second-order propositional modal logic in knowledge representation and reasoning, through exploring and securing their theoretical foundations. In particular, the original contributions of the paper can be summarised as follows.

Firstly, in Section 2 we introduce the language of multi-agent second-order propositional modal logic, and provide it with a semantics in terms of Kripke frames extended with a domain D of sets of worlds for the interpretation of quantification. The differences between our definition and the existing definitions of SOPML (e.g. [12, 22, 41]) are that *(i)* in addition to the *full*, *boolean* and *unrestricted* domains of quantification that were studied before, we also consider *modal* domains, and *(ii)* we use a multi-agent language, which allows us to express higher-order properties of knowledge, i.e., knowledge about other agents' knowledge, including truthfulness of knowledge, inclusion of one agent's knowledge in that of another.

In Section 3 we illustrate the richness of the formal framework, particularly to express local properties in modal logic (LPML) [17, 18]. We compare and contrast our approach with [19], and show that the latter can be subsumed in the account here put forward. This validates our endeavour from the viewpoint of applications. However, we maintain

that for SOPML to be adopted as a specification language in artificial intelligence and knowledge representation, appropriate theoretical results and formal tools need to be developed.

To this end, in Section 4 we present a number of results about the axiomatisation of several classes of validites. The key findings are that (i) with the exception of single-agent **S5**, SOPML is unaxiomatisable over all of the commonly used classes of frames, when a full domain of quantification is considered; (ii) on frames with a coarser domain of quantification, SOPML without a common knowledge operator is axiomatisable, but SOPML* with common knowledge is unaxiomatisable in general. As a by-product we obtain several undecidability results.

Furthermore, in Section 5 we develop original truth-preserving bisimulations for SOPML. Bisimulations bring to the fore when two models can be considered the same, and they can be used to test the limits of what can be expressed: when two models for a language $\mathcal{L}$ are bisimilar but disagree on some property $\Phi \in \mathcal{L}'$, it shows that Φ is not expressible in $\mathcal{L}$. We provide several instances of such occurrences. Then, in Section 5.3 we introduce games which characterise exactly these new notion of (bi)simulations. To conclude, our main aim in this paper is to provide formal tools so as to facilitate the use of SOPML as a language for knowledge representation, as well as temporal and spatial reasoning in artificial intelligence.

1.1. Related Work.

This contribution is inspired by a series of papers on LPML, an extension of propositional modal logic to express local properties [17, 18, 19]. Here, instead of introducing an *ad hoc* language (with an adjustment for each local property one has in mind), we make use of the general framework of second-order propositional multi-modal logic. In Section 3 we provide a detailed comparison of the two approaches.

Mono-modal SOPML was first considered in [12, 22, 41], mainly in relation with axiomatisability and (un)decidability questions. In particular, [22] provided several axiomatisations for normal modal logics interpreted on a variety of classes of frames. However, it considered only mono-modal languages, whereas here we adopt a multi-modal perspective. Then, [41] proved an independence result on the axiomatisation of a class of epistemic frames. Notwithstanding these early, significant results, the high computational complexity of SOPML and some undecidability and unaxiomatisability results might partially explain why SOPML has been studied far less than propositional modal logic, and it has been virtually unexplored as a specification language for knowledge representation and reasoning. For instance, only recently SOPML has been proved complete w.r.t. the algebraic semantics in which quantification is interpreted on arbitrary meets and joints [38]. Here we consider a multi-modal version of SOPML, and its epistemic counterpart: second-order propositional epistemic logic (SOPEL).

Among the more recent contributions, [40] shows that there is a validity-preserving translation from second-order logic to SOPML, for modalities weaker than or equal to **S4.2**, implying that for these modalities SOPML is unaxiomatisable. Hereafter we add to the picture and show that multi-modal **S5** is unaxiomatisable as well. Further, [13] provided SOPML with analogues of the van Benthem-Rosen and Goldblatt-Thomason theorems; while in [23] propositional quantification and bisimulations are analysed in the context of modal logic. However, the kind of quantification considered in [23] is preserved by standard bisimulations, and therefore the resulting logic is provably as expressive as

epistemic logic, strictly weaker than SOPML. In [43, 44] the author proves that the quantifier alternation hierarchy of SOPML formulas induces an infinite corresponding semantic hierarchy over the class of finite directed graphs. As a by product, he obtains that, for this class of structures, SOPML with the universal modality and Monadic Second-Order Logic are equally expressive.

Propositional quantification has also been considered in the context of richer modal languages, namely the temporal logics LTL and CTL. A quantified version of LTL, called QLTL, has been introduced and analysed in [53, 52], mainly in relation with the verification of reactive systems. In particular, the model-checking problem for the k-alternation fragment was proved to be k-EXPSPACE-complete. More recently, [45] discusses QCTL, a quantified version of the braching-time temporal logic CTL. The authors prove several complexity and expressivity results for a logic that has more modal operators than SOPML. They also consider two different kinds of semantics for their logic: the former is comparable to SOPML on full frames, while the latter is based on tree-unwindings. To our knowledge, no results are known about the relative expressivity of QCTL and SOPML. It is also outside the scope of this paper to find such results, although it is an interesting question for future research.

More directly related to the present contribution are [3, 4] by some of the authors. In [3] we introduced epistemic quantified boolean logic (EQBL), an epistemic variant of SOPML, and provided axiomatisability and model-checking results. Differently from the reference, here we tackle general SOPML, defined also on modalities strictly weaker than **S5**. Indeed, in this paper we analyse all normal modalities. Moreover, we provide novel unaxiomatisability and undecidability results, as well as give full details on the construction of the canonical models to prove completeness. As regards [4], we define a novel notion of (bi)simulation that generalises the one given therein. Most importantly, for this new definition we are actually able to prove equivalence with (bi)simulation games. This is in marked contrast with [4], where (bi)simulation relations are strictly stronger than (bi)simulation games. Finally, we apply these results to analyse the expressivity of SOPML in capturing relevant properties in temporal and spatial reasoning.

These investigations have been extended to public announcement logic (PAL). Specifically, in [5] the authors applied propositional quantification to PAL to analyses arbitrary public announcements and to formalise notions such as preservation, successfulness, and knowability. Hereafter we do not consider such extensions and keep on a purely epistemic setting.

2. Preliminaries

In this section we introduce the formal machinery that will be used throughout the rest of the paper, and we prove some preliminary results. First, we present the language of second-order propositional modal logic (SOPML), some of its fragments, and their interpretation on Kripke frames and models.

2.1. The Formal Languages

To introduce the language of second-order propositional modal logic, we fix a set AP of atomic propositions and a finite set I of indices. Any language $\mathcal{L}$ built upon AP (using connectives and modal operators) is said to be a language *over AP*.

6

Definition 1 (sopml). *The language $\mathcal{L}_{sopml}$ contains formulas ψ as defined by the following BNF:*

$$\psi \quad ::= \quad p \mid \neg\psi \mid (\psi \rightarrow \psi) \mid \Box_a\psi \mid \forall p\psi$$

where $p \in AP$ and $a \in I$.

For this language and those to be introduced shortly, we will omit parenthesis if doing so causes no confusion. The language $\mathcal{L}_{sopml}$ contains modal formulas $\Box_a\psi$, for every index $a \in I$. A general reading of this would be 'according to the aspect or dimension a, formula ψ holds'. The box can have more concrete interpretations, for instance dynamic (after execution of program or action a, ψ holds), temporal or spatial (along dimensions a, ψ), or deontic (in all situations that abide to norm a, ψ is true). Indices may also denote agents, in which case $\Box_a\psi$ can represent attitudes that relate to goals ('agent a desires ψ', or 'has ψ as a goal'), that are intentional (agent a intends to achieve ψ), or informational ('agent a believes ψ' or 'a knows that ψ'). The latter, epistemic interpretation of $\Box_a$ will obtain some special attention in this paper, and we will write $K_a\psi$ rather than $\Box_a\psi$.

Further, the quantified formula $\forall p\psi$ informally says that 'for all propositions, ψ is true', or, 'for all interpretations of p, ψ obtains'. As standard, the quantifier $\exists$ is dual to $\forall$: $\exists p\psi ::= \neg\forall p\neg\psi$. Analogously, in $\mathcal{L}_{sopml}$, $\Diamond_a\phi$ is a shorthand for $\neg\Box_a\neg\phi$, and M_a is dual to K_a.

Hereafter we consider also the extension $\mathcal{L}^*_{sopml}$ of $\mathcal{L}_{sopml}$ obtain by adding the following clause: if ψ is a formula, then $\Box^*\psi$ is also a formula. Instead of $\Box^*\psi$, in the epistemic interpretation we will write $C\psi$ (it is common knowledge that ψ). To give a hint of what this operator means in epistemic logic, define $E\psi$ (everybody knows that ψ) as $\bigwedge_{a \in I} K_a\psi$. Then, formula $C\psi$ intuitively captures the infinite conjunction $\psi \wedge E\psi \wedge EE\psi \wedge EEE\psi \wedge \ldots$ (the usual definitions for $\top, \bot, \vee, \wedge$, and $\leftrightarrow$ apply). To sum up, whenever we consider the epistemic interpretation of modal operators, we write K_a and C, and define formulas ψ in the language $\mathcal{L}^*_{sopel}$ of second-order propositional epistemic logic (SOPEL) according to the following BNF:

$$\psi \quad ::= \quad p \mid \neg\psi \mid (\psi \rightarrow \psi) \mid K_a\psi \mid C\psi \mid \forall p\psi$$

for $p \in AP$ and $a \in I$. Standard references for modal logic are [9, 10], while for epistemic logic we refer to [21, 49].

We write SOPML for the family of logics that are based on the languages $\mathcal{L}_{sopml}$ and $\mathcal{L}^*_{sopml}$. Throughout most of the paper, it is not very important whether we are considering a language with or without a transitive closure operator $\Box^*$. In the places where the difference between $\mathcal{L}_{sopml}$ and $\mathcal{L}^*_{sompl}$ is important, we write SOPML* for the logic based on $\mathcal{L}^*_{sopml}$.

The name 'second-order propositional modal (epistemic) logic' is related to second-order quantification, as will become apparent in Section 3. In particular, this formalism has been studied in relation to monadic second-order logic – MSO, see [13, 40] and also Section 3.

Example 2. *To give a flavour of the expressivity of $\mathcal{L}^*_{sopml}$, we present some specifications written in this language. We use variants of $\Box_a$ in our notation: their meaning will be clear from the context. Using $\mathcal{L}^*_{sopml}$ one can for instance express that agents a believes*

that agent b will always have some desire p that will remain unfulfilled: $B_a \Box^* \exists p (D_b p \wedge \neg p)$, where operators B_a and D_b are used to represent the doxastic and desire dimensions for agent a and b respectively, whereas $\Box^*$ is interpreted on the reachability relation w.r.t. all agents' moves.

As a further example, formula (i) $\forall p (\Box_a p \to \Box_b p)$ expresses, in a dynamic context, that everything brought about by program a is also brought about by program b, or, provided a doxastic interpretation of the box operator, agent b believes everything that agent a believes. Deontically, the formula $\exists p (Op \wedge \neg p)$ expresses that the current world is not ideal: there are facts that ought to hold, but they do not. Finally, the doxastic-epistemic formula (ii) $K_b \exists p (B_a p \wedge \neg p)$ intuitively expresses that agent b knows that agent a's beliefs are incorrect, while (iii) $\forall p (B_a p \to p) \wedge \Box_\alpha \exists q (B_a q \wedge \neg q))$ denotes that currently, agent a's beliefs are correct, but after executing program α, this ceases to be the case. We remark that by using propositional quantification we can reason about general properties of knowledge, e.g., truthfulness, inclusion, equivalence, of agents' knowledge and beliefs, as in specifications (i), (ii), and (iii).

In this paper we consider various fragments of $\mathcal{L}_{sopml}^*$ and $\mathcal{L}_{sopml}$. To begin with, the languages $\mathcal{L}_{ml}$ of (propositional) modal logic and $\mathcal{L}_{el}$ of (propositional) epistemic logic ($\mathcal{L}_{ml}^*$ and $\mathcal{L}_{el}^*$, respectively) are obtained by removing clause $\forall p\psi$ from the definitions of $\mathcal{L}_{sopml}$ and $\mathcal{L}_{sopel}$ ($\mathcal{L}_{sopml}^*$ and $\mathcal{L}_{sopel}^*$, respectively). Likewise, the language $\mathcal{L}_{qbf}$ of quantified boolean formulas omits clauses $\Box_a\psi$ from $\mathcal{L}_{sopml}$; while propositional logic $\mathcal{L}_{pl}$ is defined as standard by considering only propositional connectives. Moreover, the *universal* fragment $\mathcal{L}_{a-sopml}^*$ of $\mathcal{L}_{sopml}^*$ is defined by the following BNF:

$$\psi \quad ::= \quad p \mid \neg p \mid (\psi \wedge \psi) \mid (\psi \vee \psi) \mid \Box_a \psi \mid \Box^* \psi \mid \forall p \psi$$

Notice that in $\mathcal{L}_{a-sopml}^*$ negation applies to atoms only. Hence, $\mathcal{L}_{a-sopml}^*$ contains no formula of the form $\exists p\psi$, $\Diamond_a\psi$, or $\Diamond^*\psi$. For convenience, we will also denote the set of atoms AP by $\mathcal{L}_{ap}$. A special role in this paper will be played by the languages $\mathcal{L}_x$ of sort x, the set of sort symbols being $\{ap, pl, ml, sopml\}$. We will shortly see that for each sort x, the language $\mathcal{L}_x$ is linked to an interesting class of frames (defined in terms of types y: see the paragraph above Definition 9). This connection is made precise in Lemma 12, item 2. We summarise the main inclusions between languages in Figure 1. We observe that languages $\mathcal{L}_x^*$ are defined only for $x \in \{ml, a - sopml, sopml\}$.

We now introduce some syntactic notions that will be used throughout the paper. Hereafter we use $\sharp$ as a placeholder for any unary operator $\neg$, $\Box_a$, $\Box^*$, and Q for any quantifier $\forall$, $\exists$.

Definition 3 (Subformula and free atoms). *The sets $Sub(\phi)$ and $fr(\phi)$, for the subformulas and free atoms of formula $\phi \in \mathcal{L}_{sopml}^*$, respectively, are recursively defined as follows:*

$Sub(p)$	$=$	$\{p\}$	$fr(p)$	$=$	$\{p\}$
$Sub(\sharp\phi)$	$=$	$\{\sharp\phi\} \cup Sub(\phi)$	$fr(\sharp\phi)$	$=$	$fr(\phi)$
$Sub(\phi \to \phi')$	$=$	$\{\phi \to \phi'\} \cup Sub(\phi) \cup Sub(\phi')$	$fr(\phi \to \phi')$	$=$	$fr(\phi) \cup fr(\phi')$
$Sub(\forall p\phi)$	$=$	$\{\forall p\phi\} \cup Sub(\phi)$	$fr(\forall p\phi)$	$=$	$fr(\phi) \smallsetminus \{p\}$

A *sentence* is a formula ϕ with an empty set of free atoms, i.e., $fr(\phi) = \varnothing$. The set $bnd(\phi)$ of bound atoms in ϕ is defined as standard as the set of all atoms q appearing in
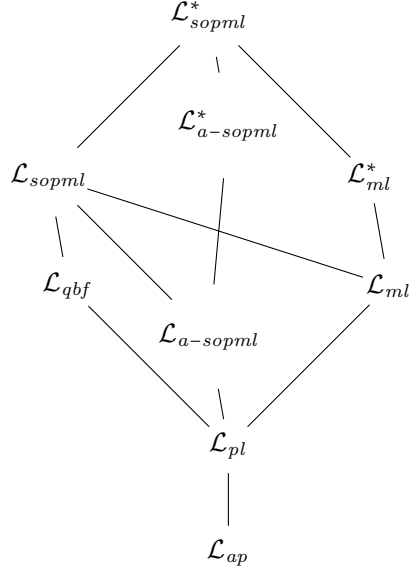
8

Figure 1: Scheme of inclusions among the different languages.

the scope of any quantifier Qq. We assume that for each formula $\phi \in \mathcal{L}^*_{sopml}$, $fr(\phi)$ and $bnd(\phi)$ are disjoint. Actually, we impose that each quantifier binds a different variable. Both constraints can be enforced without loss of generality by renaming bound variables.

Definition 4 (Free for ...). *Given an atom $p \in fr(\phi)$, a formula ψ is free for p in ϕ iff p does not appear in ϕ within the scope of any quantifier Qq for $q \in fr(\psi)$. Alternatively, we can define whether ψ is free for p in ϕ by induction on the structure of ϕ as follows:*

for ϕ atomic, ψ *is free for p in ϕ*
for $\phi = \sharp\, \phi'$, ψ *is free for p in ϕ iff it is in ϕ'*
for $\phi = \phi' \to \phi''$, ψ *is free for p in ϕ iff it is in ϕ' and ϕ''*
for $\phi = \forall q \phi'$, ψ *is free for p in ϕ iff $q \notin fr(\psi)$ and ψ is free for p in ϕ'*

We finally introduce a notion of substitution for free formulas.

Definition 5 (Substitution). *Whenever ψ is free for $p \in fr(\phi)$, the substitution $\phi[p/\psi]$ is inductively defined as follows:*

$$q[p/\psi] \quad = \quad \begin{cases} q & \text{for } q \text{ different from } p \\ \psi & \text{otherwise} \end{cases}$$

$(\sharp\, \phi')[p/\psi] \quad = \quad \sharp(\phi'[p/\psi])$
$(\phi' \to \phi'')[p/\psi] \quad = \quad (\phi'[p/\psi]) \to (\phi''[p/\psi])$
$(\forall r \phi')[p/\psi] \quad = \quad \forall r(\phi'[p/\psi]), \text{ where } r \text{ is assumed different from } p \text{ as } p \in fr(\phi)$

9

Intuitively, ψ being free for p in ϕ means that a substitution of p by ψ in ϕ does not create any new binding. As an example, $\neg q$ is free for p in $\exists r(r \to p)$ but not in $\phi = \exists q(p \leftrightarrow q)$. After we have introduced our semantics, it will be clear that, while $\exists q(p \leftrightarrow q)$ is actually a validity, if we were to blindly substitute p with $\neg q$ in ϕ, we would obtain $\exists q(\neg q \leftrightarrow q)$, which is tantamount to a contradiction. But note that, since $\neg q$ is not free for p in ϕ, by Definition 5, $\phi[p/\neg q]$ is not well-defined. Also note that the procedure above does not guarantee that after a substitution a variable r only occurs in the scope of a single quantifier Qr. For instance, $\forall r(r \to p)[p/\forall r(q \to r)] = \forall r(r \to \forall r(q \to r))$. However, reading the semantics, it will become clear that the latter formula is equivalent to $\forall r(r \to \forall s(q \to s))$: bounded variables can always be renamed (so that, in particular, every formula is equivalent to one in which every formula is bound at most once).

Example 6. *As a further example of the expressive power of* SOPEL, *consider the following specification: agent b knows everything that a knows, and agent c knows this fact, but d does not. This epistemic situation can be recast in $\mathcal{L}_{sopel}$ as the following formula:*

$$\forall p(K_a p \to K_b p) \land K_c \forall p(K_a p \to K_b p) \land \neg K_d \forall p(K_a p \to K_b p)$$

In particular, we can reason further about agent d's knowledge. Indeed, agent d might know that a knows something ignored by b, without being able to explicitly point out the content of a's extra knowledge. This can be recast in $\mathcal{L}_{sopel}$ by the following formula:

$$K_d \exists p(K_a p \land \neg K_b p) \tag{2}$$

However, d could actually know about a specific fact that a knows, but b ignores, as expressed in the following formula:

$$\exists p K_d(K_a p \land \neg K_b p) \tag{3}$$

We remark that (3) corresponds to the de re *reading of our specification, while (2) is its* de dicto *formalisation. Here we do not discuss in detail the* de re/de dicto *distinction, as it is beyond the scope of the present contribution, and refer instead to the seminal paper [50]. In particular, according to our semantics (to be introduced next), (3) is strictly stronger than (and entails) (2). Indeed, the implication (3) $\Rightarrow$ (2) is a validity, but the converse implication (2) $\Rightarrow$ (3) does not hold in general. Thus, among other things,* SOPEL *allows us to distinguish the two readings –* de re *and* de dicto *– of individual knowledge.*

2.2. Kripke Frames and Models

To provide a meaning to formulas of second-order propositional modal logic, we consider multi-modal Kripke frames and models, extended with a domain for the interpretation of quantifiers.

Definition 7 (Kripke frame). *A Kripke frame is a tuple $\mathcal{F} = \langle W, D, R \rangle$ where*

- W *is a set of* possible worlds;

- D *is the* domain of propositions, *i.e., a subset of 2^W;*

- $R : I \to 2^{W \times W}$ *assigns a binary relation on W to each index in I.*

10

As standard in propositional modal logic (PML), for every index $a \in I$, R_a is an accessibility relation between worlds in W [9]. Differently from standard Kripke frames, Definition 7 includes a set $D \subseteq 2^W$ of "admissible" propositions for the interpretation of atoms and quantifiers. Clearly, the Kripke frames in Definition 7 are related to *general frames* [9, 48]. However, there are some notable differences. Firstly, in general frames the domain D of propositions is a boolean algebra with operators, whereas no such assumption holds in the present case. Secondly, the language interpreted on general frames is usually a plain modal logic, while here we address quantification as well. Indeed, propositional quantification makes our language strictly more expressive than propositional modal logic interpreted on general frames, as will become apparent later on (see for instance Examples 10 and 11 and recall that PML on general frames is as expressive as PML).

The accessibility relations can satisfy various properties, e.g., seriality, symmetry, transitivity, reflexivity, etc. When interpreting the language $\mathcal{L}^*_{sopel}$ we assume that each R_a is an equivalence relation (i.e., symmetric, transitive and reflexive), in line with the epistemic reading of modal operators [49]. Finally, for each agent index $a \in I$ and $w \in W$, we let $R_a(w) = \{w' \mid R_a(w, w')\}$. If R_a is an equivalence relation, then $R_a(w)$ is the equivalence class of w according to R_a.

To interpret formulas in $\mathcal{L}^*_{sopml}$ on Kripke frames, we introduce *assignments* as functions $V : AP \rightarrow D$. Also, for $U \in D$, the assignment V^p_U assigns U to p and coincides with V on all other atoms. Hence, atoms can only be assigned propositions in $D \subseteq 2^W$. A *Kripke model* over $\mathcal{F}$ is then defined as a pair $\mathcal{M} = \langle \mathcal{F}, V \rangle$. In the rest of the paper we consider specific classes of Kripke frames and models, which feature pre-eminently in the literature on SOPML [22, 48]. To introduce them, we first define operators $[a] : 2^W \rightarrow 2^W$, for every $a \in I$, such that $[a](U) = \{w \in W \mid R_a(w) \subseteq U\}$; while operator $[\ ]^* : 2^W \rightarrow 2^W$ is introduced so that $[\ ]^*(U) = \{w \in W \mid$ for every $n \in \mathbb{N}$, for every sequence $w_0, \dots, w_n$, if $w_0 = w$ and for every $i < n, w_i = w_{i+1}$ or $R_a(w_i, w_{i+1})$ for some $a \in I$, then $w_n \in U\}$.

Definition 8. *A Kripke frame $\mathcal{F}$ is*

boolean	*iff*	*D is a boolean algebra, i.e., it is closed under intersection, union and complement*
modal	*iff*	*D is a boolean algebra closed under operators $[a]$, for every $a \in I$, and $[\ ]^*$*
full	*iff*	*$D = 2^W$*

A Kripke model $\mathcal{M} = \langle \mathcal{F}, V \rangle$ is *boolean* (*modal*, *full*, respectively) whenever the underlying frame $\mathcal{F}$ is. We distinguish the class $\mathcal{K}_{all}$ of all Kripke frames, the class $\mathcal{K}_{bool}$ of all boolean frames, the class $\mathcal{K}_{modal}$ of all modal frames, and the class $\mathcal{K}_{full}$ of all full frames. Observe that, by using an analogy with monadic second-order logic, the class of full frames corresponds to the basic interpretation of SOPML, where any frame is uniquely identified by fixing the set W of worlds and accessibility relations, as the domain D is equal to 2^W. On the other hand, the other classes of frames are related to the Henkin interpretation of MSO, where D can be a possibly strict subset of 2^W (cf. [55]).

Furthermore, within each of the classes in Definition 8, we will consider further conditions on the accessibility relations R_a: reflexivity r, transitivity t, and symmetry s. Hereafter, given type $y \in Y = \{all, bool, modal, full\}$ and subset $\tau \subseteq \{r, t, s\}$, $\mathcal{K}^\tau_y$ denotes the corresponding class of frames satisfying the properties in τ. For simplicity, $\mathcal{K}^e_y$ denotes

class $\mathcal{K}_y^{\{r,t,s\}}$ (which we also write as $\mathcal{K}_y^{rts}$) of frames in which all accessibility relations are equivalences, that is, the class of epistemic frames for the interpretation of SOPEL. We define a function $\widehat{\cdot}\colon X \to Y$ from language sort symbols to type symbols as follows: $\widehat{ap} = all$; $\widehat{pl} = bool$; $\widehat{ml} = modal$; and $\widehat{sopml} = full$. In total, we obtain 32 classes $\mathcal{K}_y^\tau$ of frames. However, we only consider 20 of them: the subsets $\tau \subseteq \{r,t,s\}$ corresponding to the 5 normal modalities **K**, **T**, **S4**, **B**, and **S5**, combined with the 4 types all, $bool$, $modal$, and $full$. Further classes of frames could be introduced, for instance the class where every formula in $\mathcal{L}^*_{sopml}$ defines a proposition in D. However, such a class is not directly relevant for the results below and its introduction requires a non-trivial general-isation of Kripke frames [48]. Thus, such extensions are beyond the scope of the present paper.

We finally define the notion of satisfaction for formulas in $\mathcal{L}^*_{sopml}$.

Definition 9 (Semantics). *We define whether Kripke model* $\mathcal{M} = \langle \mathcal{F}, V \rangle$ *satisfies formula* $\varphi \in \mathcal{L}^*_{sopml}$ *at world* w, *or* $(\mathcal{M}, w) \vDash \varphi$, *as follows:*

$$
\begin{array}{lll}
(\mathcal{M}, w) \vDash p & \textit{iff} & w \in V(p) \\
(\mathcal{M}, w) \vDash \neg\psi & \textit{iff} & (\mathcal{M}, w) \nvDash \psi \\
(\mathcal{M}, w) \vDash \psi \to \psi' & \textit{iff} & (\mathcal{M}, w) \nvDash \psi \textit{ or } (\mathcal{M}, w) \vDash \psi' \\
(\mathcal{M}, w) \vDash \Box_a \psi & \textit{iff} & \textit{for all } w' \in R_a(w), (\mathcal{M}, w') \vDash \psi \\
(\mathcal{M}, w) \vDash \Box^* \psi & \textit{iff} & \textit{for all } w' \in (\bigcup_{a \in I} R_a)^*(w), (\mathcal{M}, w') \vDash \psi \\
(\mathcal{M}, w) \vDash \forall p \psi & \textit{iff} & \textit{for all } U \in D, (\mathcal{M}_U^p, w) \vDash \psi
\end{array}
$$

where $(\bigcup_{a \in I} R_a)^*$ *is the reflexive and transitive closure of* $\bigcup_{a \in I} R_a$, *and* $\mathcal{M}_U^p = \langle \mathcal{F}, V_U^p \rangle$.

By Definition 9, a quantified formula $\forall p \psi$ (respectively, $\exists p \psi$) is true at world w iff for every (respectively, some) assignment of propositions in D to atom p, ψ is true. Further, as is the case for the common knowledge operator C, $(\mathcal{M}, w) \vDash \Box^* \psi$ iff $(\mathcal{M}, w') \vDash \psi$ for every world w' *reachable* from w, i.e., for every w' such that for some sequence $w_0, \dots, w_k$ of worlds, *(i)* $w_0 = w$, *(ii)* $w_k = w'$, and *(iii)* for every $i < k$, $w_i = w_{i+1}$ or $R_a(w_i, w_{i+1})$ for some $a \in I$. Hence, in non-epistemic contexts, $\Box^*$ can be interpreted as a reachability operator, analogous to the common knowledge operator C.

The satisfaction set $\llbracket \varphi \rrbracket_{\mathcal{M}}$ of formula φ in model $\mathcal{M}$ is defined as $\{w \in W \mid (\mathcal{M}, w) \vDash \varphi\}$. We omit the subscript $\mathcal{M}$ whenever clear by the context. We now introduce various notions of truth and validity. First, we write $(\mathcal{F}, V, w) \vDash \phi$ as a shorthand for $(\langle \mathcal{F}, V \rangle, w) \vDash \phi$. Then, we say that ϕ is *true* at w, or $(\mathcal{F}, w) \vDash \phi$, iff $(\mathcal{F}, V, w) \vDash \phi$ for every assignment V; ϕ is *valid* in a frame $\mathcal{F}$, or $\mathcal{F} \vDash \phi$, iff $(\mathcal{F}, w) \vDash \phi$ for every world w in $\mathcal{F}$; ϕ is *valid* in a class $\mathcal{K}$ of frames, or $\mathcal{K} \vDash \phi$, iff $\mathcal{F} \vDash \phi$ for every $\mathcal{F} \in \mathcal{K}$. Also, ϕ is *true* in a model $\mathcal{M}$, or $\mathcal{M} \vDash \phi$, iff $(\mathcal{M}, w) \vDash \phi$ for every world w. Finally, ϕ is *satisfiable* iff for some model $\mathcal{M}$ and world w, $(\mathcal{M}, w) \vDash \phi$.

Observe that if we define $\mathsf{Th}(\mathcal{K}) = \{\phi \in \mathcal{L}^*_{sopml} \mid \mathcal{K} \vDash \phi\}$, then clearly

$$\mathsf{Th}(\mathcal{K}_{all}) \subseteq \mathsf{Th}(\mathcal{K}_{bool}) \subseteq \mathsf{Th}(\mathcal{K}_{modal}) \subseteq \mathsf{Th}(\mathcal{K}_{full}) \tag{4}$$

In Section 4.1 we show that these inclusions are strict, but first we illustrate some applications of SOPEL in reasoning about knowledge.
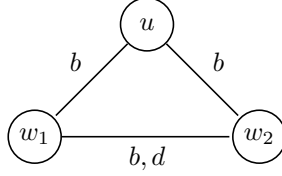
Figure 2: the frame $\mathcal{F}$ for Example 10. (reflexive edges and directions of edges are omitted for simplicity, all relations are equivalences).

Example 10. *Consider a set $I = \{a, b, d\}$ of agents and $AP = \{p\}$ of atoms. The epistemic frame $\mathcal{F} = \langle W, D, R \rangle$ in $\mathcal{K}^e_{all}$ is given with components W and R as depicted in Fig. 2. Moreover, if we suppose that $D = \{\{w_1\}, \{w_2\}\}$, then for every assignment V, $(\mathcal{F}, V, w_1) \vDash \exists p(K_a p \wedge \neg K_b p)$, as $(\mathcal{F}, V^p_{\{w_1\}}, w_1) \vDash K_a p \wedge \neg K_b p$. Similarly, $(\mathcal{F}, V, w_2) \vDash \exists p(K_a p \wedge \neg K_b p)$ by considering assignment $V^p_{\{w_2\}}$. As a consequence, $(\mathcal{F}, w_1) \vDash K_d \exists p(K_a p \wedge \neg K_b p)$, that is, the de dicto formula in Example 6 holds at w_1. However, this is not the case for its de re counterpart, as $(\mathcal{F}, w_1) \nvDash \exists p K_d(K_a p \wedge \neg K_b p)$. To see the latter, note that $K_d(K_a p \wedge \neg K_b p)$ is not true at w_1 for any valuation of p being $\{w_1\}$ or $\{w_2\}$.*

On the other hand, if we suppose that $\mathcal{F}$ is a full frame in $\mathcal{K}^e_{full}$, that is, $D = 2^W$, then we obtain that $(\mathcal{F}, V, w_1) \vDash \exists p K_d(K_a p \wedge \neg K_b p)$, as both $(\mathcal{F}, V^p_{\{w_1, w_2\}}, w_1) \vDash K_a p \wedge \neg K_b p$ and $(\mathcal{F}, V^p_{\{w_1, w_2\}}, w_2) \vDash K_a p \wedge \neg K_b p$.

Example 11. *To assess the expressivity of SOPEL in knowledge representation, we contrast it with comparative epistemic logic – CEL [19]. CEL extends propositional modal logic with formulas $a \succcurlyeq b$, the intuitive interpretation of which is:* agent b knows at least as much as agent a. *Semantically, the clause for satisfaction of such formulas at world w in model $\mathcal{M}$ is given as*

$$(\mathcal{M}, w) \vDash a \succcurlyeq b \quad \textit{iff} \quad R_a(w) \supseteq R_b(w) \tag{5}$$

In this sense $a \succcurlyeq b$ also expresses a local property of frame $\mathcal{F}$, namely the inclusion $R_b(w) \subseteq R_a(w)$.

We show that the comparison between agent a's and agent b's knowledge can be recast in SOPEL as

$$\forall p(K_a p \rightarrow K_b p) \tag{6}$$

In particular, the RHS of (5) is tantamount to the satisfaction of (6) at w, whenever model $\mathcal{M}$ is full. More precisely, for an arbitrary model $\mathcal{M}$ we have

$$(\mathcal{M}, w) \vDash a \succcurlyeq b \quad \Rightarrow \quad (\mathcal{M}, w) \vDash \forall p(K_a p \rightarrow K_b p)$$

while the converse only holds for full $\mathcal{M}$. As a result, formulas $a \succcurlyeq b$ and (6) have the same meaning in the class of full models, and therefore CEL can indeed be mimicked in SOPEL. We discuss this fact in more detail in Section 3.

Moreover, in SOPEL we can make distinctions that are not expressible in epistemic logic. Related to Example 2, in $\mathcal{L}_{sopel}$ we can state that b knows that a's beliefs are not
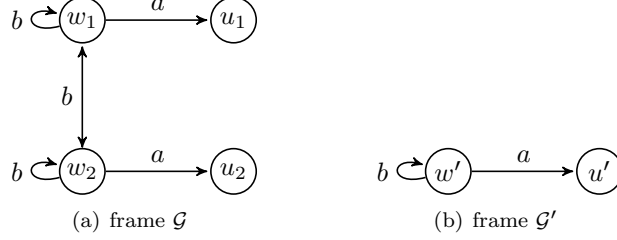
13

Figure 3: Frames $\mathcal{G}$ and $\mathcal{G}'$ in Example 11.

truthful *by using formula*

$$K_b \exists p(B_a p \wedge \neg p) \tag{7}$$

Notice that (7) expresses that b knows that there exists some fact believed by a, which is false, possibly without being able to explicitly point out the actual content of a's false belief. On the other hand, b could actually be aware of some fact which is believed by a but false, as expressed in the following:

$$\exists p K_b(B_a p \wedge \neg p) \tag{8}$$

The formula displayed at (7) is usually referred to as a de dicto *reading of the statement above, where quantifier $\exists p$ appears within the scope of modal operator K_a, while (8) corresponds to the* de re *reading of the same statement, in which $\exists p$ appears outside the scope of K_a (we refer the interested reader in the two different readings to [50]). We remark that (7) and (8) are not equivalent in general, (8) being strictly stronger than (7). Specifically, to account for the difference between (7) and (8), consider frame $\mathcal{G}$ in Fig. 3(a), where the W- and R-components are as depicted, and $D = \{\{w\} \mid w \in W\}$. Clearly, $(\mathcal{G}, V, w_1) \vDash B_a p \wedge \neg p$ for $V(p) = \{u_1\}$, and similarly $(\mathcal{G}, V', w_2) \vDash B_a p \wedge \neg p$ for $V'(p) = \{u_2\}$. Hence, $(\mathcal{G}, w) \vDash (7)$ for $w \in \{w_1, w_2\}$. On the other hand, for no $U \in D$, $(\mathcal{G}, V_U^p, w) \vDash B_a p \wedge \neg p$. Therefore, $(\mathcal{G}, w) \nvDash (8)$ for $w \in \{w_1, w_2\}$. Finally, we observe that $\exists p K_a \phi \to K_a \exists p \phi$ is a validity in every class of frames. As a result, in* SOPEL *(8) is strictly stronger than (7), and we can distinguish the* de dicto *and* de re *readings of agent b's higher-level knowledge.*

Finally, consider frame $\mathcal{G}'$ in Fig. 3(b) with $D' = \{\{w'\} \mid w' \in W'\}$. Let $\mathcal{M}$ and $\mathcal{M}'$ be models based on $\mathcal{G}$ and $\mathcal{G}'$ respectively, in such a way that assignments V and V' make the same atoms true in w_1, w_2, and w', and similarly for u_1, u_2 and u'. One can check that $(\mathcal{M}', w') \vDash (8)$ (and (7) as well). However, $(\mathcal{M}, w_2)$ and $(\mathcal{M}', w')$, satisfy the same formulas in $\mathcal{L}_{ml}$ (indeed, the two models are bisimilar), implying that the de re *formula (8) cannot be expressed in* PML. *We return to this example in Section 5.*

2.3. Preliminary Results

In this section we prove some preliminary results on the model-theory of second-order propositional modal logic, that will be frequently used in the rest of the paper. To start with, in Lemma 12 we extend some basic but useful results in the theory of quantification. In particular, in first-order logic item 1 of Lemma 12 is known as the *coincidence lemma*, and item 2b as the *substitution lemma* (cf. [20]).

14

Lemma 12.

1. Let ϕ be a formula in $\mathcal{L}^*_{sopml}$ and $\mathcal{F}$ a frame in $\mathcal{K}_{all}$. If assignments V and V' coincide on $fr(\phi)$, then

$$(\mathcal{F}, V, w) \vDash \phi \quad \textit{iff} \quad (\mathcal{F}, V', w) \vDash \phi$$

2. Recall that $X = \{ap, pl, ml, sopml\}$ and $\frown = \{(ap, all), (pl, bool), (ml, modal), (sopml, full)\}$. Let $x \in X$. Then,

   (a) For every $\psi \in \mathcal{L}^*_x$ and model $\mathcal{M}$ over $\mathcal{F} \in \mathcal{K}_{\widehat{x}}$, we have $[\![\psi]\!]_{\mathcal{M}} \in D$.

   (b) If $\mathcal{F} \in \mathcal{K}_{\widehat{x}}$ and $\psi \in \mathcal{L}^*_x$ is free for p in ϕ, then

$$(\mathcal{F}, V^p_{[\![\psi]\!]_{\langle \mathcal{F}, V \rangle}}, w) \vDash \phi \quad \textit{iff} \quad (\mathcal{F}, V, w) \vDash \phi[p/\psi]$$

The proof of this lemma is immediate, so we include it only in the appendix. These results show that quantification in SOPML is "well-behaved": by item 1 of Lemma 12, models built over the same frame and agreeing on the interpretation of free atoms, satisfy the same formulas. It follows in particular that a sentence ϕ is either satisfied by any assignment or none, that is, $(\mathcal{F}, w) \vDash \phi$ iff for every model $\mathcal{M}$ over $\mathcal{F}$, $(\mathcal{M}, w) \vDash \phi$, iff for some model $\mathcal{M}$ over $\mathcal{F}$, $(\mathcal{M}, w) \vDash \phi$. As a consequence of Lemma 12, item 2a, the domain of quantification in a model includes the set of denotations of formulas in that model, according to the various fragments of $\mathcal{L}^*_{sopml}$. Moreover, by Lemma 12, item 2b, the syntactic operation of substitution $\phi[p/\psi]$ corresponds to the semantic notion of reinterpretation $\mathcal{M}^p_{[\![\psi]\!]}$.

In Section 4 we will make use of *generated submodels*, a concept that is commonly used in modal logic.

Definition 13 (Submodel). *Given model $\mathcal{M} = \langle W, D, R, V \rangle$ and world $w \in W$, the submodel generated by w is the model $\mathcal{M}_w = \langle W_w, D_w, R_w, V_w \rangle$ such that*

- W_w *is the set of worlds reachable from w, i.e., $W_w = (\bigcup_{a \in I} R_a)^*(w)$;*

- $D_w = \{U_w \subseteq W_w \mid U_w = U \cap W_w \text{ for some } U \in D\}$;

- *for every $a \in I$, $R_{w,a} = R_a \cap W_w^2$;*

- *for every $p \in AP$, $V_w(p) = V(p) \cap W_w$.*

The relevant property of a submodel $\mathcal{M}_w$ is that $(\mathcal{M}, w) \vDash \square^* \phi$ if and only if $(\mathcal{M}_w, w') \vDash \phi$ for every $w' \in W_w$. It is also important to note that if $\mathcal{M}$ is full, modal or boolean, then so is $\mathcal{M}_w$.

Proposition 14. *For $y \in \{all, bool, modal, full\}$ and $\tau \subseteq \{r, t, s\}$, if a frame $\mathcal{F}$ belongs to $\mathcal{K}^\tau_y$ then also $\mathcal{F}_w \in \mathcal{K}^\tau_y$.*

The proof is immediate, so we omit it.

15

2.3.1. Model Checking

In order to explore the computational properties of SOPML, we consider the complexity of its model checking problem. Then, in the next section we analyse the (lack of) finite model property for SOPML. Before we can determine – or even define – the complexity of model checking, however, we first need to define the size of formulas and models. Our definition of the former is entirely as usual.

Definition 15 (Formula Size). *Let $\phi \in \mathcal{L}^*_{sopml}$ be a formula. The* size *of ϕ, denoted $|\phi|$, is given by induction on the length of ϕ as: $|p| = 1$, $|\psi_1 \rightarrow \psi_2| = |\psi_1| + |\psi_2| + 1$, and $|\natural\psi| = |\forall p\psi| = |\psi| + 1$.*

Defining the size of a model is slightly more difficult, since we have to deal with the domain of quantification, which, when counted naively, can be exponentially larger than the other parts of the model. As such, defining $|\langle W, D, R, V\rangle|$ to be equal to $|W| + |D| + |R| + |V|$ would result in an inflated estimate of model size. With respect to this inflated estimate, the model checking problem can even be solved in polynomial time. So it seems more reasonable to exclude D from the size of $\mathcal{M}$ and define $|\mathcal{M}|$ as follows.

Definition 16 (Model Size). *The* size *$|\mathcal{M}|$ of model $\mathcal{M} = \langle W, D, R, V\rangle$ is given by $|\mathcal{M}| = |W| + \sum_{a\in I}|R_a| + \sum_{p\in AP}|V(p)|$.*
A model $\mathcal{M}$ is finite *if $|\mathcal{M}| < \infty$.*

Now that we have defined the seizes of formulas and models, we can define the model checking problem and determine its complexity.

Definition 17 (Model Checking for sopml). *Given a formula $\phi \in \mathcal{L}^*_{sopml}$, a finite model $\mathcal{M}$ and a world w of $\mathcal{M}$, determine whether $(\mathcal{M}, w) \vDash \phi$.*

Then, we are able to prove the following complexity result.

Theorem 18 (Model Checking Complexity).
The model checking problem for SOPML *is PSPACE-complete.*

Proof. As regards hardness, we reduce satisfiability of quantified boolean formulas to SOPML model checking. Given a formula $\phi \in \mathcal{L}_{qbf}$, consider the frame $\mathcal{F} = \langle\{w\}, (w, w), \{\{w\}, \varnothing\}\rangle$ and an arbitrary assignment V, and define $\mathcal{M} = \langle\mathcal{F}, V\rangle$. Then, we have that ϕ is satisfiable iff $(\mathcal{M}, w) \vDash \exists\vec{p}\phi$, where $\vec{p}$ are all the atoms in ϕ. Because the satisfiability problem for quantified boolean formulas is PSPACE-hard, it follows that model checking SOPML is PSPACE-hard as well.

As regards completeness, an algorithm in PSPACE for model checking SOPML is shown as Algorithm 1. It is based on standard model checking algorithms for modal logic, which run in polynomial time. Algorithm 1 takes as input a formula $\phi \in \mathcal{L}^*_{sopml}$ and a finite model $\mathcal{M}$, and returns the set $[\![\phi]\!]_\mathcal{M} \subseteq W$ of worlds satisfying ϕ in $\mathcal{M}$. Then, the model checking problem has a positive answer iff $w \in [\![\phi]\!]_\mathcal{M}$. The case of modal operators is dealt with by computing pre-images of sets according to the accessibility relation, which can be done in polynomial time. The case of propositional quantification is dealt with by exhaustively searching all possible valuations. That is, in order to compute $[\![\forall p\psi]\!]_\mathcal{M}$ we first compute $[\![\psi]\!]_{\mathcal{M}^p_U}$ for every $U \in D$, and then return the intersection of all these extensions. As long as we go through the *for* loop in the $\forall p\psi$ case depth-first, this

algorithm takes only polynomial space. So the model checking problem is in PSPACE, which together with the PSPACE-hardness that we established before implies that the model checking problem for SOPML is PSPACE-complete. □

Algorithm 1 Computation of the satisfaction set $[\![\phi]\!]_{\mathcal{M}}$

---

   switch (ϕ):
   case p**:**
     return $V(p)$;
   case $\neg\psi$**:**
     return $W \smallsetminus [\![\psi]\!]_{\mathcal{M}}$;
   case $\psi \wedge \psi'$**:**
     return $[\![\psi]\!]_{\mathcal{M}} \cap [\![\psi']\!]_{\mathcal{M}}$;
   case $\Box_a\psi$**:**
     return $\{w \in W \mid R_a(w) \subseteq [\![\psi]\!]_{\mathcal{M}}\}$;
   case $\Box^*\psi$**:**
     return $\{w \in W \mid (\bigcup_{a \in Ag} R_a)^*(w) \subseteq [\![\psi]\!]_{\mathcal{M}}\}$;
   case $\forall p\psi$**:**
     initialise $X = W$
     for $U \in D$ **do**
       set $X = X \cap [\![\psi]\!]_{M_U^p}$
     end for
     return X

As a result, model checking SOPML is no more computationally complex than the corresponding problem for quantified boolean formulas. Thus, the enhanced expressiveness comes at no extra computational cost, when compared with QBF. With respect to propositional modal logic, the complexity increases from PTIME to PSPACE. However, this is something to be expected given the extra expressive power of propositional quantification.

2.3.2. Finite Model Property

We now briefly argue why SOPML does not have the final model property. Consider the following set of formulas:

$$\Gamma = \{\Diamond_a \top, \Box_a \Diamond_a \top, \forall p(\Box_a p \to \Box_a \Box_a p), \Box_a \exists p(p \wedge \Box_a \neg p)\}$$

Now suppose that Γ holds at some pointed model $(\mathcal{M}, w)$. Then the first two formulas of Γ require R_a to be serial on $\{w\} \cup R_a(w)$, and the third enforces transitivity of R_a (also at w). Finally, if world w satisfies $\Box_a \exists p(p \wedge \Box_a \neg p)$, then, by Example 24 item 1, we know that $\neg R_a(v, v)$ for all $v \in R_a(w)$, which implies that $\neg R_a(w, w)$, so that R_a is irreflexive over $\{w\} \cup R_a(w)$. But it is easy to verify that a transitive, serial, and irreflexive relation on $R_a(w)$ requires $R_a(w)$ to be infinite. In other words, we found a finite set Γ of formulas in SOPML that only has infinite models.

Theorem 19. *The logic* SOPML *does not have the finite model property.*

Theorem 19 is a generalisation of a result presented in [44, Section 3]. As the name of that section ('SOPMLE = MSO') suggests, it demonstrates that SOPMLE, which is SOPML with a universal modality, has the same expressive power as MSO. And obviously, MSO can force a model to be infinite (use the relational properties of our example above), and therefore SOPMLE can. Note that in our example, we don't assume a universal modality in our language, though.

3. Local Properties in Modal Logic

In the introduction we discussed the difference between a global property as expressed by the modal schema *(i)* $\Box_a \varphi \to \varphi$, whose validity entails that the accessibility relation in a given frame is reflexive, and a local property such as the one represented by the SOPML formula $\forall p (\Box_a p \to p)$ that, as we shall see, on full frames holds exactly in reflexive worlds. Along this line, in [17, 18, 19] a sophisticated account was put forward to express local properties, by adding dedicated modal operators to a basic propositional modal logic. To present the language of local properties in modal logic, or LPML, to compare the two approaches, and more generally to discuss the expressive power of SOPML, we consider a monadic second-order logic and a first-order fragment interpreted on Kripke frames.

Given a frame $\mathcal{F} = \langle W, D, R \rangle$ and a set AP of atoms, we define an MSO alphabet containing binary predicate constants R_a for every agent index $a \in I$, a unary predicate variable P for every atom $p \in AP$, and a set $\mathcal{X}$ of individual variables. Then, MSO formulas Θ in $\mathcal{L}_{mso}$ are defined in BNF as follows:

$$\Theta \quad ::= \quad P(x) \mid x = y \mid R_a(x, y) \mid \neg\Theta \mid \Theta \to \Theta \mid \forall x \Theta \mid \forall P \Theta$$

where $a \in I$ and $x, y \in \mathcal{X}$. Note that the transitive closure R^* of R can be easily defined in MSO.

We also consider the first-order fragment $\mathcal{L}_{fo}$ of MSO obtain by removing clause $\forall P \Theta$ from the BNF above. This is indeed the first-order language considered in [19]. Moreover, we denote as $\mathcal{L}_{fo}^1$ the fragment of $\mathcal{L}_{fo}$ containing formulas with at most one free individual variable. This fragment is well-known to be rich enough to express properties of frames such as reflexivity, symmetry, and transitivity (note that more than one variable is needed for e.g. transitivity, but at most one is free).

As regards the interpretation of MSO and FO (First-Order) formulas, an *assignment* ρ now is a function associating a world $w \in W$ to every individual variable x, and a set $U \in D$ to every predicate variable P. For $w \in W$ and $U \in D$, the variants ρ_w^x and ρ_U^P are defined similarly to SOPML.

Definition 20 (Semantics of mso). *We define whether frame $\mathcal{F} = \langle W, D, R \rangle$ satisfies formula $\Theta \in \mathcal{L}_{mso}$ for an assignment ρ, or $(\mathcal{F}, \rho) \vDash \Theta$, as follows:*

$(\mathcal{F}, \rho) \vDash P(x)$	*iff*	$\rho(x) \in \rho(P)$
$(\mathcal{F}, \rho) \vDash x = y$	*iff*	$\rho(x) = \rho(y)$
$(\mathcal{F}, \rho) \vDash R_a(x, y)$	*iff*	$R_a(\rho(x), \rho(y))$
$(\mathcal{F}, \rho) \vDash \neg\Theta$	*iff*	$(\mathcal{F}, \rho) \nvDash \Theta$
$(\mathcal{F}, \rho) \vDash \Theta \to \Theta'$	*iff*	$(\mathcal{F}, \rho) \nvDash \Theta$ *or* $(\mathcal{F}, \rho) \vDash \Theta'$
$(\mathcal{F}, \rho) \vDash \forall x \Theta$	*iff*	*for all* $w \in W$, $(\mathcal{F}, \rho_w^x) \vDash \Theta$
$(\mathcal{F}, \rho) \vDash \forall P \Theta$	*iff*	*for all* $U \in D$, $(\mathcal{F}, \rho_U^P) \vDash \Theta$

18

Obviously Definition 20 induces an interpretation of formulas in $\mathcal{L}_{fo}$ as well. In particular, for a formula $\Theta(x) \in \mathcal{L}_{fo}^1$, we write $(\mathcal{F}, w) \vDash \Theta$ to denote that $(\mathcal{F}, \rho) \vDash \Theta$ for $\rho(x) = w$, and $\mathcal{F} \vDash \Theta$ if $(\mathcal{F}, w) \vDash \Theta$ for all $w \in W$. The different interpretation of the satisfaction relation $\vDash$ for SOPML and MSO respectively will be clear from the context.

We now briefly recall some basic modal theory on local definability: we refer the interested reader to [9, 10] for further details. We use θ (or $\theta(\vec{a}, \vec{p})$ to emphasise sequences $\vec{a}$ of indices and $\vec{p}$ of atoms) for formulas in $\mathcal{L}_{ml}$. Likewise, we use $\Theta \in \mathcal{L}_{fo}^1$ for first-order formulas with at most one free variable interpreted over states (or $\Theta(\vec{a}, x)$ to denote that Θ mentions $\vec{a}$ as indices and has x as the free variable).

Definition 21. *Let* $\theta \in \mathcal{L}_{ml}$ *and* $\Theta \in \mathcal{L}_{fo}^1$,

1. *θ defines* frame property *Θ iff for all frames $\mathcal{F}$, $\mathcal{F} \vDash \theta$ iff $\mathcal{F} \vDash \Theta$.*

2. *θ locally defines Θ iff for all $\mathcal{F}$ and all $w \in \mathcal{F}$, $(\mathcal{F}, w) \vDash \theta$ iff $(\mathcal{F}, w) \vDash \Theta$.*

As examples of Definition 21, consider the well-known schemes **T** $\Box_a \varphi \to \varphi$, **4** $\Box_a \varphi \to \Box_a \Box_a \varphi$, and **B** $\varphi \to \Box_a \Diamond_a \varphi$, that (locally) define the properties of reflexivity, transitivity, and symmetry on frames.

In the theory of PML, when formula θ locally defines Θ and some other mild conditions hold, one obtains the following connection between axiomatisation and completeness: if an axiomatisation **Ax** is complete for a class $\mathcal{K}$ of frames, then **Ax**$+\theta$ is complete for class $\{\mathcal{F} \in \mathcal{K} \mid \mathcal{F} \vDash \forall x \Theta\}$ of frames satisfying condition Θ. So for instance, taking the basic modal logic **K**, which is sound and complete with respect to the class $\mathbb{K}$ of all frames, the logic **K** + **T** is sound and complete with respect to class $\{\mathcal{F} \in \mathbb{K} \mid \mathcal{F} \vDash \forall x R_a(x, x)\}$, that is, the class of reflexive frames. As further examples, whereas **S5** = **K** + **T** + **4** + **B** is sound and complete with respect to class $\mathbb{S}5 = \{\mathcal{F} \in \mathbb{K} \mid R_a$ is an equivalence relation$\}$, the logic **S5**$+(\Box_b \varphi \to \Box_c \varphi)$ is sound and complete with respect to $\{\mathcal{F} \in \mathbb{S}5 \mid \mathcal{F} \vDash \forall x R_c(x) \subseteq R_b(x)\}$.

This is an appealing modular feature of modal logic. Yet, as also remarked by van Ditmarsch et al. ([17, 18, 19]) this can only be applied if one adds formula θ as a *global* property: assuming θ as an axiom implies that it becomes a validity. For instance, adding formula $B_a \varphi \to \varphi$ to an axiom system, in order to model that agent a's beliefs are correct, implies that in the resulting logic, it is common knowledge that a's beliefs are correct, and this fact will remain true no matter what happens. Likewise, by adding $K_b \varphi \to K_c \varphi$ as an axiom for modelling that c is at least as knowledgeable as b, in the resulting frames and models it will be common knowledge that c knows whatever b knows, and this will again remain true no matter what happens. Thus, it is of interest to study formalisms that can express properties like: 'although a's beliefs are correct, b does not know this' and – for those familiar with dynamic propositional logic [30] – 'c knows everything that b knows, but after b opens the letter, this ceases to hold'. By using propositional quantification we can intuitively formalise such expressions as

$$\forall p(B_a p \to p) \wedge \neg K_b \forall p(B_a p \to p) \tag{9}$$

and

$$\forall p(K_b p \to K_c p) \wedge [\texttt{read}_b] \exists q(K_b q \wedge \neg K_c q) \tag{10}$$

19

$\theta(\vec{a},\vec{p})$	$\Theta(\vec{a},x)$	$\boxdot(\vec{a})$
$\Box_a p \to \Box_b p$	$\forall y(R_b(x,y) \to R_a(x,y))$	$Sup(a,b)$
$\Box_c p \to \Box_a \Box_b\, p$	$\forall y,z(R_a(x,y) \wedge R_b(y,z) \to R_c(x,z))$	$Trans(a,b,c)$
$\neg\,\Box_a \bot$	$\exists y R_a(x,y)$	$Ser(a)$
$\Box_a p \to p$	$R_a(x,x)$	$Refl(a)$
$\neg\,\Box_a\, p \to \Box_b \neg\,\Box_c\, p$	$\forall y,z(R_a(x,y) \wedge R_b(x,z) \to R_c(y,z))$	$Eucl(a,b,c)$
$\neg\,\Box_a\, p \to \neg\,\Box_b \Box_c p$	$\forall z(R_a(x,z) \to \exists y R_b(x,y) \wedge R_c(y,z))$	$Dens(a,b,c)$
$(\neg\,\Box_a\, p \wedge \neg\,\Box_b\, q) \to \neg\,\Box_c\,(p \vee q)$	$\forall y,z((R_a(x,y) \wedge R_b(x,z)) \to (y = z \wedge R_c(x,y)))$	$Func(a,b,c)$

Table 1: as in [19, Table 1], $\Theta(\vec{a},x)$ is a property of state x, and $\boxdot(\vec{a})$ is a name in the object language such that $\boxdot(\vec{a})$ holds at w iff $\Theta(\vec{a},x)$ holds in $\mathcal{M}$ for $\rho(x) = w$.

respectively[1].

To compare our approach based on SOPML to van Ditmarsch et al.'s LPML, we first provide a brief account of the latter.

3.1. Local properties and LPML

This section on LPML is based on [17, 18, 19]: we refer the reader to these references for a more extensive exposition. The term 'logic' is maybe not appropriate for LPML; rather, it is a specific approach to 'connect', in a modal object language, a modal formula $\theta \in \mathcal{L}_{ml}$ and a first-order property $\Theta \in \mathcal{L}_{fo}^1$ through the introduction of a relational atom $\boxdot$ (or $\boxdot(\vec{a})$), in such a way that on Kripke models $\boxdot$ is interpreted as Θ locally. More precisely, the language of LPML extends $\mathcal{L}_{ml}$ with formulas of type $\boxdot(\vec{a})$, whose interpretation is provided by an associated formula $\Theta_{\boxdot}(\vec{a},x) \in \mathcal{L}_{fo}^1$, according to the following satisfaction clause:

$$(\mathcal{M},w) \vDash \boxdot(\vec{a}) \quad \text{iff} \quad (\mathcal{F},w) \vDash \Theta_{\boxdot}(\vec{a},x) \tag{11}$$

By clause (11) we say that formula $\boxdot(\vec{a})$ *expresses locally* first-order property $\Theta_{\boxdot}$ (at w).

Then, LPML investigates how operator $\boxdot$ can help us, in the object language, to build a bridge between modal formulas $\theta_{\boxdot}$ and first-order properties $\Theta_{\boxdot}$ that $\theta_{\boxdot}$ locally defines. So, for instance, we can have $\boxdot(a) = Refl(a)$ for $\Theta_{\boxdot}(a,x) = R_a(x,x)$, or $\boxdot(b,c) = Sup(b,c)$ for $\Theta_{\boxdot}(b,c,x) = \forall y(R_c(x,y) \to R_b(x,y))$ (for more examples, see Table 1). In LPML, property (9) is then represented as $Refl(a) \wedge \neg K_b Refl(a)$, while property (10) is given as $Sup(b,c) \wedge [\mathtt{read}_b]\neg Sup(b,c)$.

Recalling that operator $\boxdot$ is part of the object language of LPML, [19] then adds to the basic modal logic $\mathbf{K}$, for specific formulas $\theta_{\boxdot} \in \mathcal{L}_{ml}$, an axiom $\mathbf{Ax}_{\boxdot}$ and an inference rule $\mathbf{R}_{\boxdot}$. Further, [19, Theorem 2] provides a sufficient condition on the relationship between $\theta_{\boxdot}, \boxdot$ and $\Theta_{\boxdot}$, called *local harmony*, under which $\mathbf{K} + \mathbf{Ax}_{\boxdot} + \mathbf{R}_{\boxdot}$ is a sound and complete axiomatisation for the class of models that satisfy $\Theta_{\boxdot}$.

Definition 22 (Local Harmony). *Formulas* $\theta(\vec{a},\vec{p}) \in \mathcal{L}_{ml}$, $\Theta(\vec{a},x) \in \mathcal{L}_{fo}^1$, *and* $\boxdot(\vec{a})$ *in* LPML *are in* local harmony *iff* (i) θ *(locally) defines* Θ, *and* (ii) $\boxdot$ *expresses* Θ *locally.*

---

[1]Observe that our framework can accommodate the operators of dynamic logic as well: actions are indices α in the set I, which label the different accessibility relations. The interpretation of the associated modal operator $[\alpha]$ is the same as $\Box_a$, for $a \in I$.

A model $\mathcal{M}$ for LPML is a tuple $\langle W, R, I, V \rangle$ where W, R and V are as for PML and I assigns a first order property to each relational atom $\square$. We follow [19] in assuming that for each symbol $\square_\theta(\vec{a})$, there is a $\mathcal{L}_{ml}$-formula $\theta(\vec{a}, \vec{p})$ and a $\mathcal{L}^1_{fo}$-formula $\Theta(\vec{a}, x)$ such that the three are in local harmony and $I(\square_\theta(\vec{a}) = \Theta(\vec{a}, x)$. To be explicit about this, we call such a model $\mathcal{M}$ an *intended model* for LPML.

One could say that LPML as a language is at least as expressive as first-order logic over binary relations, as there are no restrictions, in the object language, on the relational atoms $\square$ that can be added to standard PML. However, the aim of LPML is not to express arbitrary first-order properties Θ, but to reason locally about properties like truthfulness of agent a's beliefs, or an agent c knowing more than b. In particular, there has to exists a modal formula θ that (locally) defines Θ. LPML expresses such first-order properties by adding atoms like $Refl(a)$ and $Sup(b, c)$, respectively. We reckon that SOPML, allowing for quantification over propositions as in $\forall p(B_a p \to p)$ and $\forall p(K_b p \to K_a p)$, is an alternative way to study local properties which is at least as natural as LPML and provably as expressive, in a sense we explain below.

3.2. Local properties, LPML and SOPML on full frames

We first compare LPML to SOPML on full frames. On the other classes of frames there are some notable differences that we discuss in Section 3.3.

Here we show that if formulas $\Theta_\square(\vec{a}, x)$ and $\theta_\square(\vec{a}, \vec{p})$ are in local harmony with some atom $\square(\vec{a})$, then formula $\square(\vec{a})$ is equivalent to $\forall \vec{p}\theta(\vec{a}, \vec{p}) \in \mathcal{L}^*_{sopml}$, within the class of full frames. Hence, SOPML is at least as expressive as LPML. To make this more precise, note that LPML is only able to reason about local properties if all triples $\theta(\vec{a}, \vec{p})$, $\square(\vec{a})$, and $\Theta(\vec{a})$ are in local harmony. Recall that a LPML model $\mathcal{M}$ that guarantees this is an intended model. We will also interpret such an intended model $\mathcal{M}$ as a model for SOPML: one just discards the LPML information connecting $\square(\vec{a})$ and $\Theta_\square(\vec{a}, x)$, and then adds the constraint that the model is full. Now, consider the translation t from LPML formulas to SOPML formulas that distributes over all connectives and modal operators, and moreover says

$$t(\square(\vec{a})) = \forall \vec{p}\theta(\vec{a}, \vec{p})$$

We then obtain the following equivalence result.

Theorem 23. *For every intended* LPML *model* $\mathcal{M}$, $w \in \mathcal{M}$, *and formula* φ *in* LPML, *we have*

$$(\mathcal{M}, w) \vDash \varphi \quad iff \quad (\mathcal{M}, w) \vDash \tau(\varphi)$$

A proof of Theorem 23 is given in Appendix A. This theorem implies, in a sense, that what can be done in LPML, can also be done in SOPML: if $\theta(\vec{a}, \vec{p})$, $\square(\vec{a})$ and $\Theta(\vec{a})$ are in local harmony, then, to reason locally about a scheme θ, one can either use the universal closure $\forall \vec{p}\theta$ in SOPML, or atom $\square(\vec{a})$ in LPML. The result also suggests ways in which SOPML may be more appropriate to reason about local properties, namely cases where Θ is not locally defined by any formula $\theta \in \mathcal{L}_{ml}$, or, conversely, when θ does not define a first-order property Θ locally. Hereafter we consider some similar cases.

Example 24. *Consider the following first-order formulas:*

- $\Theta_1 = \neg R_a(x, x)$ *(irreflexivity)*

21

- $\Theta_2 = \exists x_1, \ldots, x_n \bigwedge_{i \leq n} (R_a(x, x_i) \wedge \bigwedge_{i \neq j \leq n} x_i \neq x_j)$ *(having at least n a-successors)*

- $\Theta_3 = \forall y (R_a(x, y) \wedge R_a(y, x) \to x = y)$ *(anti-symmetry)*

- $\Theta_4 = \forall y (R_b(x, y) \to \neg R_a(x, y))$ *(R_a and R_b have empty intersection)*

- $\Theta_5 = \forall y (R_a(x, y) \wedge R_b(x, y) \to R_c(x, y))$ *(R_c contains the intersection of R_a and R_b).*

It is well-known that these first-order properties are not definable in modal logic [9, 10]. However, consider the following formulas in SOPML*:*

- $\varphi_1 = \exists p (\Box_a p \wedge \neg p)$

- $\varphi_2 = \exists p_1, \ldots, p_n (\bigwedge_{i \leq n} \Diamond_a (p_i \wedge \bigwedge_{j \leq n, j \neq i} \neg p_j))$

- $\varphi_3 = \exists p (p \wedge \forall q (\Diamond_a (q \wedge \Diamond_a p) \to q))$

- $\varphi_4 = \exists p (\Box_a p \wedge \Box_b \neg p)$

- $\varphi_5 = \forall p (\Box_c p \to \exists q (\Box_a q \wedge \Box_b (q \to p)))$

which are such that each φ_i locally defines Θ_i ($1 \leq i \leq 5$). We formalise this result in the following lemma.

Lemma 25. *Consider formulas $\varphi_i \in \mathcal{L}_{sopml}$ and $\Theta_i \in \mathcal{L}_{fo}^1$ in Example 24, for $i = 1, \ldots, 5$.*
Let x be the only free variable in Θ_i and assume $\rho(x) = w$. Assume $\mathcal{F}$ is a full frame, then,

$$(\mathcal{F}, w) \vDash \varphi_i \quad \text{iff} \quad (\mathcal{F}, \rho) \vDash \Theta_i$$

The proof of some items of the lemma is to be found in Appendix A. In particular, SOPML can express properties that are not definable in standard modal logic.

Example 26. *[Distributed Knowledge] To come back to an example from epistemic logic, an interesting notion in collective knowledge is that of* distributed knowledge $D\varphi$*. The intuition here is that distributed knowledge is the knowledge of a 'wise man' (cf. [21]) with whom all agents have shared their knowledge. The typical example is a situation where, for instance, one agent knows φ, another knows that $\varphi \to \psi$, implying distributed knowledge of ψ. A more concrete example goes as follows: it is distributed knowledge in every group of agents (provided everybody knows their own birthday) whether two agents share their birthday. The notion of* distributed knowledge $D\varphi$ *for n agents has an axiomatisation that is sound and complete with respect to models where the corresponding relation R_D is the intersection of all the individual agents' accessibility relations. However, intersection is not locally definable in modal logic (for more on modal properties of distributed knowledge, or implicit knowledge as it is sometimes called, see for instance [35, 51].) However, in* SOPML*, using Example 24.5 we can express that agent c knows exactly what the distributed knowledge of agents a and b is:*

$$\forall p (K_a p \to K_c p) \wedge \forall p (K_b p \to K_c p) \wedge \forall p (K_c p \to \exists q (K_a q \wedge K_b (q \to p))) \qquad (12)$$

Notice that (12) *uses exactly the idea of the typical example of distributed knowledge between two agents discussed above: if agent c knows some fact p, i.e., p is distributed knowledge between a and b, then there exists some fact q such that a knows q and b knows q → p. So, they are able to derive p by pooling together their knowledge.*

Can we generalise this to n agents? Indeed we can, as follows. Define

$$\varphi = \forall p (Dp \to \exists q_1 \ldots \exists q_{n-1} (K_1 q_1 \wedge \cdots \wedge K_{n-1} q_{n-1} \wedge K_n (\bigwedge_{i<n} q_i \to p)))$$

and let

$$\Theta = \forall y ((R_1(x,y) \wedge \cdots \wedge R_n(x,y) \to R_D(x,y)))$$

Then, we can prove the following result.

Proposition 27. *For every full frame $\mathcal{F}$, $(\mathcal{F}, w) \vDash \varphi$ iff $(\mathcal{F}, w) \vDash \Theta(x)$.*

The proof is a generalisation of the proof of Lemma 25 for Θ_5. It follows that operator D locally expresses the distributed knowledge of ψ among agents $1, \ldots, n$:

$$\bigwedge_{i \leq n} \forall p (K_i p \to Dp) \wedge \varphi \wedge D\psi$$

Discussion. From Examples 24 and 26 it follows that SOPML is strictly more expressive than propositional modal logic, and it can also express local properties that cannot be dealt with in LPML. Example 24 also indicates when SOPML can axiomatise frames that cannot be characterised in PML: for instance, formula $\exists p (\Box p \wedge \neg p)$ characterises *irreflexive* frames, in the same way as $\exists p (\Box p \wedge \Diamond \Diamond \neg p)$ characterises *intransitive* frames. Venema [56] calls such characterisations *negatively definable*. The idea here is the following: suppose that formula $\theta \in \mathcal{L}_{ml}$ locally defines some property Θ; is there a formula that locally defines $\neg\Theta$? As an example, whereas $R_a(x,x)$ is (locally) defined by $\Box_a p \to p$, the negation $\neg R(x,x)$ is not (locally) defined by $\neg(\Box p \to p)$, or equivalently, $\Box_a p \wedge \neg p$, since this would require that on frames for this formula, atom p were false. Gabbay [24] came up with a derivation rule, rather than an axiom, to characterise irreflexivity; while [56] analyses more generally when a negative characterisation of some class of frames also leads to an axiomatisation of such class. For our discussion, it is important to realise that reflexivity is actually characterised by a modal *scheme* $\Box_a \varphi \to \varphi$, and, in contrast, by formula $\forall p (\Box_a p \to p)$ in SOPML. But then, irreflexivity is characterised by the negation of that SOPML formula: $\exists p (\Box p \wedge \neg p)$. Moreover, notice that SOPML allows us to interpret such formulas locally, so that we can reason about models that have both reflexive and irreflexive points.

From Example 24 we also learn that there are first-order properties Θ that cannot be characterised by any modal formula $\theta \in \mathcal{L}_{ml}$, while we do have a formula in SOPML characterising it. It is also possible to come up with formulas in SOPML that do not correspond to any first-order formula (hence, in SOPML one could reason locally about them, but not in LPML.) A first example of such formulas is $\exists p \neg\delta$ for $\delta = (\Diamond p \wedge \Diamond \Box \neg p) \to \Diamond(\Box^{-1} \Diamond p \wedge \Box \neg p)$ (here $\Box^{-1}$ is interpreted as the converse of relation R for $\Box$). As argued in [56], although δ as a scheme characterises Dedekind-complete frames among the linear orderings, the frames for $\neg\delta$ are not elementary, that is, not first-order definable. A further example is the Löb formula $\forall p (\Box(\Box p \to p) \to \Box p)$: this formula characterises frames with R being transitive and its converse well-founded [56, p. 8].

23

To conclude our comparison between SOPML and LPML, we observe that the ⊡ operators act in fact as a sort of linguistic black boxes, bringing the metalanguage of the theory of first-order logic into the object language of modal logic. In contrast, SOPML is more transparent, as everything is done in the object language. In addition, for the first-order conditions in [19] there must always be a suitable modal counterpart. Indeed, the axioms $\mathbf{Ax}_\boxdot : \boxdot(\vec{a}) \to \theta_\boxdot(\vec{a}, \vec{p})$ in [19] make sense only as long as there is a propositional modal formula $\theta_\boxdot$ related to $\boxdot$, and this is not always the case as discussed above. We will see later that none of this has to be assumed to axiomatise SOPML.

Revisiting Example 24, it is no surprise that the SOPML formulas in this example all use existential quantification, because we have the following.

Lemma 28. *For a finite set $\{p_1, \ldots, p_n\}$ of atoms, define $\forall \vec{p}$ as $\forall p_1 \ldots \forall p_n$ (this is well-defined because $\forall p \forall q \phi$ is equivalent to $\forall q \forall p \phi$). Then for all frames $\mathcal{F}$, worlds w, and assignments V, we have*

1. $(\mathcal{F}, w) \vDash \phi$ iff $(\mathcal{F}, w) \vDash \forall \vec{p} \phi$

2. $(\mathcal{F}, w) \vDash \forall \vec{fr}(\phi)\ \phi$ iff $(\mathcal{F}, V, w) \vDash \forall \vec{fr}(\phi)\ \phi$

3. $(\mathcal{F}, w) \vDash \phi$ iff $(\mathcal{F}, V, w) \vDash \phi$, where ϕ is a sentence (i.e., $fr(\phi) = \varnothing$).

3.3. Local properties and SOPML *on non-full frames*

So far, we have only looked at how SOPML can represent local properties on full frames. Here, we consider local properties on frames with a coarser domain of quantification. Let us return to formula

$$\forall p(K_a p \to p), \tag{13}$$

which is intended to express that everything a knows is true. Looking at the semantics of SOPML, we can see that (13) holds in $(\mathcal{M}, w)$ if and only if

$$\text{for all } U \in D, R_a(w) \subseteq U \text{ implies } w \in U. \tag{14}$$

If the domain D contains the singleton w, then (14) is equivalent to $w \in R_a(w)$, so to R_a being locally reflexive. In general, however, there is no guarantee that D contains all, or any singletons. So on non-full frames, (13) does not characterize reflexivity. In fact, on non-full frames there is no SOPML formula that characterizes reflexivity.

Whether this is an important downside of SOPML on non-full frames depends on the object of study. If one is after a logic that can reason about graph-theoretic properties like reflexivity, then one should consider SOPML only on full frames, since on other frames SOPML cannot express these properties. If, on the other hand, the goal is to reason about a particular subject (such as knowledge) and only use graphs to represent that subject, then SOPML is useful even on non-full frames. After all, even though (13) does not, in general, correspond to reflexivity, it does still express the fact that everything known by a is true—with one caveat.

The quantor $\forall p$ quantifies only over those valuations of p that are part of the domain D. So, strictly speaking, (13) means that "for every element $U \in D$, if a knows U then U is true." There are three main ways to interpret this.

1. We could explicitly retain the reference to D, and interpret (13) as "every atomic (resp. boolean, modal) proposition that a knows is true" if D is any (resp. boolean, modal) domain of quantification.

2. We could consider D to be the set of properties that are relevant for the problem that we are modeling. In this interpretation, the formula $\forall p(K_a p \to p)$ might hold even if there is some proposition $T \in 2^W \smallsetminus D$ such that T is false but known by a. However, because $T \notin D$ it is not a relevant property, we don't care whether a is wrong about it.

3. We could interpret D as the set of propositions that can be conceptualized. This allows us to interpret (13) as "everything that a knows is true", where it is understood that being able to conceptualize a proposition is a precondition for knowing that proposition.

As an example of the latter situation, suppose that Alice is looking at a blue object. However, due to a trick of the light, the object seems green to her. She forms the belief that the object is green. This belief is false, so $\forall p(K_a p \to p)$ does *not* hold. Now, suppose that Bob is looking at the same object, but that Bob is from a culture that does not distinguish between green and blue. Instead, Bob's culture uses a single concept for these colours that we will translate as "grue". Bob makes the same observation as Alice, but based on that observation he forms the belief that the object is grue. This belief is correct, so, assuming that b's other beliefs are correct as well, $\forall p(K_b p \to p)$ holds.[2] The difference between Alice and Bob does not lie in their accessibility relations. Instead, it is caused by the different ways in which they divide the set of possible worlds into concepts.

The conditions on D then place restrictions on the conceptual space that we assume the agents to have. If D is boolean, then the concepts "green" and "blue" need to be accompanied by concepts "not green" and "green or blue". If D is modal, then the concept "green" needs to be accompanied by a concept "knowing to be green", and if D is full then every set of worlds corresponds to some concept.

Regardless of the interpretation that we choose, every formula discussed in Table 1 expresses a property of SOPML models. The properties that are expressed by these formulas can be found in Table 2. Note that the properties in Table 2 are generalisations of those in Table 1, and that they are equivalent to their counterparts when D is full. The properties in question require some slightly awkward notation, but conceptually they are not very hard to grasp: they correspond to the properties in Table 1, except that here we allow ourselves to substitute a world v for the current world w if both are elements of U.[3]

---

[2] We are somewhat over-simplifying here. Bob is probably capable of conceptualizing several different subsets of grue, such as dark grue and light grue. Regardless, it is quite possible for Bob to have no conceptual distinction between the actual and perceived colours of the object while Alice does have such a distinction.

[3] Admittedly, this correspondence is sometimes rather hard to see due to the aforementioned awkward notation.

SOPML formula	Model property
$\forall p(\Box_a p \to \Box_b p)$	$\forall U \in D(R_a(w) \subseteq U \to R_b(w) \subseteq U)$
$\forall p(\Box_c p \to \Box_a \Box_b p)$	$\forall U \in D(R_c(w) \subseteq U \to (R_b \circ R_a)(w) \subseteq U)$
$\forall p(\neg \Box_a \bot)$	$\exists v R_a(w, v)$
$\forall p(\Box_a p \to p)$	$\forall U(R_a(w) \subseteq U \to w \in U)$
$\forall p(\neg \Box_a p \to \Box_b \neg \Box_c p)$	$\forall U(R_b(w) \nsubseteq U \to \forall v(R_a(w, v) \to R_c(v) \nsubseteq U))$
$\forall p(\neg \Box_a p \to \neg \Box_b \Box_c p)$	$\forall U(R_a(w) \nsubseteq U \to (R_c \circ R_b)(w) \nsubseteq U)$
$\forall p((\neg \Box_a p \wedge \neg \Box_b q) \to \neg \Box_c (p \vee q))$	$\forall U_1 \forall U_2((R_a(w) \nsubseteq U_1 \wedge R_b(w) \nsubseteq U_2) \to (R_c(w) \nsubseteq U_1 \cup U_2))$

Table 2: The model properties expressed by several SOPML formulas.

3.4. Monadic Second-order Logic

We conclude this section by analysing the expressiveness of second-order propositional modal logic through a correspondence between SOPML and monadic second-order logic (MSO), that extends the standard translation between modal and first-order logic [9]. More specifically, ST is the translation between SOPML and MSO defined as follows:

$$
\begin{aligned}
ST_x(p) &= P(x) \\
ST_x(\neg \phi) &= \neg ST_x(\phi) \\
ST_x(\phi \to \phi') &= ST_x(\phi) \to ST_x(\phi') \\
ST_x(\Box_a \phi) &= \forall y(R_a(x, y) \to ST_y(\phi)) \\
ST_x(\Box^* \phi) &= \forall y(R^*(x, y) \to ST_y(\phi)) \\
ST_x(\forall p \phi) &= \forall P(ST_x(\phi))
\end{aligned}
$$

(Recall from page 18 that R^* can be defined in MSO). Clearly, for every formula $\phi \in \mathcal{L}^*_{sopml}$, $ST_x(\phi) \in \mathcal{L}_{mso}$ is a formula where x is the only free individual variable. If $\psi \in \mathcal{L}_{ml}$ is a purely propositional modal formula, then $ST_x(\psi) \in \mathcal{L}_{fo}$ is a first-order formula, as obtained via the standard translation. In particular, $ST_x(\psi)$ belongs to $\mathcal{L}^1_{fo}$.

We now get the following preservation result for the standard translation, that will be used in the completeness proof.

Lemma 29. *For every model $\mathcal{M} = \langle \mathcal{F}, V \rangle$, world $w \in W$, and formula $\psi \in \mathcal{L}^*_{sopml}$,*

$$(\mathcal{M}, w) \vDash \psi \quad \textit{iff} \quad (\mathcal{F}, \rho) \vDash ST_x(\psi)$$

whenever $\rho(x) = w$ and $\rho(P_i) = V(p_i)$.

The proof is mostly standard, and can be found in Appendix A. As a consequence of Lemma 29, there is a one-to-one correspondence between formulas in SOPML and their standard translations in MSO in the following sense: a frame $\mathcal{F}$ validates the universal closure $\forall \vec{p} \psi$ of a formula $\psi \in \mathcal{L}^*_{sopml}$ iff property $\forall x \forall \vec{P} ST_x(\psi) \in \mathcal{L}_{mso}$ holds in $\mathcal{F}$, where $\vec{P}$ are all the unary predicates appearing in $ST_x(\psi)$. As an example, take the McKinsey formula $\Box \Diamond p \to \Diamond \Box p$. Its translation into MSO is $\forall y(R(x, y) \to \exists z(R(y, z) \wedge P(z))) \to \exists u(R(x, u) \wedge \forall v(R(u, v) \to P(v)))$. This is well-known from two independent results in modal logic ([7, Theorem 1] and [26, Section 3]).

4. (In)completeness

One well-established way to understand a logic is to introduce an axiomatisation for it. After all, since there is an infinite number of valid formulas, we cannot explicitly

	S5 ($|I| = 1$)	**S5** ($|I| \geq 2$)	**K**
full	axiomatisable [22]	unaxiomatisable (Thm. 51)	unaxiomatisable [22]
modal	axiomatisable (Thm. 44)		
boolean	axiomatisable [22]	axiomatisable (Thm. 44)	
all	axiomatisable (Thm. 44)		

Table 3: Axiomatisation results for languages $\mathcal{L}$ without common knowledge.

	S5 ($|I| = 1$)	**S5** ($|I| \geq 2$)	**K**
full	axiomatisable [22]	unaxiomatisable (Thm. 51)	unaxiomatisable [22]
modal	axiomatisable (Thm. 44)	unaxiomatisable (Thm. 63)	unaxiomatisable (Thm. 58)
boolean	axiomatisable [22]	unaxiomatisable (Cor. 64)	unaxiomatisable (Cor. 61)
all	axiomatisable (Thm. 44)		

Table 4: Axiomatisation results for languages $\mathcal{L}^*$ with common knowledge.

enumerate all of them. But if we have a complete axiomatisation, then we can at least implicitly know the valid formulas and understand why they are valid.

It turns out that not all variants of SOPML are axiomatisable. Still, even if a logic is unaxiomatisable it is worthwhile to prove that it is so, for two reasons. Firstly, of course, if we have a proof that no axiomatisation exists, then we can stop trying to find an axiomatisation. Secondly, even though an unaxiomatisability result arguably provides less insight regarding the theorems of the logic than an axiomatisation, it does still tell us something about the logic, particularly about its computational complexity.

In this paper we discuss many different variants of SOPML, which differ on the domain of quantification (full, modal, boolean, any), restrictions on the accessibility relations (reflexive, transitive, symmetric), availability of common knowledge, and the number of indices. For some of these variants, axiomatisability and unaxiomatisability results are known from [22, 40, 41]. In particular, [22] provides axiomatisations for all single agent normal logics interpreted on boolean and generic frames, as well as an axiomatisation for epistemic full frames. Here we extend several of these results to the multi-modal case for the first time. Tables 3 and 4 give an overview of these results. In summary, the results are that SOPML without common knowledge is unaxiomatisable on full frames (with the exception of the special case of single-agent **S5**), but axiomatisable on modal, boolean and the class of all frames; while SOPML with common knowledge is unaxiomatisable regardless of the domain of quantification[4]. Note that, if $|I| = 1$ then $\Box^*$ reduces to $\Box$ on **S5** frames, so the entries in the first column of Table 4 follow immediately from the results in the first column of Table 3.

So in most cases, adding common knowledge makes the validity problem harder. This is in contrast to the model checking problem, where adding common knowledge is "free", in the sense that model checking for SOPML is PSPACE-complete, whether or not we have a common knowledge operator (see Theorem 18).

We restrict ourselves to the classes of models for logics **S5** and **K** in these tables. This is because these two classes are the most relevant for our analysis. Our axiomatisability

---

[4]Again, with the exception of the single agent **S5** case.

results are slightly more general, see Theorem 44 for the exact statement. Our unaxiomatisability results are stated for logics **S5** and **K**, but with a few minor modifications these proofs could easily be adapted to other normal modalities, including **KD45** and **S4.2**. Such results give us useful insights into the computational properties of SOPML, as well as its amenability for knowledge representation and reasoning.

Remark 30. *In most of this paper, we use different notation for* SOPML *($\Box_a, \Diamond_a, \Box^*$) and* SOPEL *($K_a, M_a, C$). In this section we discuss both* SOPML *and* SOPEL*, so for the sake of readability we only use the* SOPML *notation.*

Also, we write SOPML *to denote generically logics without operator $\Box^*$; while* SOPML* *refers to logics with $\Box^*$. The distinction will be clear from the context.*

4.1. Complete Axiomatisations

This section is devoted to axiomatise several classes of validities on Kripke frames built on sets I of agent indexes and AP of atomic propositions. We first present a class of logics $\mathbf{K}_x$, one for each $x \in \{ap, pl, ml\}$. In this section all logics are defined on languages without common knowledge.

Definition 31 (Logics $\mathbf{K}_x$). *For each $x \in \{ap, pl, ml\}$, the axioms and inference rules of $\mathbf{K}_x$ are as follows:*

Prop	*all instances of propositional tautologies*
K	$\Box_a(\phi \to \psi) \to (\Box_a \phi \to \Box_a \psi)$
Ex$_x$	$\forall p\phi \to \phi[p/\psi]$, *where $\psi \in \mathcal{L}_x$*
BF	$\forall p \Box_a \phi \to \Box_a \forall p\phi$
MP	*from $\phi \to \psi$ and ϕ infer ψ*
Nec	*from ϕ infer $\Box_a\phi$*
Gen	*from $\phi \to \psi$ infer $\phi \to \forall p\psi$, for p not free in ϕ*

The axioms **Prop** and **K** are standard of any modal logic, as are the rules Modus Ponens (**MP**) and Necessitation (**Nec**). Note how axiom **Ex$_x$** is parameterised by $x \in \{ap, pl, ml\}$. The axiom specifies the language $\mathcal{L}_x$ which acts as the domain of quantification, or, more precisely, what kind of formulas can be substituted as an instance for the universal quantifier. Axiom **BF** is known as the Barcan formula and it says the following. In our models $\mathcal{M} = \langle W, D, R, V \rangle$ the domain of quantification D is defined globally, and does not depend on the world w of evaluation. To give an example where the dependence of D on world w would cease **BF** to hold, consider a structure $\mathcal{N} = \langle W, \{D_w\}_{w \in W}, R, V \rangle$, with $W = \{x, y, z\}$ and $R = \{(x, y), (y, z)\}$. Also, suppose $D_x = \{\{x\}, \{y, z\}\}$ and $D_y = 2^W \neq D_x$. Then, by restricting the clause for quantification in Definition 9 to each D_w, we have $(\mathcal{N}, x) \vDash \forall p\Box(p \to \Box p)$ but not $(\mathcal{N}, x) \vDash \Box\forall p(p \to \Box p)$. In Example 32 we prove that the converse of **BF** is derivable in all $\mathbf{K}_x$.

The scheme of axioms **Ex$_x$** and the Generalisation rule **Gen** are typical principles of quantification. Axiom **Ex$_x$** is the elimination axiom for $\forall$: if something holds for all allowed valuations, then it also holds for each instance from the domain (which can be the set of all atoms, boolean formulas, or modal formulas.) The rule of Generalisation is the introduction rule for $\forall$: if ψ follows from ϕ for an arbitrary p, we infer that $\forall p\psi$ follows from ϕ.

As customary in PML, by considering a suitable combination of axioms

T $\square_a \phi \to \phi$
B $\phi \to \square_a \diamond_a \phi$
4 $\square_a \phi \to \square_a \square_a \phi$

we can introduce the following normal extensions of $\mathbf{K}_x$, also for $x \in \{ap, pl, ml\}$:

$$\begin{aligned}
\mathbf{T}_x &:= \mathbf{K}_x + \mathbf{T} \\
\mathbf{S4}_x &:= \mathbf{K}_x + \mathbf{T} + \mathbf{4} \\
\mathbf{B}_x &:= \mathbf{K}_x + \mathbf{T} + \mathbf{B} \\
\mathbf{S5}_x &:= \mathbf{K}_x + \mathbf{T} + \mathbf{B} + \mathbf{4}
\end{aligned}$$

This gives us 15 logics, 5 for each type $x \in X$.

The notions of *proof* and *theoremhood* are defined as standard. A formula ϕ is *derivable* in logic **L** from a set Δ of formulas, or $\Delta \vdash_{\mathbf{L}} \phi$, iff for some $\phi_0, \ldots, \phi_m \in \Delta$, formula $\bigwedge_{i \leq m} \phi_i \to \phi$ is a theorem in **L**, or $\vdash_{\mathbf{L}} \bigwedge_{i \leq m} \phi_i \to \phi$.

Example 32. *As an example, we provide proofs in logic $\mathbf{K}_{ap}$ of the following theorems and derived inference rules, which will be routinely used in the rest of the section, often without explicit mention:*

- *converse of the Barcan formula* **CBF** $\square \forall p \phi \to \forall p \square \phi$:

1.	$\forall p \phi \to \phi$	*by axiom* $\mathbf{Ex}_{ap}$
2.	$\square_a(\forall p \phi \to \phi) \to (\square_a \forall p \phi \to \square_a \phi)$	*by axiom* $\mathbf{K}$
3.	$\square_a(\forall p \phi \to \phi)$	*from (1) by rule* $\mathbf{Nec}$
4.	$\square_a \forall p \phi \to \square_a \phi$	*from (2), (3) by rule* $\mathbf{MP}$
5.	$\square_a \forall p \phi \to \forall p \square_a \phi$	*from (4) by rule* $\mathbf{Gen}$, *as p is not free in $\square_a \forall p \phi$*

- *vacuous quantification* $\phi \to \forall p \phi$, *whenever p does not appear in ϕ:*

1.	$\phi \to \phi$	*propositional tautology*
2.	$\phi \to \forall p \phi$	*by axiom* $\mathbf{Gen}$

By axiom $\mathbf{Ex}_{ap}$ we then obtain $\phi \leftrightarrow \forall p \phi$, whenever p does not appear in ϕ.

- *distribution of quantification* $\forall p (\phi \to \psi) \to (\forall p \phi \to \forall p \psi)$:

1.	$\forall p(\phi \to \psi) \to (\phi \to \psi)$	*by axiom* $\mathbf{Ex}_{ap}$
2.	$\forall p \phi \to \phi$	*by axiom* $\mathbf{Ex}_{ap}$
3.	$\phi \to (\forall p(\phi \to \psi) \to \psi)$	*from (1) by tautology* $(A \to (B \to C)) \leftrightarrow (B \to (A \to C))$
4.	$\forall p \phi \to (\forall p(\phi \to \psi) \to \psi))$	*from (2), (3) by transitivity of implication*
5.	$\forall p(\phi \to \psi) \to (\forall p \phi \to \psi)$	*from (4) by tautology* $(A \to (B \to C)) \leftrightarrow (B \to (A \to C))$
6.	$(\forall p(\phi \to \psi) \wedge \forall p \phi) \to \psi$	*from (5) by tautology* $(A \to (B \to C)) \leftrightarrow ((A \wedge B) \to C)$
7.	$(\forall p(\phi \to \psi) \wedge \forall p \phi) \to \forall p \psi$	*from (6) by* $\mathbf{Gen}$
8.	$\forall p(\phi \to \psi) \to (\forall p \phi \to \forall p \psi)$	*from (7) by tautology* $(A \to (B \to C)) \leftrightarrow ((A \wedge B) \to C)$

Since **CBF**, *vacuous quantification and distribution of quantification are provable in* **K**$_{ap}$, *they are theorems in all the other 14 logics above. We also recall that inferring* $\Box_a \phi \rightarrow \Box_a \psi$ *from* $\phi \rightarrow \psi$ *is a derivable rule in modal logic, and whenever p does not appear free in* ϕ, *formula* $\forall p(\phi \rightarrow \psi) \rightarrow (\phi \rightarrow \forall p\psi)$ *is a theorem in all our logics.*

We now prove the soundness and completeness results for logics $\mathbf{L}_x$ w.r.t. the corresponding class $\mathcal{K}$ of Kripke frames, starting with soundness. In the rest of the paper $\mathbf{L}$ ranges over $\{\mathbf{K}, \mathbf{T}, \mathbf{S4}, \mathbf{B}, \mathbf{S5}\}$. Given a logic $\mathbf{L}_x$, let $\tau(\mathbf{L}_x)$ be a subset of $\{r, t, s\}$, such that $\mathbf{L}_x$ includes axiom $\mathbf{T}$ iff τ contains r (for reflexivity), $\mathbf{L}_x$ includes axiom $\mathbf{4}$ iff τ contains t (for transitivity), and $\mathbf{L}_x$ includes axiom $\mathbf{B}$ iff τ contains s (for symmetry).

Theorem 33 (Soundness). *For* $x \in \{ap, pl, ml\}$, *for every logic* $\mathbf{L}_x$ *and formula* $\phi \in \mathcal{L}_{sopml}$,

$$\vdash_{\mathbf{L}_x} \phi \text{ implies } \mathcal{K}_{\widehat{x}}^{\tau(\mathbf{L}_x)} \vDash \phi$$

Proof. As customary, the axioms of each logic $\mathbf{L}_x$ are shown to be valid in the corresponding class $\mathcal{K}_{\widehat{x}}^{\tau(\mathbf{L}_x)}$ of frames, and the inference rules are shown to preserve validity in $\mathcal{K}_{\widehat{x}}^{\tau(\mathbf{L}_x)}$. Specifically, axioms **Prop**, **K**, **MP**, and **Nec** are valid in any frame. The validity of axioms **T**, **4**, and **B** in specific classes of frames follows as in standard propositional modal logics [9]. The validity of axioms $\mathbf{Ex}_x$ in each corresponding class of frames follows by Lemma 12.(2); while the validity of **Gen** follows by Lemma 12.(1). We provide a proof for $\mathbf{Ex}_{ap}$: suppose that $(\mathcal{M}, w) \vDash \forall p\phi$, that is, for every $U \in D$, $(\mathcal{M}_U^p, w) \vDash \phi$. By Lemma 12.(2a), $[\![\psi]\!] \in D$, hence in particular $(\mathcal{M}_{[\![\psi]\!]}^p, w) \vDash \phi$. Then, by Lemma 12.(2b), $(\mathcal{M}, w) \vDash \phi[p/\psi]$. As regards **Gen**, suppose that $(\mathcal{M}, w) \vDash \phi$ and $p \notin fr(\phi)$. In particular, for every $U \in D$, $V(fr(\phi)) = V_U^p(fr(\phi))$, and by Lemma 12.(1), we have $(\mathcal{M}_U^p, w) \vDash \phi$ as well. By **MP** we obtain that $(\mathcal{M}_U^p, w) \vDash \psi$, and since U is arbitrary, $(\mathcal{M}, w) \vDash \forall p\psi$. Moreover, the Barcan formula **BF** is valid as in any frame all worlds have the same domain of quantification, namely $D \subseteq 2^W$. Indeed, $(\mathcal{M}, w) \vDash \forall p \Box_a \phi$ iff for all $U \in D$, $(\mathcal{M}_U^p, w) \vDash \Box_a \phi$, iff for every $w' \in R_a(w)$, $(\mathcal{M}_U^p, w') \vDash \phi$. But this means that for every $w' \in R_a(w)$, $(\mathcal{M}, w') \vDash \forall p\phi$, that is, $(\mathcal{M}, w) \vDash \Box_a \forall p\phi$. $\square$

As a consequence of Theorem 33, all our 15 logics are sound w.r.t. the corresponding classes of frames. Moreover, as a by-product of soundness, we obtain that the inclusions (4) between theories put forward in Section 2.2 are all strict:

$$\mathsf{Th}(\mathcal{K}_{all}) \subset \mathsf{Th}(\mathcal{K}_{bool}) \subset \mathsf{Th}(\mathcal{K}_{modal}) \subset \mathsf{Th}(\mathcal{K}_{full}) \tag{15}$$

To prove this, observe that each axiom $\mathbf{Ex}_x$ holds in $\mathcal{K}_x$, but in no more general class of frames. Finally, let $\mathbf{Ex}_{sopml}$ be the scheme $\forall p\phi \rightarrow \phi[p/\psi]$, for $\psi \in \mathcal{L}_{sopml}$. It is easy to check that $\mathbf{Ex}_{sopml} \in \mathsf{Th}(\mathcal{K}_{full})$, but $\mathbf{Ex}_{sopml} \notin \mathsf{Th}(\mathcal{K}_{modal})$.

Next we state the completeness result. Here we use the notation of Theorem 33.

Theorem 34 (Completeness). *For* $x \in \{ap, pl, ml\}$, *and every formula* $\phi \in \mathcal{L}_{sopml}$,

$$\mathcal{K}_{\widehat{x}}^{\tau(\mathbf{L}_x)} \vDash \phi \quad \text{implies} \quad \vdash_{\mathbf{L}_x} \phi$$

Theorem 34 guarantees completeness of a logic of sort x, with respect to models of type $\widehat{x}$, for the sorts of atomic propositions and propositional and modal formulas. Completeness also holds if we add properties such as reflexivity, transitivity, and symmetry

30

to the frames, as long as we add the corresponding axioms from $\{\mathbf{T}, \mathbf{4}, \mathbf{B}\}$ to the logic. To our knowledge this is the first completeness result for SOPML in a multi-agent setting.

To prove Theorem 34 for $x \in \{ap, pl, ml\}$, we show that if a formula ϕ is $\mathbf{L}_x$-consistent, then we can construct an appropriate model $\mathcal{M}_{\mathbf{L}_x}$ that satisfies ϕ. For logic $\mathbf{L}_{ap}$ (respectively, $\mathbf{L}_{pl}$, $\mathbf{L}_{ml}$) this amounts to finding models whose underlying frame is any frame (respectively, a boolean algebra or a boolean algebra with operators). We begin with the cases for $\mathbf{L}_{ap}$ for clarity's sake.

4.1.1. Completeness of $\mathbf{L}_{ap}$

In this section we show that if a formula ϕ is $\mathbf{L}_{ap}$-consistent, that is, $\nvdash_{\mathbf{L}_{ap}} \neg\phi$, then we can construct a (canonical) model $\mathcal{M}_{\mathbf{L}_{ap}} = \langle \mathcal{F}, V \rangle$ that satisfies ϕ. Moreover, $\mathcal{F}$ is shown to belong to the class $\mathcal{K}_{all}$ of all frames. This implies that $\mathcal{K}_{all} \nvDash \neg\phi$.

Definition 35. *Let $\Lambda \subseteq \mathcal{L}_{sopml}$ be sets of formulas over set AP of atoms, and Y a denumerable set of atoms. We say that Λ is*

$\mathbf{L}_{ap}$*-consistent*	*iff*	$\Lambda \nvdash_{\mathbf{L}_{ap}} \bot$
complete	*iff*	*for every formula $\phi \in \mathcal{L}_{sopml}$, $\phi \in \Lambda$ or $\neg\phi \in \Lambda$*
maximal	*iff*	*Λ is consistent and complete*
Y-rich	*iff*	*for every $\phi \in \mathcal{L}_{sopml}$, if $\exists p\phi \in \Lambda$ then $\phi[p/q] \in \Lambda$ for some $q \in Y$*
saturated	*iff*	*Λ is maximal and Y-rich for some $Y \subseteq AP$*

We omit the subscript $\mathbf{L}_{ap}$ whenever clear by the context.

We remark that, by the definition of derivability, a set Λ is inconsistent iff for some $\phi_0, \ldots, \phi_m \in \Lambda$, $\vdash \bigwedge_{i \leq m} \phi_i \to \bot$, that is, $\vdash \bigwedge_{i \leq m} \phi_i \to \neg\psi$ for every $\psi \in \Lambda$.

We now prove that every consistent set can be saturated.

Lemma 36 (Saturation). *Let Δ be a maximal set of formulas over AP. Then there exists a saturated set of formulas Φ over $AP \cup Y$, such that $\Delta \subset \Phi$, where Y is an infinite set of new atoms (i.e., disjoint from AP).*

Proof. Let $\theta_0, \theta_1, \ldots$ be an enumeration of the formulas over $AP \cup Y$, and $q_0, q_1, \ldots$ an enumeration of atoms in Y. We define by induction a sequence $\Phi_0, \Phi_1, \ldots$ of sets of formulas over $AP \cup Y$ as follows:

$$\Phi_0 = \Delta$$

$$\Phi_{n+1} = \begin{cases} \Phi_n \cup \{\theta_n\} & \text{if } \Phi_n \cup \{\theta_n\} \text{ is consistent and } \theta_n \text{ is not of the form } \exists p\chi; \\ \Phi_n \cup \{\theta_n, \chi[p/q]\} & \text{if } \Phi_n \cup \{\theta_n\} \text{ is consistent, } \theta_n \text{ is of the form } \exists p\chi, \\ & \text{and } q \in Y \text{ is the first atom not appearing in } \Phi_n \cup \{\theta_n\}; \\ \Phi_n \cup \{\neg\theta_n\} & \text{otherwise.} \end{cases}$$

Notice that, since Y is an infinite set of new atoms and finitely many θ appear in $\Phi_n \smallsetminus \Phi_0$, for each $n \in \mathbb{N}$, we can always find an atom $q \in Y$ that does not appear in $\Phi_n \cup \{\theta_n\}$. Now we prove by induction on n that every Φ_n is consistent. First of all, $\Phi_0 = \Delta$ is consistent by hypothesis. As to the inductive step, suppose that Φ_n is consistent, we consider the various cases. If $\Phi_{n+1} = \Phi_n \cup \{\theta_n\}$ and θ_n is not of the form $\exists p\chi$, then $\Phi_n \cup \{\theta_n\} = \Phi_{n+1}$ has to be consistent, by construction. Further, $\Phi_{n+1} = \Phi_n \cup \{\theta_n, \chi[p/q]\}$

if $\Phi_n \cup \{\theta_n\}$ is consistent, θ_n is of the form $\exists p\chi$, and q is the first atom that does not appear in $\Phi_n \cup \{\theta_n\}$. To obtain a contradiction, suppose that Φ_{n+1} is inconsistent. In particular, for some $\varphi_0, \ldots, \varphi_m \in \Phi_n$,

$$\vdash \quad (\bigwedge_{i \leq m} \varphi_i \wedge \theta_n) \to \neg\chi[p/q]$$

Since q is assumed not to appear in Φ_n nor in θ_n, by an application of **Gen** we obtain

$$\vdash \quad (\bigwedge_{i \leq m} \varphi_i \wedge \theta_n) \to \forall p \neg\chi$$

i.e., $\vdash (\bigwedge_{i \leq m} \varphi_i \wedge \theta_n) \to \neg\theta_n$, and, since $\vdash (\bigwedge_{i \leq m} \varphi_i \wedge \theta_n) \to \theta_n$ trivially, we obtain that $\vdash (\bigwedge_{i \leq m} \varphi_i \wedge \theta_n) \to \bot$, that is, $\Phi_n \cup \{\theta_n\}$ is not consistent, against hypothesis. Hence, $\Phi_{n+1} = \Phi_n \cup \{\theta_n, \chi[p/q]\}$ is indeed consistent.

Finally, $\Phi_{n+1} = \Phi_n \cup \{\neg\theta_n\}$ only if $\Phi_n \cup \{\theta_n\}$ is not consistent. Indeed, if Φ_n is consistent, $\Phi_n \cup \{\theta_n\}$ and $\Phi_n \cup \{\neg\theta_n\}$ cannot be both inconsistent, since otherwise for some $\varphi_0, \ldots, \varphi_m, \varphi'_0, \ldots, \varphi'_{m'} \in \Phi_n$,

$$\vdash \quad \bigwedge_{i \leq m} \varphi_i \to \neg\theta_n$$
$$\vdash \quad \bigwedge_{i \leq m'} \varphi'_i \to \neg\neg\theta_n$$

and by propositional reasoning,

$$\vdash \quad \bigwedge_{i \leq m} \varphi_i \wedge \bigwedge_{i \leq m'} \varphi'_i \to (\neg\theta_n \wedge \theta_n)$$

and therefore Φ_n itself is inconsistent, a contradiction. Hence, $\Phi_n \cup \{\neg\theta_n\} = \Phi_{n+1}$ is indeed consistent.

Now let $\Phi = \bigcup_{n \in \mathbb{N}} \Phi_n$: Φ is consistent as each Φ_n is. If that were not the case, there would be $\varphi_0, \ldots, \varphi_m \in \Phi$ such that $\vdash \bigwedge_{i \leq m} \varphi_i \to \bot$. Then suppose that k is the smallest index such that all ϕ_i appear in Φ_k. It follows that Φ_k is inconsistent as well, against hypothesis. Moreover, Φ extends Δ and it is maximal and Y-rich by construction. $\square$

We now describe informally the construction of the canonical model for a formula ϕ such that $\nvdash \neg\phi$. First, define W as the set of all saturated sets w of formulas over $AP \cup Y$ as obtained in Lemma 36. Notice that W is non-empty as the set $\{\phi\}$ is consistent by hypothesis, and by Lemma 36 there exists a saturated set $\Phi \supseteq \{\phi\}$ in W. Further, for $w, w' \in W$ and $a \in I$, define $R_a(w, w')$ iff $\{\psi \mid \Box_a \psi \in w\} \subseteq w'$. Finally, for every atom $p \in AP \cup Y$, we consider set $U_p = \{w \in W \mid p \in w\} \subseteq W$ and define the domain D of propositions as $\{U_p \mid p \in AP \cup Y\}$.

Definition 37 (Canonical Model). *The canonical model* **L** *is a tuple* $\mathcal{M}_\mathbf{L} = \langle W, D, R, V \rangle$ *where* (i) W, D *and* R *are defined as above; and* (ii) V *is the assignment such that* $V(p) = U_p$.

Note that every consistent formula ϕ must be contained in some $\Phi \in W$. Next we prove that the canonical model w.r.t. any $\mathbf{L}_{ap}$ is indeed a model based on a frame in $\mathcal{K}_{all}^{\tau(\mathbf{L}_{ap})}$ (recall that $\widehat{ap} = all$ and that $\mathbf{L}_{ap}$ represents 5 different logics: $\mathbf{K}_{ap}, \mathbf{T}_{ap}, \mathbf{S4}_{ap}, \mathbf{B}_{ap}$, and $\mathbf{S5}_{ap}$).

Lemma 38. *The canonical model $\mathcal{M}_{\mathbf{L}_{ap}}$ in Definition 37 is a Kripke model based on a frame in $\mathcal{K}_{all}^{\tau(\mathbf{L}_{ap})}$.*

Proof. By the remarks above, W is a non-empty set of saturated sets, D is a subset of 2^W, and V is a function from $AP \cup Y$ to D. Moreover, axiom **T** (respectively, **4**, **B**) enforces relation R_a on W to be reflexive (respectively, transitive, symmetric), as it is the case for propositional modal logic. As an illustrative example, we consider the case for **T**: by maximality, for every $w \in W$, $\Box_a \psi \to \psi \in w$ for every formula $\psi \in \mathcal{L}_{sopml}$, and by closure under MP we have $\{\psi \in \mathcal{L}_{sopml} \mid \Box \psi \in w\} \subseteq w$, that is, $R_a(w, w)$ by definition. $\square$

We can finally prove the truth lemma for logics $\mathbf{L}_{ap}$. Here we adapt the proof in [21] for propositional epistemic languages without common knowledge.

Lemma 39 (Truth lemma). *For every logic $\mathbf{L}_{ap}$, in the canonical model $\mathcal{M}_{\mathbf{L}_{ap}}$, for every $w \in W$ and every formula ψ over $AP \cup Y$,*

$$(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash \psi \quad \textit{iff} \quad \psi \in w$$

Proof. The proof is by induction on the length of ψ. As to the base of induction for $\psi = p$, by definition of satisfaction, $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash p$ iff $w \in V(p)$ iff $p \in w$.

For $\psi = \neg\chi$, $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash \psi$ iff $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \nvDash \chi$, iff by induction hypothesis $\chi \notin w$. Since w is maximal, this is the case iff $\psi \in w$.

For $\psi = \chi \to \chi'$, $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash \psi$ iff $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \nvDash \chi$ or $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash \chi'$. By induction hypothesis this is the case iff $\chi \notin w$ or $\chi' \in w$; in both cases we have that $\psi \in w$, as w is maximal.

Suppose that $\psi = \forall p \chi$. $\Leftarrow$ Let $\psi \in w$. By axiom $\mathbf{Ex}_{ap}$ we have that $\chi[p/q] \in w$ for every $q \in AP \cup Y$. By induction hypothesis $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash \chi[p/q]$ for every q. Now, take an arbitrary $U_q = \{w \in W \mid q \in w\}$ in the domain D of the canonical model. By Lemma 12.(2b), $((\mathcal{M}_{\mathbf{L}_{ap}})_{U_q}^p, w) \vDash \chi$, and since variant $V_{U_q}^p$ was chosen arbitrarily, we obtain that $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash \psi$. $\Rightarrow$ Assume that $\psi \notin w$. Since w is maximal, $\exists p \neg\chi \in w$, and w is Y-rich, so $\neg\chi[p/q] \in w$ for some atom $q \in Y$. Then, by induction hypothesis, $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \nvDash \chi[p/q]$, and by Lemma 12.(2b), $((\mathcal{M}_{\mathbf{L}_{ap}})_{V(q)}^p, w) \nvDash \chi$. In particular, for $U_q = V(q) \in D$, $((\mathcal{M}_{\mathbf{L}_{ap}})_{U_q}^p, w) \nvDash \chi$, i.e., $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \nvDash \psi$.

Suppose that $\psi = \Box_a \chi$. $\Leftarrow$ Assume that $\psi \in w$ and $v \in R_a(w)$. By definition of R_a, $\chi \in v$; therefore by induction hypothesis $(\mathcal{M}_{\mathbf{L}_{ap}}, v) \vDash \chi$. Thus, $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \vDash \psi$. $\Rightarrow$ Assume that $\psi \notin w$ and consider set $\{\phi \mid \Box_a \phi \in w\} \cup \{\neg\chi\}$. This set is consistent, for if not, then for some $\phi_1, \ldots, \phi_n \in \{\phi \mid \Box_a \phi \in w\}$, $\vdash \bigwedge \phi \to \chi$. Then, by axiom $\mathbf{K}$, $\vdash \bigwedge \Box_a \phi \to \Box_a \chi$ and since $\bigwedge \Box_a \phi \in w$, also $\Box_a \chi \in w$ against hypothesis. Now we want to saturate $\Delta = \{\phi \mid \Box_a \phi \in w\} \cup \{\neg\chi\}$ to obtain $v \in W$ such that $R_a(w, v)$ and $\neg\chi \in v$. However, we cannot directly apply Lemma 36 to Δ, as it is a set of formulas over $AP \cup Y$. We prove that Δ can nonetheless be extended to a saturated set Φ of formulas over $AP \cup Y$. The proof structure is similar to the one for Lemma 36, namely, we define

33

a sequence $\Phi_0, \Phi_1, \ldots$ of sets of formulas over $AP \cup Y$ as follows:

$$
\begin{aligned}
\Phi_0 \;&=\; \Delta \\[4pt]
\Phi_{n+1} \;&=\;
\begin{cases}
\Phi_n \cup \{\theta_n\} & \text{if } \Phi_n \cup \{\theta_n\} \text{ is consistent and } \theta_n \text{ is not of the form } \exists p\zeta; \\
\Phi_n \cup \{\theta_n, \zeta[p/q]\} & \text{if } \Phi_n \cup \{\theta_n\} \text{ is consistent, } \theta_n \text{ is of the form } \exists p\zeta, \\
& \text{and } q \in AP \cup Y \text{ is the first atom such that} \\
& \Phi_n \cup \{\theta_n, \zeta[p/q]\} \text{ is consistent}; \\
\Phi_n \cup \{\neg\theta_n\} & \text{otherwise.}
\end{cases}
\end{aligned}
$$

We prove by induction on n that every Φ_n is consistent (and well-defined). First of all, $\Phi_0 = \Delta$ is consistent as shown above. As to the inductive step, suppose that Φ_n is consistent. We only consider the case where $\Phi_n \cup \{\theta_n\}$ is consistent and θ_n is of the form $\exists p\zeta$, and show that Φ_{n+1} is well-defined, that is, there exists $q \in AP \cup Y$ such that $\Phi_n \cup \{\theta_n, \zeta[p/q]\}$ is consistent as well. To obtain a contradiction, suppose that for every $q \in AP \cup Y$, there exist $\Box_a \varphi_0, \ldots, \Box_a \varphi_m$ in w such that

$$
\vdash \bigwedge_{i \le m} \varphi_i \to (\bigwedge_{i \le n} \psi_i \wedge \theta_n \to \neg\zeta[p/q])
$$

where $\psi_0, \ldots, \psi_n$ are all the formulas in $\Phi_n \smallsetminus \Delta$.

By axiom **K** we obtain

$$
\vdash \bigwedge_{i \le m} \Box_a \varphi_i \to \Box_a(\bigwedge_{i \le k} \psi_i \wedge \theta_n \to \neg\zeta[p/q])
$$

and since all $\Box_a \varphi_i$ belong to w, by maximality we derive that $\Box_a(\bigwedge_{i \le k} \psi_i \wedge \theta_n \to \neg\zeta[p/q]) \in w$ for all $q \in AP \cup Y$ (*). Take an atom q not occurring in $\bigwedge_{i \le k} \psi_i \wedge \theta_n$ and consider the formula $\forall q \, \Box_a \, (\bigwedge_{i \le k} \psi_i \wedge \theta_n \to \neg\zeta[p/q])$. We claim that this formula is a member of w, because, if not, by maximality, $\exists q \Diamond_a (\bigwedge_{i \le k} \psi_i \wedge \theta_n \wedge \zeta[p/q]) \in w$, and, by saturation, for some $q' \in AP \cup Y$, we have $\Diamond_a(\bigwedge_{i \le k} \psi_i \wedge \theta_n \wedge \zeta[p/q']) \in w$, contradicting (*).

Since $\forall q \Box_a (\bigwedge_{i \le k} \psi_i \wedge \theta_n \to \neg\zeta[p/q]) \in w$ by axiom **BF** we obtain that $\Box_a \forall q(\bigwedge_{i \le k} \psi_i \wedge \theta_n \to \neg\zeta[p/q]) \in w$ as well, and since q is assumed not to appear in $\bigwedge_{i \le k} \psi_i \wedge \theta_n$ we derive $\Box_a(\bigwedge_{i \le k} \psi_i \wedge \theta_n \to \forall q \neg\zeta[p/q]) \in w$, and therefore $\bigwedge_{i \le k} \psi_i \wedge \theta_n \to \forall q \neg\zeta[p/q] \in \Delta$. Further, since $\psi_0, \ldots, \psi_n$ belong to Φ_n, we obtain that $\Phi_n \cup \{\theta_n\} \vdash \forall q \neg\zeta[p/q]$. But this contradicts the fact that $\Phi_n \cup \{\theta_n\}$ is consistent (recall that $\theta_n = \exists p\zeta$). As a result, $\Phi_n \cup \{\theta_n, \zeta[p/q]\}$ is indeed consistent for some $q \in AP \cup Y$.

The other inductive cases of the construction go as in Lemma 36. Finally, $\Phi = \bigcup_{n \in \mathbb{N}} \Phi_n$ is consistent as each Φ_n is so. In particular, Φ is a saturated set in W such that $\Phi \in R_a(w)$ by construction. By induction hypothesis $(\mathcal{M}_{\mathbf{L}_{ap}}, \Phi) \not\models \chi$, that is, $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \not\models \psi$. $\qquad\square$

By Lemma 39, if $\nvdash_{\mathbf{L}_{ap}} \neg\phi$ then there exists a saturated set $w \supseteq \{\phi\}$ such that in the canonical model $\mathcal{M}_{\mathbf{L}_{ap}}$, we have $(\mathcal{M}_{\mathbf{L}_{ap}}, w) \models \phi$. Moreover, $\mathcal{M}_{\mathbf{L}_{ap}}$ is based on a frame $\mathcal{F} \in \mathcal{K}_{all}$. Thus, $\mathcal{K}_{all} \not\models \neg\phi$. This concludes the completeness proof for $\mathbf{L}_{ap}$.

4.1.2. Completeness of $\mathbf{L}_{pl}$ and $\mathbf{L}_{ml}$

In this section we discuss how to adapt the completeness proof for $\mathbf{L}_{ap}$ in the previous section to logics $\mathbf{L}_{pl}$ and $\mathbf{L}_{ml}$. As regards $\mathbf{L}_{pl}$, we need to modify the definition of the canonical model and the proof of the truth lemma, starting with the former.

Definition 40 (Canonical Model). *The canonical model for* $\mathbf{L}_{pl}$ *is a tuple* $\mathcal{M}_{\mathbf{L}_{pl}} = \langle W, D, R, V \rangle$ *where*

- *W, R and V are given as in Definition 37;*

- *D is the domain of sets $U_\psi = \{w \in W \mid \psi \in w\} \subseteq W$, for every propositional formula $\psi \in \mathcal{L}_{pl}$ over $AP \cup Y$.*

Given Definition 40 of canonical model, we can show that it is indeed based on a boolean frame.

Lemma 41. *The canonical model $\mathcal{M}_{\mathbf{L}_{pl}}$ is boolean.*

Proof. We have to prove that domain D is closed under boolean operations. Let U_ϕ and $U_{\phi'}$ be sets in D, we show that $U_\phi \cap U_{\phi'} = U_{\phi \wedge \phi'} \in D$. Clearly, $w \in U_\phi \cap U_{\phi'}$ iff $\phi \in w$ and $\phi' \in w$, and by maximality, this is the case iff $\phi \wedge \phi' \in w$ as well. Closure under disjunction is proved similarly. As to taking complement, we show that $W \smallsetminus U_\phi = U_{\neg\phi} \in D$. Again, $w \notin U_\phi$ iff $\phi \notin w$, and by maximality, this is the case iff $\neg\phi \in w$. $\qquad\square$

As a consequence of Lemma 41, $\mathcal{M}_{\mathbf{L}_{pl}}$ is based on a frame in $\mathcal{K}_{bool}$. Moreover, we are able to prove the following version of the truth lemma.

Lemma 42 (Truth lemma). *For every logic $\mathbf{L}_{pl}$, in the canonical model $\mathcal{M}_{\mathbf{L}_{pl}}$, for every $w \in W$ and every formula ψ over $AP \cup Y$,*

$$(\mathcal{M}_{\mathbf{L}_{pl}}, w) \vDash \psi \quad \textit{iff} \quad \psi \in w$$

Proof. To prove the truth lemma for $\mathbf{L}_{pl}$ we have to modify the proof of Lemma 39. Specifically, we can first prove that for every $\psi \in \mathcal{L}_{pl}$, $(\mathcal{M}_{\mathbf{L}_{pl}}, w) \vDash \psi$ iff $\psi \in w$, as in Lemma 39. In particular, we obtain that $U_\varphi = [\![\varphi]\!]_{\mathcal{M}_{\mathbf{L}_{pl}}} = \{w \in W \mid (\mathcal{M}_{\mathbf{L}_{pl}}, w) \vDash \varphi\} \in D$. Then, the proof for any $\psi \in \mathcal{L}_{sopml}$ is given by induction on the length of ψ, similarly as in Lemma 39, the only case of interest being quantified formulas. For $\psi = \forall p\chi$, if $\psi \in w$ then by axiom $\mathbf{Ex}_{pl}$ we have that $\chi[p/\varphi] \in w$, for any $\phi \in \mathcal{L}_{pl}$. By induction hypothesis, $(\mathcal{M}_{\mathbf{L}_{pl}}, w) \vDash \chi[p/\varphi]$. Now consider the set $U_\varphi = [\![\varphi]\!]_{\mathcal{M}_{\mathbf{L}_{pl}}} \in D$. By Lemma 12.(2b), it is the case that $((\mathcal{M}_{\mathbf{L}_{pl}})^p_{U_\varphi}, w) \vDash \chi$, and since the choice of φ (and therefore of variant $V^p_{U_\varphi}$) is arbitrary, we obtain that $(\mathcal{M}_{\mathbf{L}_{pl}}, w) \vDash \forall p\chi$. As to the implication from left to right, the proof is the same as in Lemma 39, as each w is maximal and rich. $\qquad\square$

As a consequence of Lemma 42, the truth lemma also holds for boolean frames and we obtain a completeness proof for $\mathbf{L}_{pl}$.

We now discuss how to modify the procedure above to obtain completeness results for the logics $\mathbf{L}_{ml}$. Firstly, the canonical model for $\mathbf{L}_{ml}$ is now defined as a tuple $\mathcal{M}_{\mathbf{L}_{pl}} = \langle W, D, R, V \rangle$ where *(i)* W, R and V are given as in Definition 37; and *(ii)* D is the domain of sets $U_\psi = \{w \in W \mid \psi \in w\} \subseteq W$, for every modal formula $\psi \in \mathcal{L}_{ml}$ over $AP \cup Y$. In particular, it is easy to check that the domain D in $\mathcal{M}_{\mathbf{L}_{pl}}$ is a boolean algebra with operators. Secondly, by adapting the proof of Lemma 42, we can prove the truth lemma for $\mathcal{M}_{\mathbf{L}_{ml}}$:

Lemma 43 (Truth lemma). *For every logic $\mathbf{L}_{ml}$, in the canonical model $\mathcal{M}_{\mathbf{L}_{ml}}$, for every $w \in W$ and every formula ψ over $AP \cup Y$,*

$$(\mathcal{M}_{\mathbf{L}_{ml}}, w) \vDash \psi \quad \textit{iff} \quad \psi \in w$$

35

This completes the completeness proof for the logics $\mathbf{L}_{ml}$.

We conclude the section by summarising the soundness and completeness results for our logics w.r.t. the relevant classes of frames.

Theorem 44 (Soundness and Completeness). *For $x \in \{ap, pl, ml\}$, each logic $\mathbf{L}_x$ is sound and complete w.r.t. the class $\mathcal{K}_{\widehat{x}}^{\tau(\mathbf{L}_x)}$ of frames that are reflexive (respectively, transitive, symmetric), whenever $\mathbf{L}_x$ includes axiom $\mathbf{T}$ (respectively, $\mathbf{4}$, $\mathbf{B}$).*

As a result, for types ap, pl, and ml we are able to prove soundness and completeness for all normal modalities (i.e., $\mathbf{K}, \mathbf{T}, \mathbf{S4}, \mathbf{B}$, and $\mathbf{S5}$) in a multi-modal setting.

4.1.3. Generalised Completeness

We now extend the completeness results in the previous section by considering extra axioms expressing properties of frames. Specifically, let $\mathbf{L}$ be any axiomatisation mentioned in Theorem 44. Then, if we extend $\mathbf{L}$ with the universal closure $\forall \vec{p} \psi$ of a formula $\psi \in \mathcal{L}_{sopml}$, the resulting calculus $\mathbf{L} + \forall \vec{p} \psi$ is sound and complete w.r.t. the class of frames satisfying the MSO condition $\forall x \forall \vec{P} ST_x(\psi)$, where $\vec{P}$ are all the unary predicates appearing in $ST_x(\psi)$.

Theorem 45. *Let ψ be a formula in SOPML, then the logic $\mathbf{L} + \forall \vec{p} \psi$ is sound and complete w.r.t. the corresponding class $\mathcal{K}$ of frames satisfying $\forall x \forall \vec{P} ST_x(\psi)$.*

Proof. Soundness follows immediately by Lemma 29, as every frame that satisfies condition $\forall x \forall \vec{P} ST_x(\psi) = \forall x ST_x(\forall \vec{p} \psi)$, also validates $\forall \vec{p} \psi$. As to completeness, if $\forall \vec{p} \psi$ is an axiom, then it appears in every state of the canonical model $\mathcal{M}$, and by the truth lemma, $\mathcal{M}$ validates $\forall \vec{p} \psi$. Finally, by another application of Lemma 29, $\mathcal{F}$ validates $\forall x \forall \vec{P} ST_x(\psi)$. $\qquad\square$

By the result above we immediately obtain that for every formula $\theta(\vec{a}, \vec{p})$ appearing in Table 1, $\mathbf{L} + \forall p \theta(\vec{a}, \vec{p})$ is sound and complete w.r.t. the class of frames satisfying $\forall x \forall \vec{P} \Theta(\vec{a}, x)$. For instance, $\mathbf{K} + \exists p(\Box_a p \wedge \neg p)$ is a sound and complete axiomatisation of the class of irreflexive frames (notice that, since $\exists p(\Box_a p \wedge \neg p)$ is a sentence, its universal closure is equal to the formula itself.) More generally, there is a one-to-one correspondence between a SOPML axiom $\forall \vec{p} \theta$ and the MSO condition $\forall x \forall \vec{P} ST_x(\theta)$ on the corresponding class of sound and complete frames.

4.2. Incompleteness

In Section 4.1 we provided complete axiomatisations for languages without common knowledge, in the classes of all, boolean, and modal frames. In this section we prove our main unaxiomatisability results for SOPML. Specifically, in Section 4.2.1 we show that the set of validities in SOPEL (without common knowledge) interpreted on full frames is unaxiomatisable whenever we assume at least two agents in our frames. Further, in Section 4.2.2 we prove that, with common knowledge, SOPML is unaxiomatisable on modal frames already when considering a single agent. We then demonstrate (Corollaries 59, 62 and 64 as well as Theorem 63) how this proof can be extended to the other classes of frames. These results complete Tables 3 and 4.

₁₁₀₀ *4.2.1. 2-agent* SOPEL *on Full Frames is Unaxiomatisable*

Recall that, as discussed at the beginning of Section 4, an axiomatisation for single-agent SOPEL on full frames was introduced in [22]. Here we show that, this result cannot be generalised to the multi-agent case, i.e., we demonstrate that multi-agent SOPEL is not recursively axiomatisable on full frames.

₁₁₀₅ We prove this unaxiomatisability result by reducing the validity problem of diadic second order logic (DSO) to the validity problem of SOPEL. Since the former is known not to be recursively enumerable, this implies that the latter is also not recursively enumerable, and therefore in particular not axiomatisable. The reduction that we use is somewhat similar to the one used in [40] to prove the unaxiomatisability of single-agent

₁₁₁₀ SOPML on full frames with S4.2 or weaker modalities. Specifically, both the proof from [40] and the proof presented here represent a second order domain $\mathcal{D}$ in a Kripke model by taking $\mathcal{D} \subseteq W$ and $(\mathcal{D} \times \mathcal{D}) \subseteq W$. Quantification over diadic relations on *Dom* then corresponds to propositional (i.e., monadic) quantification over $(\mathcal{D} \times \mathcal{D})$. The difference between the two proofs lies in how they characterize the models where $\mathcal{D} \subseteq W$ and

₁₁₁₅ $\mathcal{D} \times \mathcal{D} \subseteq W$, and in how second order formulas are translated once such a characterisation has been established.

Since the reduction is from the validity problem of diadic second order logic, let us first briefly define this logic.

Let a set $\mathcal{X}$ of first-order variables and a set $\mathcal{R}$ of second order variables be given. Then the language of diadic second order logic is given by the following normal form:

$$\theta ::= R(x,y) \mid x = y \mid \neg\theta \mid \theta \to \theta \mid \forall x\theta \mid \forall R\theta$$

where $x, x \in \mathcal{X}$ and $R \in \mathcal{R}$.

₁₁₂₀ The formulas of DSO can be evaluated on models (Dom, ρ) that consist of a domain *Dom* and an assignment function ρ that assigns to each first-order variable x an element $\rho(x) \in Dom$ and to each second-order variable R a relation $\rho(R) \subseteq Dom \times Dom$. Given a model (Dom, ρ) an element $d \in Dom$ and a relation $E \subseteq Dom \times Dom$, the assignments $\rho[x \mapsto d]$ and $\rho[R \mapsto E]$ are the modifications of ρ that map $x \in \mathcal{X}$ to $d \in Dom$ and $R \in \mathcal{R}$

₁₁₂₅ to $E \subseteq Dom \times Dom$, respectively.

Given these preliminaries, we can define the semantics of DSO in the usual way.

Definition 46. *We define whether a model* (Dom, ρ) *satisfies a formula* θ *of* DSO *recursively as follows:*

$(Dom, \rho) \vDash R(x,y)$	*iff*	$(\rho(x), \rho(y)) \in \rho(R)$
$(Dom, \rho) \vDash x = y$	*iff*	$\rho(x) = \rho(y)$
$(Dom, \rho) \vDash \neg\theta$	*iff*	$(Dom, \rho) \nvDash \theta$
$(Dom, \rho) \vDash \theta \to \theta'$	*iff*	$(Dom, \rho) \nvDash \theta$ *or* $(Dom, \rho) \vDash \theta'$
$(Dom, \rho) \vDash \forall x\theta$	*iff*	*for every* $d \in Dom$, $(Dom, \rho[x \mapsto d]) \vDash \theta$
$(Dom, \rho) \vDash \forall R\theta$	*iff*	*for every* $E \subseteq Dom \times Dom$, $(D, \rho[R \mapsto E]) \vDash \theta$

A formula θ *of* DSO *is valid, denoted* $\vDash \theta$, *if* $(Dom, \rho) \vDash \theta$ *for every model* (Dom, ρ).

₁₁₃₀ In general, a formula of DSO can contain free first- and second-order variables. Our goal is to make a reduction from the validity problem of DSO, however, so we only care about whether a formula is valid. If θ is a DSO formula with a free first-order variable x (a free second-order variable R, respectively), then θ is valid if and only if $\forall x\theta$ ($\forall R\theta$,

37

respectively) is valid. As a result, it suffices for us to consider only the sentences of DSO,
i.e., the formulas without free variables. Furthermore, if θ is a sentence then whether
$(Dom, \rho) \vDash \theta$ depends only on Dom. As such, we can consider our models to be given by
the domain Dom only, where we say that $Dom \vDash \theta$ if and only if $(Dom, \rho) \vDash \theta$ for every
assignment ρ.

Now, let us introduce the reduction from the validity problem of DSO to SOPEL's.
This reduction has two parts: firstly, we define a formula ψ_{model} of SOPEL and use it to
characterize a specific class of pointed models. Then, we define a translation f from the
formulas of DSO to the formulas of SOPEL, with the property that

> for every formula θ in DSO, there is a model Dom such that $Dom \vDash \theta$ iff there
> is a pointed model $(\mathcal{M}, w)$ of SOPML such that $(\mathcal{M}, w) \vDash \psi_{model} \wedge f(\theta)$.

Before defining ψ_{model} and $f(\phi)$, however, let us present an auxiliary formula $\psi_{unique}(a, \chi)$
that will be useful in several places.

Definition 47. *Let $a \in I$ be an index and $\chi \in \mathcal{L}_{sopml}$ a formula. Then*

$$\psi_{unique}(a, \chi) := \Diamond_a \chi \wedge \forall q(\Diamond_a(q \wedge \chi) \to \Box_a(\chi \to q)).$$

Lemma 48. *Let a be an index in I, $\chi \in \mathcal{L}_{sopml}$ any formula, $\mathcal{M}$ any full epistemic model,
and w any world in $\mathcal{M}$. Then $(\mathcal{M}, w) \vDash \psi_{unique}(a, \chi)$ if and only if there is exactly one
a-successor w' of w such that $(\mathcal{M}, w') \vDash \chi$.*

Proof. Suppose $(\mathcal{M}, w) \vDash \psi_{unique}(a, \chi)$. Then, in particular, $(\mathcal{M}, w) \vDash \Diamond_a \chi$, so there
is at least one a-successor w' of w such that $(\mathcal{M}, w') \vDash \chi$. Suppose now, towards a
contradiction, that there are two different a-successors w' and w'' of w such that χ holds
on both w' and w''. Then, since $\mathcal{M}$ is a full model, there is some assignment for q such
that q holds in w' but not in w''. As a result, for this choice of q, we have $(\mathcal{M}, w) \vDash$
$\Diamond_a(q \wedge \chi) \wedge \neg \Box_a(\chi \to q)$, contradicting the fact that $(\mathcal{M}, w) \vDash \forall q(\Diamond_a(q \wedge \chi) \to \Box_a(\chi \to q))$.
It follows that the assumption of such w' and w'' existing must be false, so there is at
most one a-successor w' of w such that $(\mathcal{M}, w') \vDash \chi$.

Suppose then that there is exactly one a-successor w' of w such that $(\mathcal{M}, w') \vDash \chi$.
Then there is at least one such successor, so $(\mathcal{M}, w) \vDash \Diamond_a \chi$. Furthermore, for every
assignment of q, we have $\Diamond_a(q \wedge \chi)$ if and only if $(\mathcal{M}, w') \vDash q \wedge \chi$, in which case we also
have $\Box_a(\chi \to q)$. It follows that $(\mathcal{M}, w) \vDash \forall q(\Diamond_a(q \wedge \chi) \to \Box_a(\chi \to q))$. Together with
the previously established $(\mathcal{M}, w) \vDash \Diamond_a \chi$, this implies that $(\mathcal{M}, w) \vDash \psi_{unique}(a, \chi)$. $\square$

Now we can use ψ_{unique} to define ψ_{model} and f.

Definition 49. *The formula ψ_{model} is given by*

$$\psi_{model} := \Box_a (\psi_D \wedge \psi_{excl} \wedge \psi_{connect} \wedge \psi_{\geq 1} \wedge \psi_{\leq 1})$$

where

$$\psi_D := p_D \wedge \psi_{unique}(b, p_D)$$
$$\psi_{excl} := \Box_b \neg(p_D \wedge (start \vee end)) \wedge \Box_b \neg(start \wedge end) \wedge \Box_b(p_D \vee start \vee end)$$
$$\psi_{connect} := \Box_b (\neg p_D \to (\Box_a(start \vee end) \wedge \psi_{unique}(a, start) \wedge \psi_{unique}(a, end)))$$
$$\psi_{\geq 1} := \forall p(\Diamond_a p \to \Diamond_b(p_{start} \wedge \Diamond_a(p_{end} \wedge \Diamond_b p)))$$
$$\psi_{\leq 1} := \forall p(\psi_{unique}(a, p) \to \psi_{unique}(b, start \wedge \Diamond_a(end \wedge \Diamond_b(p \wedge p_D))))$$

38

The meaning of the named subformulas $(\psi_D, \psi_{excl}, \psi_{connect}, \psi_{\geq 1}$ and $\psi_{\leq 1})$ is discussed in the proof of Theorem 51.

Definition 50. *Let translation function f is recursively defined as follows.*

$$
\begin{aligned}
f(x = y) &= \Diamond_a(p_x \wedge p_y) \\
f(R(x,y)) &= \Diamond_a(p_x \wedge \Diamond_b(p_R \wedge start \wedge \Diamond_a(end \wedge \Diamond_b p_y))) \\
f(\neg\theta) &= \neg f(\theta) \\
f(\theta_1 \rightarrow \theta_2) &= f(\theta_1) \rightarrow f(\theta_2) \\
f(\forall x\theta) &= \forall p_x(\psi_{unique}(a, p_x) \rightarrow f(\theta)) \\
f(\forall R\theta) &= \forall p_R f(\theta)
\end{aligned}
$$

We can now prove the main result of this section.

Theorem 51. *For every* DSO *sentence θ,*

$$\vDash \theta \quad iff \quad \mathcal{K}^e_{full} \vDash \psi_{model} \rightarrow f(\theta)$$

Proof. First, let us consider the models that satisfy ψ_{model}. Let $(\mathcal{M}, w_0)$ be any full epistemic pointed model such that $(\mathcal{M}, w_0) \vDash \psi_{model}$, and let $Dom := R_a(w_0)$. The primary connective of ψ_{model} is $\Box_a$, so let us consider any $w \in Dom$.

The conjunct ψ_D holds at w if and only if (i) w satisfies p_D and (ii) w has only one b-successor that satisfies p_D (and by (i) this unique successor has to be w itself). We refer to the b-successors of w as the *cone* on w, as drawn in the following figure. Note that cones of any two different $w, w' \in Dom$ cannot overlap, since by the fact that $\mathcal{M}$ is an epistemic model, this would imply that $R_b(w, w')$, contradicting the uniqueness of the p_D-world.
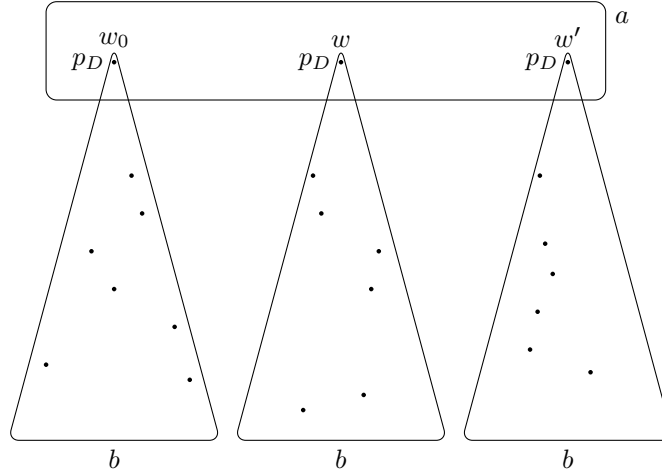
The conjunct ψ_{excl} holds if and only if every state in a cone satisfies exactly one of p_D, *start* and *end*. The formula $\psi_{connect}$ then holds if and only if all *start* states are paired with exactly one *end* state by the relation a, and likewise all *end* states are paired with exactly one *start* state. Note that if a *start* or *end* state is in the cone of w, then

39

the state that it is paired up with could be in the cone of w, in the cone of some different state w' or it might not be in any cone at all.

Now, consider formula $\psi_{\geq 1}$. It states that for every choice of p, if w has an a-successor w' that satisfies p, then there is a path $w \overset{b}{\longmapsto} e_1 \overset{a}{\longmapsto} e_2 \overset{b}{\longmapsto} w''$ such that e_1 satisfies $start$, e_2 satisfies end and w'' satisfies p. This is the case if and only if for every $w, w' \in D$, there is at least one such path $w \overset{b}{\longmapsto} e_1 \overset{a}{\longmapsto} e_2 \overset{b}{\longmapsto} w'$. Note that this also applies for $w = w'$. Our schematic illustration can therefore be extended to the following:
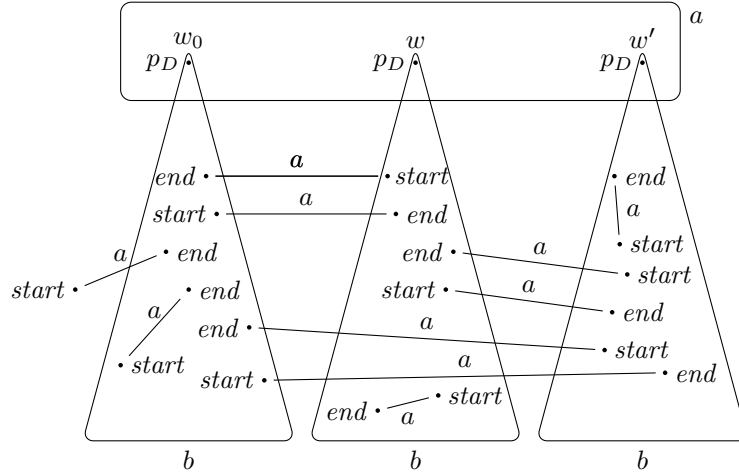
Finally, formula $\psi_{\leq 1}$ states that for any choice of p, if w has a unique a-successor w' that satisfies p, then w also has a unique b-successor e_1 such that (i) e_1 satisfies $start$, (ii) e_1 has an a-successor e_2 that satisfies end, and (iii) e_2 has a b-successor w'' that satisfies $p \wedge p_D$. This is the case if and only if for every $w, w' \in D$ there is at most one path $w \overset{b}{\longmapsto} e_1 \overset{a}{\longmapsto} e_2 \overset{b}{\longmapsto} w'$ where e_1 satisfies $start$ and e_2 satisfies end.

All in all, the conjuction of $\psi_{\geq 1}$ and $\psi_{\leq 1}$ means that there is exactly one path $w \overset{b}{\longmapsto} e_1 \overset{a}{\longmapsto} e_2 \overset{b}{\longmapsto} w'$ with $(\mathcal{M}, e_1) \vDash start$ and $(\mathcal{M}, e_2) \vDash end$ between each $w, w' \in D$.

The pointed model $(\mathcal{M}, w_0)$ is then used in the following way. The suggestively named set Dom can be treated as the domain of a DSO model. First-order quantification is then interpreted on selecting exactly one world $w \in Dom$. Note that this is precisely what translation function f does: first-order quantification $\forall x \theta$ is translated as $\forall p_x (\psi_{unique}(a, p_x) \to f(\theta))$. On the other hand, second-order quantification is interpreted on selecting exactly those $start$ worlds that are on a path $w \overset{b}{\longmapsto} e_1 \overset{a}{\longmapsto} e_2 \overset{b}{\longmapsto} w'$, where pair (w, w') is in the chosen relation. Again, this is exactly what translation f does: $\forall R \theta$ is translated as $\forall p_R f(\theta)$, and the relational atom $R(x, y)$ is then translated as $\Diamond_a(p_x \wedge \Diamond_b(p_R \wedge start \wedge \Diamond_a(end \wedge \Diamond_b p_y)))$, which means exactly that $R(x, y)$ holds iff p_R is true in the $start$-world in between the unique p_x- and p_y-worlds.

It follows that $Dom \vDash \theta$ if and only if $(\mathcal{M}, w_0) \vDash f(\theta)$. Since all we assumed about $\mathcal{M}$ is that it is a full epistemic model and that $(\mathcal{M}, w_0) \vDash \psi_{model}$, it follows that $\vDash \theta$ if and only if $\mathcal{K}^e_{full} \vDash \psi_{model} \to f(\theta)$. $\qquad\qquad\square$

As a consequence of the fact that validity in DSO is not recursively enumerable, we immediately obtain the following corollary.

Corollary 52. *Multi-agent* SOPML *on full epistemic frames is unaxiomatisable.*

Remark 53. *We used only two indexes for modalities in our reduction, so the unaxiomatisability result holds whenever the set of indexes contains at least two elements.*

4.2.2. Single-agent SOPML* *is Unaxiomatisable on Modal Frames*

In Section 4.1.3 we introduced sound and complete axiomatisations for SOPML on the classes of *modal*, *boolean*, and *all* frames. The version of SOPML that we considered there does not contain common knowledge, however. This is a fundamental restriction; the variant SOPML*, which is obtained by adding common knowledge to SOPML, is not axiomatisable on any class of frames. Here, we show that SOPML* is not axiomatisable with respect to general or epistemic models that have *modal*, *boolean* or *arbitrary* domain of quantification. It can also be shown that SOPML* is unaxiomatisable on intermediate classes of models, such as those for logics **KD45** (i.e., serial, transitive and symmetric models) or **S4.2** (i.e., reflexive, transitive and convergent models). We do not include proofs for these classes, however, since they are very similar to the proofs for the cases we treat.

We prove the unaxiomatisability of SOPML* on *modal* models by a reduction from the non-halting problem to the validity problem of SOPML*. Since the non-halting Turing machines are not recursively enumerable, neither are the validities of SOPML* with respect to *modal* models. The proof that we use here is inspired by a similar proof from [42], in which the non-halting problem is reduced to the validity problem of a logic called *Arbitrary Arrow Update Logic with Common Knowledge*.

Before defining our reduction, let us first define the non-halting problem for Turing machines. A Turing machine, first defined in [54], is an abstract model of computation that consists of (i) an infinite tape, with $\mathbb{Z}$ cells that can each contain a symbol, but that initially contain a "null" symbol α_0; (ii) a read/write head that can read the symbol in a cell or write a symbol to it; (iii) a method to keep track of what *state* the machine is currently in; and (iv) a set of instructions that, based on the current state and the current symbol under the read/write head, determines what symbol is to be written, what the next state of the machine should be, and whether the read/write head should move to the left, to the right or remain in place. The Turing machines that we consider here are deterministic so for each combination of state and symbol there is exactly one instruction. Formally, a Turing machine T can be defined as follows.

Definition 54. *A Turing machine T is a triple $T = (\Lambda, S, \Delta)$, where Λ is a finite alphabet containing the symbol α_0, S is a finite set of states containing the distinct states s_0 and s_{end}, and $\Delta : \Lambda \times S \to \Lambda \times S \times \{left, right, remain\}$ is a transition function.*

The functions Δ_1, Δ_2 and Δ_3 are the projections of Δ to its first, second and third coordinate, respectively.

We can assume without loss of generality that $\Delta_2(\alpha, s) \neq s_0$ for all $\alpha \in \Lambda$ and $s \in S$, so that the machine never returns to its initial state.[5] The execution of such a machine T can be represented as a function $E_T : \mathbb{Z} \times \mathbb{N} \to \Lambda \times S \times \{yes, no\}$. The horizontal

---

[5]Where necessary, an extra copy s_0^{copy} of s_0 can be added; whenever the system would go to state s_0 we can then let it go to s_0^{copy} instead.

direction $\mathbb{Z}$ represents positions on the tape. The vertical direction $\mathbb{N}$ represents time. So $E_T(n,m) = (\alpha, s, yes)$ indicates that at time m, the n-th position on the tape contains symbol α, the machine is in state s and the read/write head is in position n. Likewise, $E_T(n,m) = (\alpha, s, no)$ indicates that at time m the read/write head is not at position n. Note that E_T contains some redundant information: the state is a property of the machine, not of any particular position on the tape. Furthermore, the read/write head is in exactly one position at a time. So, for example, if $E_T(n,m) = (\alpha, s, yes)$, then we must have $E_T(n+1,m) = (\beta, s, no)$ for some symbol β.

A Turing machine T has a designated end state s_{end}. We say that T *halts* if the machine ever reaches this end state, so if for any (n,m) and any $\alpha \in \Lambda$, we have $E_T(n,m) = (\alpha, s_{end}, yes)$.

Definition 55 (Halting Problem). *The* halting problem *is to determine, for a given Turing machine T, whether T halts. The* non-halting problem *is the complement of the halting problem, i.e., to determine whether a given Turing machine doesn't halt.*

Although neither of them used this terminology, Church [14] and Turing [54] independently showed that the halting problem is undecidable. More precisely, the halting Turing machines are recursively enumerable, but the non-halting Turing machines are not.

Remark 56. *The intuition behind the end state is that the machine stops when it reaches s_{end}, so no more computation happens after that point. Here, however, in order to avoid special cases it is more convenient to assume that every Turing machine keeps going forever, whether or not it reaches s_{end}. So, whether or not a machine formally halts (reaches the end state), it never informally halts (stops).*

When reducing the non-halting problem to validity in SOPML*, it is useful to represent time not by the naturals $\mathbb{N}$ but by the integers $\mathbb{Z}$, since this allows us to avoid special cases at time $t = 0$. For every time $t < 0$ we then say that the machine is in a dummy state $s_{void} \notin S$, which can be read as "the computation has not started yet."

Now that the necessary concepts are defined, we can give our reduction of the non-halting problem to the validity problem of SOPML* on *modal* frames. This reduction has three parts. Firstly, we define a SOPML* formula ψ_{grid} that holds in a pointed model $(\mathcal{M}, w)$ if and only if $\mathcal{M}$ represents a $\mathbb{Z} \times \mathbb{Z}$ grid. Then, we define a formula ψ_{sane} that enforces a few sanity constraints that are necessary in order for us to interpret the grid as representing the execution of some Turing machine. Finally, we define a formula ψ_T that holds if and only if the Turing machine encoded by the grid is the specific machine T. Before defining these formulas, however, let us explain a "trick" that we will use. Typically, if we wanted to have a $\mathbb{Z} \times \mathbb{Z}$ grid in a modal logic, we would use four accessibility relations $R_{right}, R_{left}, R_{up}$ and R_{down}. However, we also want to show that even single-agent SOPML* is unaxiomatisable. So we will use only a single relation R.

We still need to represent the four possible directions, however. We do this by "coloring" the worlds of our model with propositional atoms $\{1, \cdots, 9\}$, in a repeating pattern

$$
\begin{array}{ccc}
7 & 8 & 9 \\
4 & 5 & 6 \\
1 & 2 & 3
\end{array}
$$

42

So if a world satisfies 5, then the world above it satisfies 8, and if a world satisfies 6 then the world to its right satisfies 4. For ever direction $x \in dir := \{left, right, up, down\}$, let $f_x : \{1, \cdots, 9\} \to \{1, \cdots, 9\}$ be the function that gives the next number in direction x, e.g. $f_{up}(8) = 2$. Further, since we use only one agent a, we write $\Box$ for $\Box_a$. This operator, together with functions f_x, can then be used to define $\Box_x$ and $\Diamond_x$ for $x \in dir$ as abbreviations:

$$\Box_x \phi := \bigvee_{1 \leq i \leq 9} (i \wedge \Box(f_x(i) \to \phi))$$

$$\Diamond_x \phi := \bigvee_{1 \leq i \leq 9} (i \wedge \Diamond(f_x(i) \wedge \phi))$$

1290   These abbreviations are then used in formulas ψ_{grid}, ψ_{sane} and ψ_T defined as follows.

Definition 57. *Let $S, \Lambda, \{pos, rpos, rpos\} \subseteq AP$, and let $T = (\Lambda, S, \Delta)$ be a Turing machine. The formula χ_T is given as $\chi_T := \psi_{grid} \wedge \psi_{sane} \wedge \psi_T \wedge s_0 \wedge pos$, where formulas ψ_{grid}, ψ_{sane} and ψ_T are as shown in Tables 5–7.*

The formulas ψ_{grid}, ψ_{sane} and ψ_T may look complex, but that it slightly misleading.
1295   While these formulas are certainly long, every named subformula encodes a property of either a grid or a Turing machine in a rather straightforward way.

We start by considering the formula ψ_{grid} which, as the name implies, encodes a grid. Specifically, let $(\mathcal{M}, w_0)$ be any pointed model (based on a modal frame) such that $(\mathcal{M}, w_0) \vDash \psi_{grid}$. We use the worlds of $\mathcal{M}$ to represent the points of $\mathbb{Z} \times \mathbb{Z}$, in the
1300   following way: (i) the world w_0 represents $(0,0)$; (ii) if a world w represents (n,m), then every *right*-successor of w represents $(n+1, m)$; and (iii) similarly for the other three directions.

We show that every point (n,m) is represented by at least one world, that every world in the generated submodel represents at least one point, and that all the worlds
1305   that represent a single point (n,m) are modally indistinguishable. Let w be any world in the generated submodel of $(\mathcal{M}, w_0)$. The main connective of ψ_{grid} is $\Box^*$, so *labels*, *direction*, *unique$_x$* for every $x \in dir$, *inverse$_{xy}$* for every $(x,y) \in inv\_dir$ and *commute$_{xy}$* for all $(x,y) \in perp\_dir$ hold at w.

- By *labels*, the world w satisfies exactly one label $i \in \{1, \cdots, 9\}$.

1310
- By the first conjunct of *direction*, the world w has at least one x-successor for every $x \in dir$. It follows that every point in $\mathbb{Z} \times \mathbb{Z}$ is represented by at least one world. Furthermore, by the second conjunct of *direction*, every successor of w satisfies one of the labels that allow us to identify it as an x-successor for some $x \in dir$. It follows that every world in the generated submodel represents at least one point.

1315
- The formula *unique$_x$* implies that, for every U in the domain of quantification D, either every x-successor of w is in U, or no x-successor is in U. Since $\mathcal{M}$ is based on a modal frame, this implies that all x-successors of w are modally indistinguishable from one another.

- The formula *inverse$_{xy}$* implies that, for every $U \in D$, if $w \in U$ then every xy-
1320   successor of w is also in U. Again, since $\mathcal{M}$ is based on a modal frame, this implies that w is modally indistinguishable from its xy-successors. Note that this formula holds for all $(x,y) \in inv\_dir$.

43

$$\psi_{grid} := \Box^* \left(labels \wedge direction \wedge \bigwedge_{x \in dir} unique_x \wedge \right.$$

$$\left. \bigwedge_{(x,y) \in inv\_dir} inverse_{xy} \wedge \bigwedge_{(x,y) \in perp\_dir} commute_{xy} \right)$$

$$labels := \bigvee_{1 \leq j \leq 9} i \wedge \bigwedge_{i \neq j} \neg (i \wedge j)$$

$$direction := \bigwedge_{x \in dir} \Diamond_x \top \wedge \bigvee_{1 \leq i \leq 9} (i \wedge \Box \bigvee_{x \in dir} f_x(i))$$

$$unique_x := \forall p (\Diamond_x p \rightarrow \Box_x p))$$

$$inverse_{xy} := \forall p (p \rightarrow \Box_x \Box_y p)$$

$$commute_{xy} := \forall p (\Diamond_x \Diamond_y p \rightarrow \Box_y \Box_x p)$$

$$dir := \{ left, right, up, down \}$$

$$inv\_dir := \{ (left, right), (right, left), (up, down), (down, up) \}$$

$$perp\_dir := \{ (left, up), (left, down), (right, up), (right, down),$$

$$(up, left), (up, right), (down, left), (down, right) \}$$

Table 5: The formula ψ_{grid}.

- The formula $commute_{xy}$ implies that, for every $U \in D$, if there is an xy-successor w' of w such that $w' \in U$, then every yx-successor of w is in U. This implies that the xy-successors of w are modally indistinguishable from its yx-successors.

- Taken together, $unique_x$ (for every $x \in dir$), $inverse_{xy}$ (for every $(x,y) \in inv\_dir$) and $commute_{xy}$ (for every $(x,y) \in perp\_dir$) imply that all worlds representing a single point (n, m) are modally indistinguishable from one another.

We say that tape position n at time m contains the symbol α if the worlds representing (n, m) satisfy the propositional atom α. Likewise, if the worlds representing (n, m) satisfy the propositional atom s, then the machine is in state s at time m. Finally, if the worlds representing (n, m) satisfy the atom pos, then the read/write head is in position n at time m, if they satisfy $rpos$ then position n is to the right of the read/write head at time m and if they satisfy $lpos$ then position n is to the left of the head.

Note that, because all worlds representing (n, m) are modally indistinguishable, it does not matter at which world we check whether these atoms are true; if s holds in any world representing (n, m) then s holds on all worlds representing (n, m). In particular, since we use propositional atoms to represent the state of the Turing machine, the symbols on the tape, as well as the position of the read/write head, the modal indistinguishability of all worlds representing (n, m) means that all those worlds agree on the symbol, state, and on whether the read/write head is in that position. If a modal formula ϕ holds on the worlds that represent (n, m), we abuse notation by writing $(\mathcal{M}, (n, m)) \vDash \phi$.[6]

---

[6]Note that we have not shown that w represents a *unique* point (n, m). It is, in fact, true that if

$$\psi_{sane} \coloneqq \square^* \, (position_1 \wedge position_2 \wedge one\_state \wedge same\_state \wedge one\_symbol \wedge$$
$$void\_state \wedge initial\_symbol \wedge unchanged_i)$$
$$position_1 \coloneqq \neg(pos \wedge lpos) \wedge \neg(pos \wedge rpos) \wedge \neg(rpos \wedge lpos)$$
$$position_2 \coloneqq ((pos \vee rpos) \to \square_{right} rpos) \wedge ((pos \vee lpos) \to \square_{left} lpos)$$
$$one\_state \coloneqq \bigvee_{s \in states} (s \wedge \bigwedge_{s' \in states \setminus \{s\}} \neg s')$$
$$same\_state \coloneqq \bigwedge_{s \in states} (s \to (\square_{right} s \wedge \square_{left} s))$$
$$one\_symbol \coloneqq \bigvee_{\alpha \in \Lambda} (\alpha \wedge \bigwedge_{\beta \in \Lambda \setminus \{\alpha\}} \neg\beta)$$
$$void\_state \coloneqq (s_0 \vee s_{void}) \to \square_{down} s_{void}$$
$$initial\_symbol \coloneqq s_0 \to \alpha_0$$
$$unchanged \coloneqq \bigwedge_{\alpha \in \Lambda} ((\neg pos \wedge \alpha) \to \square_{up} \alpha)$$

Table 6: The formula ψ_{sane}.

The formula ψ_{sane} imposes a number of sanity constraints: if $(\mathcal{M}, w)$ satisfies both ψ_{grid} and ψ_{sane}, then $\mathcal{M}$ can *almost* be seen as the execution of a Turing machine. We will first discuss the subformulas of ψ_{sane} in detail, and explain which sanity constraints they represent. After that, we will briefly discuss why ψ_{sane} only guarantees that $\mathcal{M}$ can *almost* be seen as the execution of a Turing machine.

As with ψ_{grid}, the main connective of ψ_{sane} is $\square^*$. So suppose $(\mathcal{M}, w_0) \vDash \psi_{grid} \wedge \psi_{sane}$, and let w be any world in the generated submodel of $(\mathcal{M}, w_0)$.

- The formula $position_1$ implies that the grid points represented by w being (i) the location of the read/write head, (ii) to the right of the head, and (iii) to the left of the head, are mutually exclusive.

- The formula $position_2$ implies that if the point represented by w is either at the read/write head or to its right, then w's right-successor is to the right of the read/write head, and similarly for the left side. Together with $position_1$, this implies that for every $m \in \mathbb{Z}$, there is at most one n such that $(\mathcal{M}, (n, m)) \vDash pos$.

- The formula one_state implies that there is exactly one state s such that $(\mathcal{M}, w) \vDash s$.

- The formula $same\_state$ implies that w satisfies the same state as its *left-* and *right-* successors. So for every time m, there is exactly one state s such that $(\mathcal{M}, (n, m)) \vDash s$ for all n.

- The formula one_symbol implies that there is exactly one symbo l α such that $(\mathcal{M}, w) \vDash \alpha$.

$(\mathcal{M}, w_0) \vDash \chi_T$ then every world in the generated submodel represents exactly one point, but we do not need this fact so we will not prove it here.

45

$$\psi_T := \Box^* \bigwedge_{1 \leq i \leq 9} (position\_change \land state\_change \land symbol\_change)$$

$$position\_change := \bigwedge_{\{(s,\alpha)|\Delta_3(s,\alpha)=left\}} ((pos \land s \land \alpha) \to \Box_{up} \Box_{left}\, pos) \land$$

$$\bigwedge_{\{(s,\alpha)|\Delta_3(s,\alpha)=right\}} ((pos \land s \land \alpha) \to \Box_{up} \Box_{right}\, pos) \land$$

$$\bigwedge_{\{(s,\alpha)|\Delta_3(s,\alpha)=remain\}} ((pos \land s \land \alpha) \to \Box_{up} pos)$$

$$state\_change := \bigwedge_{s' \in states} \bigwedge_{\{(s,\alpha)|\Delta_2(s,\alpha)=s'\}} ((pos \land s \land \alpha) \to \Box_{up} s')$$

$$symbol\_change := \bigwedge_{\beta \in \Lambda} \bigwedge_{\{(s,\alpha)|\Delta_1(s,\alpha)=\beta\}} ((pos \land s \land \alpha) \to \Box_{up}\beta)$$

Table 7: The formula ψ_T.

- Recall that we use s_{void} as a dummy state which indicates that the execution of T has not started yet. The formula *void_state* implies that s_{void} holds on all worlds below worlds that satisfy either s_0 or s_{void}, so s_{void} does indeed hold in all worlds that represent points before the execution of T started in s_0.

- The formula *initial_symbol* guarantees that, in the initial state, the entire tape contains the symbol α_0.

- The formula *symbol_unchanged* guarantees that whenever the read/write head is not at a particular position of the tape, then the symbol at that position remains unchanged.

The above is almost sufficient to show that $\mathcal{M}$ represents the execution of some Turing machine, except for the following: (i) there is no guarantee that s_0 is satisfied anywhere, so the execution might never start; (ii) while the read/write head is guaranteed to be in at most one position, it is not guaranteed to be in at least position at all times; and (iii) the symbols that are written, the state changes, and the movement of the read/write head could be random, as opposed to fully determined by a set of rules. Whereas ψ_{sane} almost guarantees that $\mathcal{M}$ can be seen as the execution of some Turing machine, the formula ψ_T narrows this down to the specific machine T. In the process, it also solves one of the problems that remained after considering ψ_{sane}.

Suppose that $(\mathcal{M}, w_0) \vDash \chi_T$, so in addition to $(\mathcal{M}, w_0) \vDash \psi_{grid} \land \psi_{sane}$ we also have $(\mathcal{M}, w_0) \vDash \psi_T \land s_0 \land pos$. Then, the subformula *position_change* in ψ_T guarantees that the read/write head moves in accordance with Δ_3, the subformula *state_change* guarantees that the state changes in accordance with Δ_2, and the subformula *symbol_change* guarantees that the symbol that is written on the tape by the read/write head is in accordance with Δ_1. Note that this solves problem (iii) of ψ_{sane}: the movement, state changes, and written symbols are in accordance with the deterministic set of rules that is represented by Δ.

Finally, consider the conjuncts s_0 and *pos* in χ_T. These solve the other two problems of ψ_{sane}: the execution starts at state s_0 in the point represented by w_0, which represent

46

point $(0, 0)$. Furthermore, the read/write head starts there as well, so the read/write head is initially in at least one position. From the fact that the head moves deterministically, and that this is encoded in ψ_T, it then follows that the read/write head is also in at least one position at every time after $m = 0$.

To conclude, the above shows that if $(\mathcal{M}, w_0) \vDash \chi_T$, then the generated submodel of $(\mathcal{M}, w_0)$ represents a grid, and this grid contains the encoding of an execution of T.

Theorem 58. *A Turing machine T is halting if and only if $\mathcal{K}_{modal} \vDash \chi_T \to \Diamond^* s_{end}$, and non-halting if and only $\mathcal{K}_{modal} \vDash \chi_T \to \Box^* \neg s_{end}$.*

Proof. Let $(\mathcal{M}, w_0)$ be any pointed model such that $(\mathcal{M}, w_0) \vDash \chi_T$. Then, as shown above, every world in the generated submodel represents some point (n, m) in a $\mathbb{Z} \times \mathbb{Z}$ grid and, furthermore, the worlds representing a point (n, m) satisfy $s \in S$ if and only if the system is in state s at time m. It follows that the generated submodel contains a s_{end} world if and only if T is halting, from which the theorem follows immediately. □

As an immediate consequence of Theorem 58, we obtain the following result.

Corollary 59. SOPML* *is not axiomatisable on modal frames for any number of agents.*

This unaxiomatisability result can be extended to the classes of *boolean* and *all* frames, since the modal frames can be characterized inside *boolean* or *any* frame. Let $\phi_\neg :=$ $\forall p \exists q \, \Box^* (\neg p \leftrightarrow q)$, $\phi_\vee := \forall p \forall q \exists r \, \Box^* ((p \vee q) \leftrightarrow r)$ and $\phi_\Box := \bigwedge_{a \in Ag} \forall p \exists q \, \Box^* (\Box_a p \leftrightarrow q)$. The following lemma is entirely straightforward, so we state it without proof.

Lemma 60. *Let $(\mathcal{M}, w)$ be a pointed model based on any frame. Then, the generated submodel of $(\mathcal{M}, w)$ is based on a modal frame if and only if $(\mathcal{M}, w) \vDash \phi_\neg \wedge \phi_\vee \wedge \phi_\Box$.*

Hence, the following corollary follows immediately from Lemma 60.

Corollary 61. *A formula $\phi \in \mathcal{L}_{sopml}$ is valid on modal frames if and only if $(\phi_\neg \wedge \phi_\vee \wedge \phi_\Box) \to \phi$ is valid on all or boolean frames.*

As a result of the unaxiomatisability of SOPML* on modal frames, we finally obtain unaxiomatisability for all classes of frames.

Corollary 62. SOPML* *is not axiomatisable neither on the class of all frames nor on the class of boolean frames, for any number of agents.*

So far, we have shown that SOPML* is unaxiomatisable for modal, boolean, and all frames. This result can be extended to the corresponding classes of epistemic frames as well. The proofs for epistemic frames is very similar to the proofs above however, so we include them in the appendix only.

Theorem 63. *If $|I| \geq 2$, then the validities in SOPML* over modal epistemic frames are not recursively enumerable.*

In particular, SOPML* *is not axiomatisable on the class of modal epistemic frames.*

The same characterisation of boolean and all frames also applies to epistemic frames. So we also have the following corollary.

Corollary 64. SOPML* *is not axiomatisable on the class of all or boolean epistemic frames, for any number of agents.*

47

5. Simulations and Games

In this section we investigate the expressive power of second-order propositional modal logic by introducing truth-preserving (bi)simulation relations for SOPML. Bisimulations are an essential tool for the model theory of propositional modal logic, as they provide sufficient conditions under which two models satisfy the same formulas in PML [9, 32]. Moreover, propositional modal logic is characterized by the well-known van Benthem theorem as the bisimulation-invariant fragment of first-order logic [8]. Hereafter we introduce simulations and bisimulations for SOPML and prove that they are indeed truth-preserving. Further, in Section 5.2 we present a notion of abstraction for frames and show that an abstraction of a frame simulates that frame. In Section 5.3 we define games for (bi)simulations and prove the equivalence of the two approaches: model- and game-theoretic. Finally, in Section 5.4 we provide examples on the application of (bi)simulations to the analysis of the expressive power of SOPML in spatial and temporal reasoning.

5.1. Simulations and Bisimulations

We define the notion of (bi)simulations on frames, although it is immediate to extend this definition to models. In the rest of the section we consider frames $\mathcal{F} = \langle W, D, R \rangle$, $\mathcal{F}' = \langle W', D', R' \rangle$, and models $\mathcal{M} = \langle \mathcal{F}, V \rangle$, $\mathcal{M}' = \langle \mathcal{F}', V' \rangle$ defined on $\mathcal{F}$ and $\mathcal{F}'$ respectively. In the following, Σ denotes a relation $\Sigma \subseteq D \times D$?

Definition 65 (Frame Simulation). *Given frames $\mathcal{F}$ and $\mathcal{F}'$, a* simulation *is a pair (σ, Σ) of relations $\sigma \subseteq W \times W'$, $\Sigma \subseteq D \times D'$ such that* (i) *for every $U \in D$, $\Sigma(U, U')$ for some $U' \in D'$; and* (ii) *$\sigma(w, w')$ implies*

1. for every $v \in W$, $a \in I$, if $R_a(w, v)$ then $\sigma(v, v')$ for some $v' \in R'_a(w')$;

2. for every $U \in D$, $U' \in D'$, $\Sigma(U, U')$ implies $w \in U$ iff $w' \in U'$.

Notice that condition 1 in Definition 65 expresses the standard notion of simulation in PML. Hence, simulations for SOPML extend the corresponding definition for PML (we devote more discussion to this point later on). The definition of simulation above differs from a similar notion put forward in [4]. Specifically, in [4] only a relation on states is considered, thus obtaining a strictly weaker notion. This will become apparent when analysing simulation games in Section 5.3.

We say that state w' *simulates* w, or $w \preceq w'$, iff $\sigma(w, w')$ holds for some simulation pair (σ, Σ). Similarly, a set U' *simulates* U, or $U \preceq U'$, iff $\Sigma(U, U')$ holds for some simulation pair (σ, Σ). Note that it may be that $w \prec w'$ holds because of (σ_1, Σ_1) and $U \prec U'$ holds because of (σ_2, Σ_2), while (σ_1, Σ_2) is not a simulation pair. To see this, consider the frames $\mathcal{G}_1 = \langle \{w_1, w_2\}, \{\{w_1\}, \{w_2\}\}, \{(w_1, w_2), (w_2, w_1)\} \rangle$ and $\mathcal{G}_2 = \langle \{x_1, x_2\}, \{\{x_1\}, \{x_2\}\}, \{(x_1, x_2), (x_2, x_1)\} \rangle$. Clearly, $w_1 \preceq x_1$ and $\{w_1\} \preceq \{x_2\}$. However, it is not the case that $w_1 \in \{w_1\}$ iff $x_1 \in \{x_2\}$. Nonetheless, each $\preceq$ is a preorder, i.e., a reflexive and transitive relation. Finally, a frame $\mathcal{F}'$ *simulates* $\mathcal{F}$, or $\mathcal{F} \preceq \mathcal{F}'$, iff for every $w \in W$, $w \preceq w'$ for some $w' \in W'$.

Observe that for PML (that is, whenever we ignore the quantification domain D), the notion of simulation given on frames is vacuous, as we discard the evaluation of propositional atoms in the various states. Then, for instance, all serial frames simulate

each other. However, this remark does not apply to SOPML, as we also have to take into account propositional quantification.

We illustrate the newly introduced notion by an example.

Example 66. *Consider frames $\mathcal{G} = \langle W, R, D \rangle$ and $\mathcal{G}' = \langle W', R', D' \rangle$ over set $I = \{a, b, c\}$ of indexes, depicted in Figure 4, with*

- *$W = \{w_1, w_2, w_3\}$;*

- *$R_a = \{(w_1, w_3), (w_3, w_1)\}$, $R_b = \{(w_1, w_2), (w_2, w_1)\}$, $R_c = \{(w_2, w_3), (w_3, w_2)\}$;*

- *$D = \{\{w_1\}, \{w_2\}, \{w_3\}\}$;*

- *$W' = \{u_s \mid s$ is a finite sequence on $\{1, 2, 3\}$ starting with 1, with no adjacent repetition$\}$;*

- *for every $i \in I$, $R'_i = \{(u_s, u_{s'}) \mid s' = s \cdot m$ and $R_i(w_{last(s)}, w_m)\}$;*

- *let $U'_n = \{u_s \mid last(s) = n\}$, then $D' = \{U'_1, U'_2, U'_3\}$.*

Intuitively, frame $\mathcal{G}$ can be thought of as a scenario where robots a, b, and c move around locations w_1, w_2, w_3 (robot a moves between w_1 and w_3, etc.) Frame $\mathcal{G}'$ then captures the same scenario but with the additional possibility to reason about some notion of history, or time (one might for instance add an atom p_i which is true exactly at nodes at level i. To do this, one needs to make appropriate assumptions about D' in $\mathcal{G}'$, like requiring that the frame is full. We do not consider these matters further.)

Now consider the pair (σ, Σ) of relations $\sigma \subseteq W \times W'$ and $\Sigma \subseteq D \times D'$ such that $\sigma(w_n, u_s)$ holds iff $last(s) = n$ and $\Sigma(\{w_n\}, U'_m)$ holds iff $n = m$. We check that (σ, Σ) is indeed a simulation. Firstly, for every $\{w_n\} \in D$, we have $\Sigma(\{w_n\}, U'_n)$ for $U_n \in D'$. Secondly, if $\sigma(w_n, u_s)$ and $R_i(w_n, w_m)$, then $s' = s \cdot m$ is such that $R'_i(u_s, u_{s'})$ and $\sigma(w_m, u_{s'})$. Thirdly, if $\sigma(w_n, u_s)$ and $\Sigma(\{w_k\}, U'_m)$, then $last(s) = n$ and $k = m$. Therefore, $w_n \in \{w_k\}$ iff $n = k$, iff $last(s) = m$, iff $u_s \in U'_m$.

Finally, we observe that for every $w_n \in W$, $\sigma(w_n, u_s)$ for $last(s) = n$. Thus, frame $\mathcal{G}'$ simulates $\mathcal{G}$.

Remark 67. *We observe the following regarding the relation between simulations and properties of frames.*

1. *It can be checked in NPTIME whether there exists a simulation relation between two frames.*

2. *If a frame $\mathcal{F}'$ simulates a boolean (respectively modal, full) frame $\mathcal{F}$, then $\mathcal{F}'$ need not to be boolean (respectively modal, full). Nor does $\mathcal{F}'$ being boolean (modal, full) imply that $\mathcal{F}$ is also boolean (modal, full).*

As regards the first point, observe that we can guess a pair (σ, Σ) of relations on states and sets respectively, and then check in polynomial time whether it is actually a frame simulation.

As to the second point, consider that a simulating frame $\mathcal{F}'$ may contain sets of states that do not simulate any state in $\mathcal{F}$, which are not closed under set-theoretic operations.
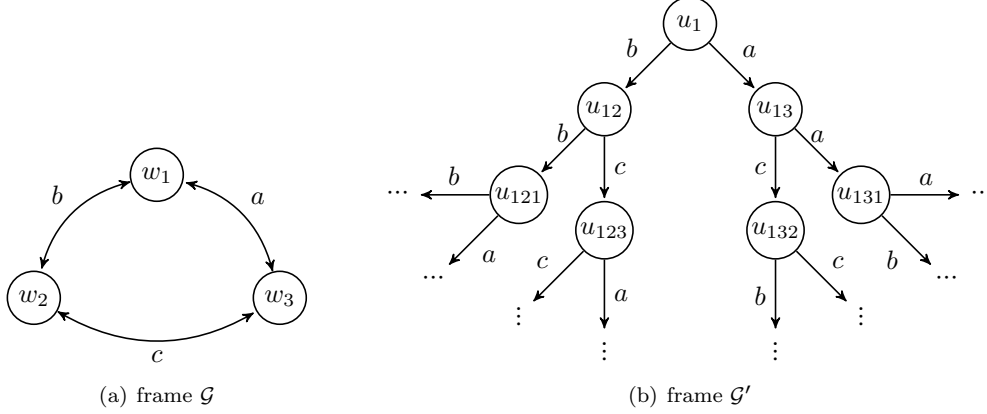
49

(a) frame $\mathcal{G}$ (b) frame $\mathcal{G}'$

Figure 4: frames $\mathcal{G}$ and $\mathcal{G}'$ in Example 66. D components are omitted for clarity.

The other implication can be proved by a similar line of reasoning. As an example, consider frames $\mathcal{F} = \{\{w_1, w_2\}, \{(w_1, w_2)\}, 2^{\{w_1, w_2\}}\}$ and $\mathcal{F}' = \{\{u_1, u_2, u_3, u_4\}, \{(u_1, u_2), (u_1, u_3), (u_1, u_4)\}, \{\{u_1\}, \{u_2, u_3\}, \{u_1, u_2, u_3\}, \{u_4\}, \varnothing\}\}$. In particular, $\mathcal{F}'$ is a simulation of $\mathcal{F}$ with relations σ and Σ such that $\sigma(w_1, u_1)$, $\sigma(w_2, u_2)$, $\sigma(w_2, u_3)$, and $\Sigma(\{w_1\}, \{u_1\})$, $\Sigma(\{w_2\}, \{u_2, u_3\})$, $\Sigma(\{w_1, w_2\}, \{u_1, u_2, u_3\})$, $\Sigma(\varnothing, \varnothing)$. But $\mathcal{F}$ is a full frame, while $\mathcal{F}'$ is not even boolean. Thus, similar frames do not necessarily belong to the same class. Below we compare these results with those available for bisimulations.

We now state the following preservation result for the universal fragment of SOPML.

Theorem 68. *If $w \preceq w'$, then for every $\varphi \in \mathcal{L}^*_{a-sopml}$,*

$$(\mathcal{F}', w') \vDash \varphi \quad implies \quad (\mathcal{F}, w) \vDash \varphi$$

Proof. Since $w \preceq w'$, there is a simulation pair (σ, Σ) such that $\sigma(w, w')$. Fix this σ. Once can prove by induction on φ that if $(\mathcal{F}, V, w) \nvDash \varphi$ for some assignment V, then $(\mathcal{F}', \Sigma(V), w') \nvDash \varphi$, where $\Sigma(V)$ is any assignment such that for every $p \in AP$, $(\Sigma(V))(p) = U'$ with $\Sigma(V(p), U')$. We only show the step for the quantifier. We write $\Sigma(V)(p)$ for $(\Sigma(V))(p)$. By clause *(i)* of Definition 65, $\Sigma(V)(p) \in D'$.

For $\varphi = \forall p \psi$, $(\mathcal{F}, V, w) \nvDash \varphi$ iff for some $U \in D$, $(\mathcal{F}, V^p_U, w) \nvDash \psi$. By induction hypothesis, $(\mathcal{F}', \Sigma(V^p_U), w') \nvDash \psi$. By condition *(i)* in Definition 65, for $U \in D$, $\Sigma(U, U')$ for some $U' \in D'$. In particular, we have that $\Sigma(V^p_U) = \Sigma(V)^p_{U'}$, and therefore $(\mathcal{F}', \Sigma(V)^p_{U'}, w') \nvDash \psi$ for $U' \in D'$, that is, $(\mathcal{F}', \Sigma(V), w') \nvDash \varphi$. $\qquad\square$

As an immediate consequence of Theorem 68 we obtain the following corollary.

Corollary 69. *If $\mathcal{F} \preceq \mathcal{F}'$, then for every $\varphi \in \mathcal{L}^*_{a-sopml}$,*

$$\mathcal{F}' \vDash \varphi \quad implies \quad \mathcal{F} \vDash \varphi$$

Thus, the notion of simulation introduced in Definition 65 preserves the universal fragment of SOPML, similarly to the case for standard simulations and PML.

50

Example 70. *Consider again frames $\mathcal{G}$ and $\mathcal{G}'$ in Example 66. We showed that $\mathcal{G}'$ simulates $\mathcal{G}$. Moreover, we can easily check that $\mathcal{G}'$ validates the following formula in* SOPML

$$\forall p \left(p \to \bigvee_{i \in I} \Box_i \neg p \right) \tag{16}$$

which intuitively says that at each position some agent moves to a different position. By Corollary 69 we deduce that (16) is valid in $\mathcal{G}$ as well.

Simulations can naturally be extended to bisimulations. Also in this case, our focus is at the level of frames. In the following the *converse* of a relation R is the relation $R^{-1} = \{(u,v) \mid R(v,u)\}$.

Definition 71 (Frame Bisimulation). *Given frames $\mathcal{F}$ and $\mathcal{F}'$, a bisimulation is a pair (ω, Ω) of relations $\omega \subseteq W \times W'$, $\Omega \subseteq D \times D'$ such that both (ω, Ω) and $(\omega^{-1}, \Omega^{-1})$ are simulations. That is, (i) for every $U \in D$, $\Omega(U, U')$ for some $U' \in D'$, and for every $U' \in D'$, $\Omega(U, U')$ for some $U \in D'$; and (ii) $\omega(w, w')$ implies*

1. *for every $v \in W$, $a \in I$, if $R_a(w, v)$ then $\omega(v, v')$ for some $v' \in R_a'(w')$;*

2. *for every $v' \in W'$, $a \in I$, if $R_a'(w', v')$ then $\omega(v, v')$ for some $v \in R_a(w)$;*

3. *for every $U \in D$, $U' \in D'$, $\Omega(U, U')$ implies $w \in U$ iff $w' \in U'$.*

States w and w' are *bisimilar*, or $w \approx w'$, iff $\omega(w, w')$ holds for some bisimulation pair (ω, Ω). Similarly, sets U' and U are *bisimilar*, or $U \approx U'$, iff $\Omega(U, U')$ holds for some bisimulation pair (ω, Ω). Again, the pair $(\approx, \approx)$ is not necessarily a bisimulation, similarly to what was shown above for simulations, but each $\approx$ is an equivalence relation. Finally, frames $\mathcal{F}$ and $\mathcal{F}'$ are *bisimilar*, or $\mathcal{F} \approx \mathcal{F}'$, iff *(i)* for every $w \in W$, $w \approx w'$ for some $w' \in W'$; and *(ii)* for every $w' \in W'$, $w \approx w'$ for some $w \in W$.

Example 72. *Notice that frames $\mathcal{G}$ and $\mathcal{G}'$ in Example 66 are actually bisimilar. To prove this fact, we show that the converse relations $\sigma^{-1} \subseteq W' \times W$ and $\Sigma^{-1} \subseteq D' \times D$ form a simulation pair. Firstly, for every $U_n' \in D'$, the set $U = \{w_n\} \in D$ is such that $\Sigma(U, U')$. Secondly, if $\sigma^{-1}(u_s, w_n)$ and $R_i'(u_s, u_{s'})$ then $last(s) = n$ and $s' = s \cdot m$ for $w_m \in W$ such that $R_i(w_n, w_m)$. Hence, $\sigma^{-1}(u_{s'}, w_m)$. As to (3), the proof is identical as for simulations.*

Remark 73. *We now state the following adaptation of Remark 67.*

1. *It can be checked in NPTIME whether there exists a bisimulation relation between two frames.*

2. *Suppose that $\mathcal{F}$ and $\mathcal{F}'$ are bisimilar. Then, $\mathcal{F}$ is boolean (respectively modal) iff $\mathcal{F}'$ is. However, if $\mathcal{F}$ is full, then $\mathcal{F}'$ need not to be full. Nor does $\mathcal{F}'$ being full imply that $\mathcal{F}$ is also full. Moreover, if $\mathcal{F}$ and $\mathcal{F}'$ are both bisimilar and full, then they are isomorphic, that is, any bisimulation between full $\mathcal{F}$ and $\mathcal{F}'$ is an isomorphism.*

Compare the situation for bisimulations with the weaker results available in Remark 67 for simulations. Specifically, bisimulations preserve the class of boolean and

51

modal frames. Moreover, in the case of full frames, bisimulations collapse into isomorphisms.

We now state the main preservation result of this section. Its proof is similar to that of Theorem 68, and it is in the appendix.

Theorem 74. *If* $w \approx w'$, *then for every formula* $\varphi \in \mathcal{L}^*_{sopml}$,

$$(\mathcal{F}, w) \vDash \varphi \quad iff \quad (\mathcal{F}', w') \vDash \varphi.$$

As an immediate consequence of Theorem 74 we obtain the following.

Corollary 75. *If* $\mathcal{F} \approx \mathcal{F}'$, *then for every* $\varphi \in \mathcal{L}^*_{sopml}$,

$$\mathcal{F} \vDash \varphi \quad iff \quad \mathcal{F}' \vDash \varphi$$

We can now infer that bisimulations in SOPML are 'stronger' than the corresponding notion for PML: whereas we noticed that the frames of Figure 3 are bisimilar in PML, as a consequence of Theorem 74, and Example 11, which says that the frames do not agree on formula (8), we conclude that they are not bisimilar in the SOPML sense.

Example 76. *We now consider two graph-theoretic properties. First, the notion of 3-colorability, as formalised by the following* SOPML *formula, where operator* $\square$ *is interpreted on the edges* $E \subseteq W^2$ *of a graph* $G = \langle W, E \rangle$, *while* $\square^*$ *is interpreted on the reflexive and transitive closure of* E:

$$\exists p_1, p_2, p_3 \left(\square^*(p_1 \vee p_2 \vee p_3) \wedge \square^* \bigwedge_{i \neq j} \neg(p_i \wedge p_j) \wedge \bigwedge_{1,2,3} \square^*(p_i \rightarrow \neg \diamond p_i) \right) \qquad (17)$$

The truth of (17) in a vertex $v \in G$ *implies that* (i) *all vertices in the subgraph generated by* v *are coloured by either* p_1, p_2, *or* p_3; (ii) *each vertex has at most one colour; and* (iii) *no two adjacent vertices have the same colour. Thus, the subgraph generated by* v *is 3-colorable. Observe that frame* $\mathcal{G}$ *in Figure 4(a) is indeed 3-colorable, and since states* w_1 *and* u_1 *are bisimilar, as an immediate consequence of Theorem 74, also frame* $\mathcal{G}'$ *is 3-colorable. Notice that the truth of (17) implies that the graph is 3-colorable; while the converse holds only for full frames.*

To illustrate further the (in)expressivity of SOPML *through simulations, we consider one more graph-theoretic property: the existence of a Hamiltonian path, i.e., a path that visits all vertices in a graph exactly once. Again, frame* $\mathcal{G}$ *in Figure 4(a) has a Hamiltonian path* w_1, w_2, w_3. *Now consider the finite graph* $\mathcal{G}''$ *which is obtained from* $\mathcal{G}$ *and* $\mathcal{G}'$ *in Figure 4 as follows. In* $\mathcal{G}'$ *replace the worlds* u_{12} *and* u_{13} *by a copy of* w_2 *and* w_3 *from* $\mathcal{G}$ *respectively, and remove all the worlds different from* u_1. *It is not difficult to see that* $\mathcal{G}''$ *is bisimilar with* $\mathcal{G}$, *but does not allow a Hamiltonian path.*

Proposition 77. *The property of having a Hamiltonian path is not expressible in* SOPML.

Proof.   The proof follows directly from the second part of Example 76, which implies that such paths are not expressible under the general semantics.   □

Proposition 77 also follows from [46, Corollary 7.24], which says that it is impossible to test in MSO whether a graph is Hamiltonian. Indeed, it is known that such property is

52

expressible in the language MSO$_2$, an extension of MSO, which is strictly more expressive than SOPML [16, Prop. 5.13].

Discussion. We now compare our definition of (bi)simulation for SOPML, with the standard notion of (bi)simulation for PML [9]. Observe that if a frame $\mathcal{F}'$ simulates $\mathcal{F}$ in SOPML, with simulation pair (σ, Σ), then for every model $\mathcal{M} = \langle \mathcal{F}, V \rangle$ based on $\mathcal{F}$, model $\mathcal{M}' = \langle \mathcal{F}', \Sigma(V) \rangle$ on $\mathcal{F}'$ PML-simulates $\mathcal{M}$. In particular, if $\sigma(w, w')$ then for every $v \in W$, $a \in I$, $R_a(w, v)$ implies that $\sigma(v, v')$ for some $v' \in R'_a(w')$ by condition *(ii)*.1 in Definition 65. Moreover, $w \in V(p) \in D$ iff $w' \in \Sigma(V)(p) \in D'$ by conditions *(i)* and *(ii)*.2. Therefore, if $\mathcal{M}'$ satisfies any universal formula ϕ in PML, then ϕ also holds in $\mathcal{M}$. Hence, Definition 65 of simulation for frames in SOPML is indeed a generalisation of the model-theoretic notion in PML. Furthermore, if frames $\mathcal{F}'$ and $\mathcal{F}$ are bisimilar in SOPML, with bisimulation pair (ω, Ω), then models $\mathcal{M} = \langle \mathcal{F}, V \rangle$ and $\mathcal{M}' = \langle \mathcal{F}', \Omega(V) \rangle$ are also bisimilar in PML. Likewise, models $\mathcal{M}' = \langle \mathcal{F}', V' \rangle$ and $\mathcal{M} = \langle \mathcal{F}, \Omega^{-1}(V') \rangle$ are PML-bisimilar as well. Also in this case, SOPML bisimulations on frames generalise PML bisimulations on models.

5.2. Abstraction

This section is devoted to the definition of a notion of abstraction for Kripke frames. Abstractions are deemed useful for system verification, as they allow to ignore some selected features of the system, thus focusing only on the properties relevant for the verification task [15]. Indeed, a key fact about abstractions is that they simulate the original system. Hereafter we prove such a result for SOPML, starting with a family of equivalence relations on states.

Definition 78 (Equivalence). *Given a frame $\mathcal{F}$, consider an equivalence relation $\sim$ on W such that for every state $w, w' \in W$, $w \sim w'$ implies that for every $U \in D$, $w \in U$ iff $w' \in U$. Further, we denote by $[w] = \{w' \in W \mid w' \sim w\}$ the equivalence class of w in $\mathcal{F}$, and for a set $U \subseteq W$, we let $[U]$ be $\{[w] \mid w \in U\}$.*

Clearly, if we replace 'implies' in Definition 78 by 'iff', we obtain the coarsest equivalence relation satisfying the conditions in Definition 78.

Definition 79 (Abstraction). *Given a frame $\mathcal{F}$, the* abstraction *$\mathcal{F}^A = \langle W^A, D^A, R^A \rangle$ of $\mathcal{F}$ (according to equivalence relation $\sim$) is the frame such that*

- $W^A = \{[w] \mid w \in W\}$*;*

- $D^A = \{[U] \mid U \in D\}$*;*

- *for every $a \in I$, $R^A_a([w], [w'])$ iff $R_a(v, v')$ for some $v \in [w]$, $v' \in [w']$.*

Notice that the coarsest abstraction $\mathcal{F}^A$ is finite whenever the interpretation domain D in $\mathcal{F}$ is, and of size $|W^A| = \mathcal{O}(2^D)$ at most.

Example 80. *To illustrate abstractions, we show that the frame $\mathcal{G}$ in Example 66 is (isomorphic to) the coarsest abstraction $\mathcal{G}'^A$ of $\mathcal{G}'$. First of all, two worlds u_s and $u_{s'}$ are equivalent according to the coarsest equivalence $\sim$ iff for all $U'_n \in D'$, $u_s \in U'_n$ iff $u_{s'} \in U'_n$, iff $last(s) = last(s')$. So, in abstraction $\mathcal{G}'^A$ we have three equivalence classes $[u_{t \cdot 1}]$, $[u_{t \cdot 2}]$, and $[u_{t \cdot 3}]$, for (possibly empty) sequences $t \in \{1, 2, 3\}^*$ beginning*

53

with 1. *As to the accessibility relations, $R_i'^A([u_{t\cdot n}][u_{t'\cdot m}])$ iff for $u_{t\cdot n}$, $u_{t'\cdot m}$ in W', $R_i'(u_{t\cdot n}, u_{t'\cdot m})$, that is, $t' = t \cdot n$ and $R_i(w_n, w_m)$. Hence, for instance, for agent a, we have $R_a'^A([u_{t\cdot 1}][u_{t'\cdot 3}])$ and $R_a'^A([u_{t\cdot 3}][u_{t'\cdot 1}])$, as required. Finally, $D'^A = \{[U_n'] \mid U_n' \in D'\} = \{\{[u_{t\cdot 1}]\}, \{[u_{t\cdot 2}]\}, \{[u_{t\cdot 3}]\}\}$. Clearly, the abstraction $\mathcal{G}'^A$ of $\mathcal{G}'$ is isomorphic to $\mathcal{G}$, with mapping $w_i \mapsto [u_{t\cdot i}]$ for $i = 1, 2, 3$.*

We now extend a standard result in modal logic, namely that abstractions are indeed simulations, to SOPML.

Lemma 81. *Given a frame $\mathcal{F}$ with abstraction $\mathcal{F}^A$, the pair of mappings $w \mapsto [w]$ and $U \mapsto [U]$ is a simulation.*

Proof. We show that the pair $(\mapsto, \mapsto)$ of mappings satisfies Definition 65. As to condition *(i)*, if $U \in D$ then $U \mapsto [U]$ for $[U] \in D^A$. Next, for *(i)*.1 suppose that $R_a(w, v)$. Then, for $[v] \in W^A$ we have that $R_a^A([w], [v])$ and $v \mapsto [v]$. Finally, as to *(ii)*.2, if $w \mapsto [w]$ and $w \in U$, then clearly $[w] \in [U]$. On the other hand, if $[w] \in [U]$ then for some $v \in [w]$, $v \in U$. However, $v \in [w]$ implies that $v \sim w$. In particular, $w \in U$ by the constraint on $\sim$. $\square$

We remark that the abstraction $\mathcal{F}^A$ of a full frame $\mathcal{F}$ is isomorphic to $\mathcal{F}$. In fact, for every $w \in W$, the set $\{w\}$ belongs to D, and since $w \sim w'$ iff for all $U \in D$, $w \in U$ iff $w' \in U$, $w \sim w'$ implies in particular that $w \in \{w'\}$, that is, $w = w'$. As a consequence, $w \mapsto \{w\}$ is the only simulation on states between $\mathcal{F}$ and $\mathcal{F}^A$, and it is also an isomorphism. Further, in Example 80 we observed that frame $\mathcal{G}$ is (isomorphic to) the coarsest abstraction of $\mathcal{G}'$. Hence, Lemma 81 provides an alternative proof of the fact that $\mathcal{G}$ simulates $\mathcal{G}'$, that we discussed in Example 72.

The following corollary follows immediately from Lemmas 68 and 81.

Corollary 82. *Let $\mathcal{F}$ be a frame with abstraction $\mathcal{F}^A$. For every universal formula $\varphi \in \mathcal{L}_{a-sopml}^*$,*

$$(\mathcal{F}^A, [w]) \vDash \varphi \quad implies \quad (\mathcal{F}, w) \vDash \varphi$$

The results presented above have an impact that goes beyond their theoretical interest. As an example, we observed that relevant properties P of frames (such as reflexivity, transitivity, symmetry, etc.) are definable in propositional modal logic in the sense that for some formula ϕ in PML, a frame $\mathcal{F}$ validates ϕ iff $\mathcal{F}$ satisfies property P. In SOPML more properties become frame-definable within the class of full frames. For instance, in Section 3 we showed that a full frame $\mathcal{F}$ is irreflexive iff $\mathcal{F} \vDash \exists p(\Box p \wedge \neg p)$. On the other hand, whenever we consider the class of all frames, several properties are non-definable (even where they might be definable in PML). For instance, the frame $\mathcal{G}$ in Example 66 is symmetric, while $\mathcal{G}'$ is not. Since, $\mathcal{G}$ and $\mathcal{G}'$ are bisimilar, and therefore satisfy the same formulas in SOPML, we conclude that symmetry is not definable in the class of all frames. Likewise, irreflexivity nor reflexivity are definable on the class of all frames: take $\mathcal{F}$ consisting of only one reflexive world w with $D = \{\{w\}, \varnothing\}$, and $\mathcal{F}'$ consisting of two worlds w_1', w_2' with $R' = \{(w_1', w_2'), (w_2', w_1')\}$ and $D' = \{\{w_1', w_2'\}, \varnothing\}$. The pointed frames $(\mathcal{F}, w)$ and $(\mathcal{F}', w_1')$ are bisimilar, but $\mathcal{F}$ is reflexive (hence reflexivity cannot be expressed on all frames) while $\mathcal{F}'$ is irreflexive (hence irreflexivity cannot be expressed). Such results provide us with further knowledge on the expressive power of SOPML.

5.3. Simulation Games for SOPML

The relation between logic and games is a domain of investigation that has produced fundamental results and fruitful applications to the analysis of the expressiveness of formal languages. As an example, Ehrenfeucht-Fraïssé games are a valuable tool for the model theory of first-order logic [46]. Similarly, Fagin-Ajtan games (and variations thereof) characterise (fragments of) Second-order Logic [28]. More strictly related to the present contribution, bisimulation games are routinely applied to derive (in-)expressivity results in propositional modal logic [6, 9].

In this section we introduce (bi)simulation games for SOPML. Similarly to the case for PML, the existence of a winning strategy for Duplicator guarantees the preservation of (universal) formulas in SOPML. We start by considering simulation games played by Spoiler and Duplicator.

Definition 83 (Simulation Game). *A simulation game G starting from pointed frames $(\mathcal{F}, w)$ and $(\mathcal{F}', w')$ is defined as follows. Let $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ be the current state of the game, where $v \in W$ (respectively, $v' \in W'$) and $\vec{U}$ (respectively, $\vec{U}'$) is a possibly empty tuple of sets in D (respectively, D').*

Then, the game proceeds according to the following rules:

1. *Either Spoiler picks a set $U \in D$ and Duplicator has to reply with a set $U' \in D'$ such that $v \in U$ iff $v' \in U'$. The new state of the game is $(\mathcal{F}, v, \vec{U} \cdot U)$, $(\mathcal{F}', v', \vec{U}' \cdot U')$.*

2. *Or, for some $a \in I$, Spoiler picks a state $u \in R_a(v)$ and Duplicator has to reply with state $u' \in R'_a(v')$ such that for every i, $u \in U_i$ iff $u' \in U'_i$. The new state of the game is $(\mathcal{F}, u, \vec{U})$, $(\mathcal{F}', u', \vec{U}')$.*

If Duplicator cannot match a move by Spoiler, then Spoiler wins the game. Otherwise, Duplicator wins the game. A *winning strategy* is a strategy whereby Duplicator can reply to all of Spoiler's moves, thus winning the game.

By the following result the existence of a winning strategy is tantamount to the existence of a simulation.

Theorem 84. *Duplicator has a winning strategy for the simulation game starting in $(\mathcal{F}, w)$, $(\mathcal{F}', w')$ iff $(\mathcal{F}, w) \preceq (\mathcal{F}', w')$.*

Proof. As to the $\Leftarrow$-direction, we show that if (σ, Σ) is a simulation pair such that $\sigma(v, v')$, then Duplicator can always reply to any Spoiler's move in state $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ with $\Sigma(U_i, U'_i)$. Moreover, the new state $(\mathcal{F}, u, \vec{T})$, $(\mathcal{F}', u', \vec{T}')$ is such that $\sigma(u, u')$ and $\Sigma(T_i, T'_i)$. As a consequence, Duplicator has a winning strategy and the result follows. As regards move (1) in Definition 83, suppose that Spoiler picks a set $U \in D$. Then Duplicator can reply with $U' \in D'$ such that $\Sigma(U, U')$: by *(i)* in Definition 65 such U' always exists. Moreover, by *(ii)*.2, $v \in U$ iff $v' \in U'$. On the other hand, if Spoiler chooses $u \in R_a(v)$, then by *(ii)*.1 Duplicator can reply with some $u' \in R'_a(v')$ such that $\sigma(u, u')$. In particular, for every i, $u \in U_i$ iff $u' \in U'_i$, as $\Sigma(U_i, U'_i)$ holds. In both cases the new state $(\mathcal{F}, u, \vec{T})$, $(\mathcal{F}', u', \vec{T}')$ is such that $\sigma(u, u')$ and $\Sigma(T_i, T'_i)$ as required.

As to the $\Rightarrow$-direction, we show that relations $\sigma \subseteq W \times W'$ and $\Sigma \subseteq D \times D'$ defined as: $\sigma(v, v')$ *and* $\Sigma(U_i, U'_i)$ *hold iff Duplicator has a winning strategy at state* $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ *form a simulation pair.* As regards condition *(i)* in Definition 65, consider

any winning state $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ in the game starting in $(\mathcal{F}, w)$, $(\mathcal{F}', w')$. Spoiler can play any $U \in D$, but then Duplicator has to reply with some $U' \in D'$ so that he has a winning strategy in the resulting state $(\mathcal{F}, v, \vec{T})$, $(\mathcal{F}', v', \vec{T}')$. In particular, by the definition of Σ, we have $\Sigma(U, U')$. Further, for condition *(ii)*.1, if $\sigma(v, v')$ then $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ is a winning state for some $\vec{U}, \vec{U}'$ with $\Sigma(U_i, U'_i)$. Then, for every $u \in R_a(v)$, Duplicator can reply with $u' \in R'_a(v')$ (otherwise, Duplicator has no winning strategy.) Moreover, Duplicator has a winning strategy in the resulting state $(\mathcal{F}, u, \vec{U})$, $(\mathcal{F}', u', \vec{U}')$. Hence, we have $\sigma(u, u')$ by definition of σ. Finally, for condition *(ii)*.2, if $\sigma(v, v')$ then again $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ is a winning state for some $\vec{U}$ and $\vec{U}'$. Further, if $\Sigma(T, T')$ then $(\mathcal{F}, u, \vec{T})$, $(\mathcal{F}', u', \vec{T}')$ is a winning state for some $u \in W$ and $u' \in W'$ such that $\sigma(u, u')$. Now we analyse the following two cases: if couple (v, v') has appeared before (u, u') in the game then, when introducing sets U and U', condition $v \in U$ iff $v' \in U'$ has to be satisfied by (1) in Definition 83. On the other hand, if couple (u, u') has appeared first, when introducing (v, v') condition $v \in U$ iff $v' \in U'$ has to be satisfied by (2). Finally, since Duplicator has a winning strategy for game $(\mathcal{F}, w)$, $(\mathcal{F}', w')$, it is the case that w' simulates w. $\qquad\square$

Notice that this result is in marked contrast with [4], where the notion of simulation provided entails the existence of a winning strategy for Duplicator, but the existence of a winning strategy does not imply the existence of a simulation. On the contrary, here we have a perfect match between the two notions. This is due to the novel notion of simulation put forward in Definition 65.

As a direct consequence of Theorems 68 and 84, we obtain that the existence of a winning strategy for Duplicator implies the preservation of formulas in $\mathcal{L}^*_{a-sopml}$.

Corollary 85. *If Duplicator has a winning strategy for the game starting in state* $(\mathcal{F}, w)$, $(\mathcal{F}', w')$, *then for every universal formula* $\varphi \in \mathcal{L}^*_{a-sopml}$,

$$(\mathcal{F}', w') \vDash \varphi \quad implies \quad (\mathcal{F}, w) \vDash \varphi$$

It appears that a similar notion of games was already hinted at, but not developed, in the conclusion sections of two papers by Kuusisto [44]. Specifically, talking about MSO, the author conjectures that its expressivity can be characterized by a game that combines bisimulation games with moves that correspond to choosing subsets. Here it looks like we have a related notion of *global bisimulation game*, but in the context of SOPML, where global bisimulations contain jumps to arbitrary points in addition to standard bisimulation moves [27].

Next, simulation games can be easily generalized to bisimulation games.

Definition 86 (Bisimulation Games). *A bisimulation game* G *starting from pointed frames* $(\mathcal{F}, w)$ *and* $(\mathcal{F}', w')$ *is defined as follows. Let* $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ *be the state of the game, where* $v \in W$ *(respectively,* $v' \in W'$*) and* $\vec{U}$ *(respectively,* $\vec{U}'$*) is a possibly empty tuple of sets in* D *(respectively,* D'*).*

Then, the game proceeds according to the following rules:

1. *Either Spoiler picks a set* $U \in D$ *(respectively,* $U' \in D'$*) and Duplicator has to reply with a set* $U' \in D'$ *(respectively,* $U \in D$*) such that* $v \in U$ *iff* $v' \in U'$. *The new state of the game is* $(\mathcal{F}, v, \vec{U} \cdot U)$, $(\mathcal{F}', v', \vec{U}' \cdot U')$.

56

2. *Or, for some $a \in I$, Spoiler picks a state $u \in R_a(v)$ (respectively, $u' \in R'_a(v')$) and*
    Duplicator has to reply with state $u' \in R'_a(v')$ (respectively, $u \in R_a(v)$) such that
    for every i, $u \in U_i$ iff $u' \in U'_i$. The new state of the game is $(\mathcal{F}, u, \vec{U})$, $(\mathcal{F}', u', \vec{U}')$.

As above, if Duplicator cannot match a move of Spoiler, then Spoiler wins the game. Otherwise, Duplicator wins the game. A winning strategy is defined as usual, as a strategy whereby Duplicator can reply to all of Spoiler's moves.

Theorem 87. *Duplicator has a winning strategy for the bisimulation game starting in $(\mathcal{F}, w)$, $(\mathcal{F}', w')$ iff $(\mathcal{F}, w) \approx (\mathcal{F}', w')$.*

Proof. The proof follows the one for from Theorem 84, once we notice that the behaviour of Spoiler and Duplicator on frames $\mathcal{F}$ and $\mathcal{F}'$ is completely symmetric. As regards the $\Leftarrow$-direction, if (ω, Ω) is a bisimulation pair such that $\omega(v, v')$, then Duplicator can always reply to any Spoiler's move in state $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ with $\Omega(U_i, U'_i)$. As to the $\Rightarrow$-direction, the relations $\omega \subseteq W \times W'$ and $\Omega \subseteq D \times D'$ defined as: $\omega(v, v')$ *and* $\Omega(U_i, U'_i)$ *hold iff Duplicator has a winning strategy at state* $(\mathcal{F}, v, \vec{U})$, $(\mathcal{F}', v', \vec{U}')$ form a bisimulation pair. $\square$

Again, the existence of a winning strategy for Duplicator matches the existence of a bisimulation pair. By Theorems 74 and 87 we are then able to prove the following preservation result.

Corollary 88. *If Duplicator has a winning strategy for the bisimulation game starting in state $(\mathcal{F}, w)$, $(\mathcal{F}', w')$, then for every formula $\varphi \in \mathcal{L}_{sopml}$,*

$$(\mathcal{F}', w') \vDash \varphi \quad iff \quad (\mathcal{F}, w) \vDash \varphi$$

We conclude by discussing the two groups of preservation results. Both Theorems 68 and 74 and Corollaries 85 and 88 provide results on the preservation of (the universal fragment of) SOPML. However, (bi)simulations define global concepts, as these are defined on the whole state space $W \times W'$ and $D \times D'$; while games are played locally, as at each point in the game the players have only a local view on the frames, centred on a pair of states and a finite sequence of sets. Hence, the nature of these two notions is profoundly different. However, they are provably equivalent by Theorems 84 and 87. Moreover, both bisimulation relations and bisimulation games appear to be related to global bisimulations, a generalisation of the notion that allows for jumps to arbitrary states in the model [27]. Yet, in our case the choice does not regard arbitrary states, but rather arbitrary sets. Thus, the two notions are indeed distinct.

Finally, we envisage different applications for the two notions. For instance, (bi)simulations are typically used to prove inexpressibility results; while games can be used to show that two frames are not bisimilar, by providing moves for Spoiler to which Duplicator cannot reply. These applications are tackled in the following section.

5.4. Bisimulations and Expressivity

In this section we explore the expressivity of SOPML, also by using the (bi)simulations and (bi)simulation games introduced in Section 5.1 and 5.3. We focus on some temporal and spatial properties typically used in artificial intelligence. In what follows we say that a property P is *expressible* in a language $\mathcal{L}$ and class $\mathcal{K}$ of frames iff for some formula
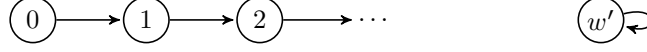
Figure 5: Frames $\mathcal{G}_1$ and $\mathcal{G}_2$. The D-components are omitted for clarity.

$\phi \in \mathcal{L}$, we have that for all $\mathcal{F} \in \mathcal{K}$, $\mathcal{F} \vDash \phi$ iff $\mathcal{F}$ has property P. Sometimes we omit either $\mathcal{L}$ or $\mathcal{K}$, whenever these are clear from the context.

First of all, consider Dedekind-completeness of a total order $\le$, i.e., a total, transitive, and antisymmetric binary relation: a totally ordered set is *Dedekind-complete* if every non-empty subset that has an upper bound, has a least upper bound. We recall that the Dedekind-completeness of the real numbers is not expressible in PML: the proof makes use of a propositional bisimulation between the structure $(\mathbb{R}, \le)$ of reals and the rationals $(\mathbb{Q}, \le)$ [1]. Thus, by using simulations we immediately obtain the following inexpressibility result.

Lemma 89. *Dedekind-completeness is not expressible in the universal fragment $\mathcal{L}_{a-sopml}$ in the class of full frames.*

Proof. Clearly, the identity relation is a simulation between structures $(\mathbb{Q}, \le)$ and $(\mathbb{R}, \le)$ seen as full frames, i.e., $(\mathbb{Q}, \le) \preceq (\mathbb{R}, \le)$, and if Dedekind-completeness were expressible as a formula ϕ in A-SOPML, $(\mathbb{R}, \le) \vDash \phi$ would imply $(\mathbb{Q}, \le) \vDash \phi$, a contradiction. □

Intuitively, formula δ fails in $(\mathbb{Q}, \le)$ since, for instance, the set $\{q \in \mathbb{Q} \mid q < \sqrt{2}\}$ is non-empty and upper bounded, and therefore satisfies the antecedent. However, it has no least upper bound to satisfy the consequent.

As a further example, we prove that neither finiteness nor infinity of the state space W are expressible in boolean frames. This is in line with the situation in PML.

Lemma 90. *In language $\mathcal{L}_{sopml}^*$ neither finiteness nor infinity are expressible in the class of boolean frames.*

Proof. Consider frame $\mathcal{G}_1 = \langle \mathbb{N}, succ, \{\mathbb{N}, \varnothing\} \rangle$ of the naturals with the successor relation and the reflexive-point frame $\mathcal{G}_2 = \langle \{w'\}, \{(w', w')\}, \{\{w'\}, \varnothing\} \rangle$ in Figure 5, which are boolean by definition of D_1 and D_2. In particular, the relations ω mapping every natural $n \in \mathbb{N}$ to w', and Ω mapping $\mathbb{N}$ to $\{w'\}$ and the empty set $\varnothing$ to itself, form a bisimulation pair. Equivalently, it is easy to see that Duplicator has a winning strategy in the game starting from state $(\mathcal{G}_1, n)$, $(\mathcal{G}_2, w')$, for every $n \in \mathbb{N}$: Duplicator has only to reply with w' to any $m \in \mathbb{N}$ chosen by Spoiler, and with $\{w'\}$ (respectively, $\varnothing$) whenever Spoiler chooses $\mathbb{N}$ (respectively, $\varnothing$). Thus, $\mathcal{G}_1$ and $\mathcal{G}_2$ validate the same formulas in SOPML. However, $\mathcal{G}_1$ is infinite while $\mathcal{G}_2$ is finite. □

To conclude our brief review of expressivity results in SOPML, we show that for the sublanguage of $\mathcal{L}_{sopml}$ without the reflexive and transitive closure operator $\square^*$, finiteness is not even expressible in full frames. For $n \in \mathbb{N}$, let $[n]$ be the set $\{0, \ldots, n\}$, $\mathcal{G}_n$ the frame $\langle [n], succ, 2^{[n]} \rangle$, and $\mathcal{G}_\mathbb{N} = \langle \mathbb{N}, succ, 2^\mathbb{N} \rangle$ the frame isomorphic to the natural numbers. Both $\mathcal{G}_\mathbb{N}$ and each $\mathcal{G}_n$ are full. Let G be the class of all frames $\mathcal{G}_n$, for $n \in \mathbb{N}$, and consider the following result.

Lemma 91. *In language $\mathcal{L}_{sopml}$ the theory $\mathsf{Th}(G)$ is a subset of $\mathsf{Th}(\mathcal{G}_\mathbb{N})$.*

58

Proof. Consider $\phi \notin \mathsf{Th}(\mathcal{G}_\mathbb{N})$ with finite modal depth $k \in \mathbb{N}$, where the modal depth is defined as in the propositional case, as the maximum nesting of modal operators [9]. We can assume without loss of generality that $(\mathcal{G}_\mathbb{N}, V, 0) \nvDash \phi$ for some assignment V. We prove that $(\mathcal{G}_k, V', 0) \nvDash \phi$, where assignment V' is such that $V'(p) = V(p) \cap [k]$ for every $p \in AP$ (since our frames are full, $V(p) \cap [k]$ is admissiable in $\mathcal{G}_k$). More precisely we prove, using induction on n, that if ψ is a subformula of ϕ of modal depth $n \leq k$, then for all ℓ with $0 \leq \ell \leq k - n$, it holds that $(\mathcal{G}_\mathbb{N}, V, \ell) \vDash \psi$ iff $(\mathcal{G}_k, V', \ell) \vDash \psi$.

We start with the case for $n = 0$. If ψ is an atom p, then $(\mathcal{G}_\mathbb{N}, V, \ell) \vDash \psi$ iff $\ell \in V(p)$, iff $\ell \in V'(p)$, iff $(\mathcal{G}_k, V', \ell) \vDash \psi$, for all $0 \leq \ell \leq k$. The cases for propositional connectives are immediate. Finally, suppose $\psi = \exists p \chi$, and that $(*)$ $(\mathcal{G}_\mathbb{N}, V, \ell) \vDash \chi$ iff $(\mathcal{G}_k, V', \ell) \vDash \chi$ for all $0 \leq \ell \leq k$. Then $(\mathcal{G}_\mathbb{N}, V, \ell) \vDash \psi$ implies that for some $U \in D$, $(\mathcal{G}_\mathbb{N}, V_U^p, \ell) \vDash \psi$. Consider $U' = U \cap [k] \in D'$. In particular, $(V_U^p)' = V_{U'}'^p$. By $(*)$, $(\mathcal{G}_k, V_{U'}'^p, \ell) \vDash \chi$, that is, $(\mathcal{G}_k, V', \ell) \vDash \psi$.

Now suppose that ψ is a subformula of ϕ of modal depth $n + 1 \leq k$, and let $0 \leq \ell \leq k - (n+1)$. Assume that $\psi = \Diamond \chi$. Then, $(\mathcal{G}_\mathbb{N}, V, \ell) \vDash \psi$ implies that $(\mathcal{G}_\mathbb{N}, V, \ell + 1) \vDash \chi$ for $1 \leq \ell + 1 \leq k - n$. By induction hypothesis $(\mathcal{G}_k, V', \ell + 1) \vDash \chi$, that is, $(\mathcal{G}_k, V', \ell) \vDash \psi$. The direction from from $(\mathcal{G}_k, V', \ell)$ to $(\mathcal{G}_\mathbb{N}, V, \ell)$ is similar, and the cases for the propositional connectives and the quantifier go as before. $\square$

Corollary 92. *In language $\mathcal{L}_{sopml}$ finiteness is not expressible on full frames.*

Proof. As a consequence of Lemma 91, $(\mathcal{G}_\mathbb{N}, V, 0) \nvDash \phi$ implies $(\mathcal{G}_k, V', 0) \nvDash \phi$, i.e., $\phi \notin \mathsf{Th}(G)$. Hence, if ϕ expressed 'being finite', then it would be valid in G, and hence also in $\mathcal{G}_\mathbb{N}$, a contradiction. Thus, finiteness is not expressible even in the class of full frames. $\square$

In this section we made use of (bi)simulations and (bi)simulations games to show that SOPML can express notions, such as Dedekind-completeness, that are not expressible in PML; whereas other properties, such as finiteness, cannot even be expressed in SOPML. Together with the remarks in Section 5.1 on 3-colorability and the existence of Hamiltonian paths, these results provide us with some interesting insight on the application of model-theoretic techniques to the analysis of the expressivity of SOPML.

In our opinion bisimulations for SOPML raise a number of interesting questions. We believe that one in particular deserves more attention. The Van Benthem theorem is a well-known result in model theory, stating that modal logic is the bisimulation-invariant fragment of first-order logic [8]. In the light of the notion of bisimulation provided above, it makes sense to ask the same question in the present context: is SOPML the bisimulation-invariant fragment of monadic second-order logic, possibly when interpreted on a particular class of frames? In [39] it is proved that the modal μ-calculus is the bisimulation-invariant fragment of MSO, but according to the standard notion of bisimulation for PML. Presently it is not clear how this result relates to the current setting. We leave this problem open for future work.

6. Conclusion

In this paper we motivated and studied (the use of) second-order propositional modal logic as a specification language for reasoning about knowledge as well as spatial and

temporal properties in artificial intelligence. Specifically, we aimed at developing proof- and model-theoretic techniques, notably complete axiomatisations and truth-preserving (bi)simulations, to support the use of SOPML in applications. In Section 2 we introduced 20 different classes of Kripke frames, according to the structure of the domain D of quantification and the features of the accessibility relations. In Section 4 we provided complete axiomatisations for some of these classes, while proving that other classes are unaxiomatisable.

Further, we introduced suitable notions of (bi)simulation and proved that they preserve the satisfaction of (universal) SOPML. Then, we defined game-theoretical counterparts to (bi)simulations and showed that the two approaches are equivalent. This is in marked contrast with [4], which put forward a different, strictly stronger notion of (bi)simulation. We remarked that, while set-theoretical (bi)simulations might be more appropriate to prove inexpressibility results, their game-theoretic counterparts might be better computationally to actually show whether two frames are bisimilar. Finally, we made use of (bi)simulations to obtain some inexpressibility results. Specifically, we showed that being finite and having a Hamiltonian path are not expressible in SOPML; while other properties, viz. topological completeness and 3-colorability, are indeed expressible. We conclude that SOPML can indeed be used as a modelling language for artificial intelligence, particularly for temporal and spatial reasoning, as well as to describe higher-level knowledge of agents, that is, the knowledge agents have about other agents' knowledge and beliefs, as shown in Section 3. In this respect, we reckon that the development of model-theoretic techniques is key for applications.

6.1. Future Research

We have presented several results about SOPML, but there are of course also a number of remaining questions for future research.

One such question is the relative expressivity of SOPML and QCTL. As mentioned in Section 1.1, the variant of SOPML with full domain of quantification is similar to one of the variants of QCTL presented in [45], with the main difference being that QCTL is based on CTL, while SOPML is based on basic modal logic. The temporal logic CTL has more operators than modal logic, and it is strictly more expressive. It is not currently clear whether QCTL is also strictly more expressive than SOPML, however.

Another direction for future research is to find fragments of SOPML that are "well-behaved". As discussed in Section 4, most of SOPML variants are unaxiomatisable on full frames, and variants that include the reachability operator $\Box^*$ are unaxiomatisable even on frames with coarser domains of quantification. It would be interesting to find fragments of SOPML that are large enough to be interesting, but that are axiomatisable or even decidable. In this respect the monodic fragment of first-order modal logic, that restrict the number of free varibles in the scope of any modal operator, looks promising [2, 34, 57]. We leave this direction for future work.

At the end of Section 5 we mentioned the interest of checking whether SOPML is the bisimulation-invariant fragment of MSO, according to the newly introduced notion of bisimulation. Such a result would provide us with a precise characterisation of the expressive power of SOPML (with respect to MSO), in the spirit of van Benthem's result for modal logic [8].

Finally, here we considered a unique domain D of quantification in any of our frames. But more elaborate forms of quantification can be taken into account, for instance, we

might consider a different quantification domain for each state in the frame. Then, we might also consider state-dependent assignments. These variants have been studied extensively in the realm of first-order modal logic [11, 25], but never addressed in SOPML. They might be of interest for specific applications.

References

[1] A. Baltag, S. Smets (Eds.), Johan van Benthem on Logic and Information Dynamics, Springer, 2014.

[2] F. Belardinelli, A. Lomuscio, Interactions between knowledge and time in a first-order logic for multi-agent systems: Completeness results, Journal of Artificial Intelligence Research 45 (2012) 1–45.

[3] F. Belardinelli, W. van der Hoek, Epistemic quantified boolean logic: Expressiveness and completeness results, in: Q. Yang, M. Wooldridge (Eds.), Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, AAAI Press, 2015, pp. 2748–2754.

[4] F. Belardinelli, W. van der Hoek, A semantical analysis of second-order propositional modal logic, in: Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-16), AAAI Press, 2016, pp. 886–892.

[5] F. Belardinelli, H. van Ditmarsch, W. van der Hoek, Second-order propositional announcement logic, in: C. Jonker, S. Marsella, J. Thangarajah, K. Tuyls (Eds.), Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, ACM, 2016, pp. 635–643.
URL http://dl.acm.org/citation.cfm?id=2937019

[6] J. van Benthem, Modal Logic and Classical Logic, Vol. 3 of Monographs in Philosophical Logic and Formal Linguistics, Bibliopolis, 1988.
URL https://books.google.fr/books?id=j8AVAQAACAAJ

[7] J. van Benthem, A note on modal formulae and relational properties, The Journal of Symbolic Logic 40 (1) (1975) 55–58.
URL http://www.jstor.org/stable/2272270

[8] J. van Benthem, Modal correspondence theory, Ph.D. thesis, University of Amsterdam (1976).

[9] P. Blackburn, M. de Rijke, Y. Venema, Modal Logic, Vol. 53 of Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, 2001.

[10] P. Blackburn, J. van Benthem, F. Wolter (Eds.), Handbook of Modal Logic, Elsevier, Amsterdam, the Netherlands, 2007.

[11] T. Brauner, S. Ghilardi, First-order modal logic, in: P. Blackburn, J. van Benthem, F. Wolter (Eds.), Handbook of Modal Logic, Elsevier, 2007, pp. 549–620.

[12] R. A. Bull, On modal logic with propositional quantifiers, The journal of symbolic logic 34 (1969) 257–263.

[13] B. ten Cate, Expressivity of second order propositional modal logic., Journal of Philosophical Logic 35 (2) (2006) 209–223.
URL http://dblp.uni-trier.de/db/journals/jphil/jphil35.html#Cate06

[14] A. Church, A note on the entscheidungsproblem, Journal of Symbolic Logic 1 (1) (1936) 40–41.

[15] E. M. Clarke, O. Grumberg, D. E. Long, Model Checking and Abstraction, ACM Transactions on Programming Languages and Systems 16 (5) (1994) 1512–1542.

[16] P. B. Courcelle, D. J. Engelfriet, Graph Structure and Monadic Second-Order Logic: A Language-Theoretic Approach, 1st Edition, Cambridge University Press, New York, NY, USA, 2012.

[17] H. P. van Ditmarsch, W. van der Hoek, B. P. Kooi, Knowing more - from global to local correspondence, in: C. Boutilier (Ed.), IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, 2009, pp. 955–960.
URL http://ijcai.org/papers09/Papers/IJCAI09-162.pdf

[18] H. P. van Ditmarsch, W. van der Hoek, B. P. Kooi, Reasoning about local properties in modal logic, in: L. Sonenberg, P. Stone, K. Tumer, P. Yolum (Eds.), Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), IFAAMAS, 2011, pp. 711–718.
URL `http://portal.acm.org/citation.cfm?id=2031719&CFID=54178199&CFTOKEN=61392764`

[19] H. P. van Ditmarsch, W. van der Hoek, B. P. Kooi, Local properties in modal logic, Artificial Intelligence 187 (2012) 133–155.

[20] H. Ebbinghaus, J. Flum, W. Thomas, Mathematical Logic, Springer, 1994.

[21] R. Fagin, J. Halpern, Y. Moses, M. Vardi, Reasoning About Knowledge, The MIT Press, 1995.

[22] K. Fine, Propositional quantifiers in modal logic, Theoria 36 (3) (1970) 336–346.

[23] T. French, Bisimulation quantifiers for modal logic, Ph.D. thesis, University of Western Australia (2006).

[24] D. M. Gabbay, An irreflexivity lemma with applications to axiomatizations of conditions on tense frames, in: U. Mönnich (Ed.), Aspects of Philosophical Logic: Some Logical Forays into Central Notions of Linguistics and Philosophy, Springer Netherlands, Dordrecht, 1981, pp. 67–89. `doi:10.1007/978-94-009-8384-7_3`.
URL `http://dx.doi.org/10.1007/978-94-009-8384-7_3`

[25] J. Garson, Quantification in modal logic, in: D. Gabbay, F. Guenthner (Eds.), Handbook of Philosophical Logic, Vol. 3, Reidel, 2001, pp. 267–323.

[26] R. I. Goldblatt, First-order definability in modal logic, The Journal of Symbolic Logic 40 (1) (1975) 35–40.
URL `http://www.jstor.org/stable/2272267`

[27] V. Goranko, M. Otto, Model theory of modal logic, in: P. Blackburn, J. van Benthem, F. Wolter (Eds.), Handbook of Modal Logic, Elsevier, 2007, pp. 249–329.

[28] E. Grädel, P. Kolaitis, L. Libkin, M. Marx, J. Spencer, M. Vardi, Y. Venema, S. Weinstein, Finite Model Theory and Its Applications, Texts in Theoretical Computer Science. An EATCS Series, Springer Berlin Heidelberg, 2007.
URL `https://books.google.fr/books?id=WXSqQKPKWKsC`

[29] J. Halpern, Y. Moses, Knowledge and common knowledge in a distributed environment, Journal of the ACM 37 (3) (1990) 549–587.

[30] D. Harel, D. Kozen, J. Tiuryn, Dynamic Logic, MIT Press, 2000.

[31] F. van Harmelen, V. Lifschitz, B. Porter, Handbook of Knowledge Representation, Elsevier Science, San Diego, USA, 2007.

[32] M. Hennessy, R. Milner, On observing nondeterminism and concurrency, in: Proceedings of the 7th Colloquium on Automata, Languages and Programming, Springer-Verlag, London, UK, UK, 1980, pp. 299–309.
URL `http://dl.acm.org/citation.cfm?id=646234.758793`

[33] J. Hintikka, Knowledge and belief: an introduction to the logic of the two notions, Contemporary philosophy, Cornell University Press, 1962.
URL `http://books.google.fr/books?id=N28OAAAAIAAJ`

[34] I. Hodkinson, F. Wolter, M. Zakharyaschev, Monodic fragments of first-order temporal logics: 2000-2001 a.d., in: R. Nieuwenhuis, A. Voronkov (Eds.), LPAR, Vol. 2250 of Lecture Notes in Computer Science, Springer, 2001, pp. 1–23.

[35] W. van der Hoek, J.-J. C. Meyer, Making some issues of implicit knowledge explicit, International Journal of Foundations of Computer Science 3 (2) (1992) 193–224.

[36] W. van der Hoek, M. Pauly, Modal logic for games and information, in: Blackburn et al. [10], pp. 1077–1148.

[37] W. van der Hoek, M. Wooldridge, Multi-agent systems, in: F. van Harmelen, V. Lifschitz, B. Porter (Eds.), Handbook of Knowledge Representation, Elsevier, 2008, p. 887928.

[38] W. Holliday, A note on algebraic semantics for s5 with propositional quantifiers, Tech. rep., UC Berkeley: Group in Logic and the Methodology of Science, to appear in Notre Dame Journal of Formal Logic (2017).
URL `http://escholarship.org/uc/item/303338xr`

[39] D. Janin, I. Walukiewicz, On the expressive completeness of the propositional mu-calculus with respect to monadic second order logic, in: U. Montanari, V. Sassone (Eds.), Proceedings of the 7th International Conference on Concurrency Theory (CONCUR '96), Springer, 1996, pp. 263–277. `doi:10.1007/3-540-61604-7_60`.
URL `http://dx.doi.org/10.1007/3-540-61604-7_60`

62

[40] M. Kaminski, M. L. Tiomkin, The expressive power of second-order propositional modal logic., Notre Dame Journal of Formal Logic 37 (1) (1996) 35–43.
URL http://dblp.uni-trier.de/db/journals/ndjfl/ndjfl37.html#KaminskiT96

[41] D. Kaplan, S5 with quantifiable propositional variables, Journal of Symbolic Logic 35 (1970) 355.

[42] L. B. Kuijer, Arbitrary arrow update logic with common knowledge is neither re nor co-re, to appear. (2017).

[43] A. Kuusisto, A modal perspective on monadic second-order alternation hierarchies, in: Advances in Modal Logic, 2008, p. 231247.

[44] A. Kuusisto, Second-order propositional modal logic and monadic alternation hierarchies, Annals of Pure and Applied Logic 166 (1) (2015) 1 – 28. doi:http://doi.org/10.1016/j.apal.2014.08.003.
URL http://www.sciencedirect.com/science/article/pii/S0168007214000876

[45] F. Laroussinie, N. Markey, Quantified CTL: Expressiveness and complexity, Logical Methods in Computer Science 10 (4:17). doi:10.2168/LMCS-10(4:17)2014.
URL http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/LM-lmcs14.pdf

[46] L. Libkin, Elements of Finite Model Theory, Texts in Theoretical Computer Science. An EATCS Series, Springer Berlin Heidelberg, 2004.
URL https://books.google.fr/books?id=zsJlEK4nK7sC

[47] Z. Manna, A. Pnueli, The Temporal Logic of Reactive and Concurrent Systems, Vol. 1, Springer-Verlag, 1992.

[48] E. Mares, R. Goldblatt, An alternative semantics for quantified relevant logic, The Journal of Symbolic Logic 71 (1) (2006) 163–187.

[49] J.-J. C. Meyer, W. v. d. Hoek, Epistemic Logic for AI and Computer Science, Cambridge University Press, New York, NY, USA, 1995.

[50] W. Quine, Quantifiers and propositional attitudes, The Journal of Philosophy 53 (5) (1956) 177 187.

[51] F. Roelofsen, Distributed knowledge, Journal of Applied Non-Classical Logic 17 (2) (2007) 255–273.

[52] A. P. Sistla, M. Y. Vardi, P. Wolper, The complementation problem for büchi automata with applications to temporal logic (extended abstract), in: W. Brauer (Ed.), Automata, Languages and Programming, 12th Colloquium, Nafplion, Greece, July 15-19, 1985, Proceedings, Vol. 194 of Lecture Notes in Computer Science, Springer, 1985, pp. 465–474. doi:10.1007/BFb0015772.
URL https://doi.org/10.1007/BFb0015772

[53] A. P. Sistla, Theoretical issues in the design and verification of distributed systems, Ph.D. thesis, Carnegie-Mellon University, Cambridge, MA, USA, aAI8403047 (1983).

[54] A. M. Turing, On computable numbers, with an application to the entscheidungsproblem, Proceedings of the London Mathematical Society s2-42 (1) (1937) 230–265. doi:10.1112/plms/s2-42.1.230.
URL http://dx.doi.org/10.1112/plms/s2-42.1.230

[55] J. Vaananen, Second-order logic and foundations of mathematics, The Bulletin of Symbolic Logic 7 (4) (2001) 504–520.
URL http://www.jstor.org/stable/2687796

[56] Y. Venema, Derivation rules as anti-axioms in modal logic, The Journal of Symbolic Logic 58 (1993) 1003–1034. doi:10.2307/2275109.
URL http://journals.cambridge.org/article_S0022481200021010

[57] F. Wolter, M. Zakharyaschev, Axiomatizing the monodic fragment of first-order temporal logic., Annals of Pure and Applies Logic 118 (1-2) (2002) 133–145.

Appendix A. Selected Proofs

Lemma 12.

1. Let ϕ be a formula in $\mathcal{L}^*_{sopml}$ and $\mathcal{F}$ a frame in $\mathcal{K}_{all}$. If assignments V and V' coincide on $fr(\phi)$, then

$$(\mathcal{F}, V, w) \vDash \phi \quad iff \quad (\mathcal{F}, V', w) \vDash \phi$$

2. Recall that $X = \{ap, pl, ml, sopml\}$ and $\frown = \{(ap, all), (pl, bool), (ml, modal), (sopml, full)\}$. Let $x \in X$. Then,

   (a) For every $\psi \in \mathcal{L}^*_x$ and model $\mathcal{M}$ over $\mathcal{F} \in \mathcal{K}_{\widehat{x}}$, we have $[\![\psi]\!]_{\mathcal{M}} \in D$.

   (b) If $\mathcal{F} \in \mathcal{K}_{\widehat{x}}$ and $\psi \in \mathcal{L}^*_x$ is free for p in ϕ, then

$$(\mathcal{F}, V^p_{[\![\psi]\!]_{\langle \mathcal{F}, V \rangle}}, w) \vDash \phi \quad iff \quad (\mathcal{F}, V, w) \vDash \phi[p/\psi]$$

Proof. The proofs are by induction on the structure of $\phi \in \mathcal{L}^*_{sopml}$.

1. If $\phi = p$, then $fr(\phi) = \{p\}$ and $(\mathcal{M}, w) \vDash \phi$ iff $w \in V(p) = V'(p)$, iff $(\mathcal{M}', w) \vDash \phi$. The inductive cases for the propositional connectives are immediate.
   If $\phi = \Box_a \psi$, then $(\mathcal{M}, w) \vDash \phi$ iff for all $w' \in R_a(w)$, $(\mathcal{M}, w') \vDash \psi$. Since $fr(\phi) = fr(\psi)$, V and V' coincide on $fr(\psi)$ as well, and by induction hypothesis for all $w' \in R_a(w)$, $(\mathcal{M}', w') \vDash \psi$, i.e., $(\mathcal{M}', w) \vDash \phi$. The case for $\phi = \Box^* \psi$ is similar.
   If $\phi = \forall p \psi$, then $(\mathcal{M}, w) \vDash \phi$ iff for any $U \in D$, $(\mathcal{M}^p_U, w) \vDash \psi$. Since $fr(\phi) = fr(\psi) \smallsetminus \{p\}$, V^p_U and V'^p_U coincide on $fr(\psi)$, and by induction hypothesis $(\mathcal{M}'^p_U, w) \vDash \psi$. Since U has been chosen arbitrarily, this is the case iff $(\mathcal{M}', w) \vDash \phi$.

2a The case for $x = ap$ is immediate, as assignments are functions in D. Hence, $V(p) \in D$ for every $p \in AP$.
   The case for $x = pl$, follows from equalities $[\![\neg\psi]\!] = W \smallsetminus [\![\psi]\!]$, $[\![\psi \wedge \psi']\!] = [\![\psi]\!] \cap [\![\psi']\!]$, $[\![\psi \vee \psi']\!] = [\![\psi]\!] \cup [\![\psi']\!]$ and the fact that D is a boolean algebra.
   As for $x = ml$, notice that $[\![\Box_a \psi]\!] = [a]([\![\psi]\!])$, $[\![\Box^* \psi]\!] = [\ ]^*([\![\psi]\!])$, and D is a boolean algebra closed under operators $[a]$ and $[\ ]^*$.
   The case for $x = sopml$, is immediate, as $[\![\psi]\!] \subseteq W$ for every $\psi \in \mathcal{L}^*_{sopml}$.

2b Let us first consider $x = ap$. If ϕ is an atom r, $(\mathcal{M}^p_{V(q)}, w) \vDash \phi$ iff $w \in V^p_{V(q)}(r)$, iff $w \in V(r)$ whenever $r \neq p$ or $w \in V(q)$ for $r = p$. In both cases $(\mathcal{M}, w) \vDash \phi[p/q]$.
   The inductive cases for propositional connectives and modal operators are immediate, as these simply commute with substitution.
   If $\phi = \forall r \varphi$ for $r \neq p$, then $(\mathcal{M}^p_{V(q)}, w) \vDash \phi$ iff for every $U \in D$, $((\mathcal{M}^p_{V(q)})^r_U, w) \vDash \varphi$. Since $r \neq p$ and q is free for p in ϕ, we have $q \neq r$ and assignment $(V^p_{V(q)})^r_U$ is equal to $(V^r_U)^p_{V^r_U(q)}$. As a consequence, we obtain $((\mathcal{M}^r_U)^p_{V^r_U(q)}, w) \vDash \varphi$, i.e., $(\mathcal{M}^r_U, w) \vDash \varphi[p/q]$ by induction hypothesis. But this means that $(\mathcal{M}, w) \vDash \forall r(\varphi[p/q]) = (\forall r \varphi)[p/q]$.
   As regards cases $x = pl, ml, sopml$, we make use of item 1. We only prove the inductive step for $\phi = \forall r \varphi$, with $r \neq p$, the other cases being similar to the case for $x = ap$ above. Observe that $(\mathcal{M}^p_{[\![\psi]\!]}, w) \vDash \phi$ iff for every $U \in D$, $((\mathcal{M}^p_{[\![\psi]\!]})^r_U, w) \vDash \varphi$.

64

Since $r \neq p$ and ψ is free for p in ϕ, we have $r \notin fr(\psi)$, and by item 1 above, $\llbracket \psi \rrbracket_\mathcal{M} = \llbracket \psi \rrbracket_{\mathcal{M}_U^r}$. Therefore assignment $(V_{\llbracket \psi \rrbracket}^p)_U^r$ is equal to $(V_U^r)_{\llbracket \psi \rrbracket_{\mathcal{M}_U^r}}^p$. Hence, we obtain $((\mathcal{M}_U^r)_{\llbracket \psi \rrbracket_{\mathcal{M}_U^r}}^p, w) \vDash \varphi$, i.e., $(\mathcal{M}_U^r, w) \vDash \varphi[p/\psi]$ by induction hypothesis. But this means that $(\mathcal{M}, w) \vDash \forall r(\varphi[p/\psi]) = (\forall r\varphi)[p/\psi]$.

$\square$

Theorem 23. *For every intended* LPML *model* $\mathcal{M}$, $w \in \mathcal{M}$, *and formula* φ *in* LPML, *we have*

$$(\mathcal{M}, w) \vDash \varphi \quad \textit{iff} \quad (\mathcal{M}, w) \vDash \tau(\varphi)$$

Proof. We will only prove the crucial clause

$$(\mathcal{M}, w) \vDash \forall \vec{p}\theta(\vec{a}, \vec{p}) \quad \text{iff} \quad (\mathcal{M}, w) \vDash \boxdot(\vec{a})$$

Before doing that, let us first show what it means for the following specific case: $\theta(a, b, p) = \square_a p \to \square_b p$, $\boxdot(a, b) = Sup(a, b)$, and $\Theta(a, b, x) = \forall y(R_b(x, y) \to R_a(x, y))$, also written as $R_b(x) \subseteq R_a(x)$.

$\Leftarrow$ Since $(\mathcal{M}, w) \vDash Sup(a, b)$, we have that $R_b(x) \subseteq R_a(x)$ holds in $(\mathcal{M}, w)$, and hence in $(\mathcal{F}, w)$. Since $\square_a p \to \square_b p$ locally defines $R_b(x) \subseteq R_a(x)$, we have $(\mathcal{F}, w) \vDash (\square_a p \to \square_b p)$, and in particular $(\mathcal{F}, w) \vDash \forall p(\square_a p \to \square_b p)$. Since $\forall p(\square_a p \to \square_b p)$ is a sentence, we obtain $(\mathcal{M}, w) \vDash \forall p(\square_a p \to \square_b p)$. $\Rightarrow$ Now suppose that $(\mathcal{M}, w) \not\vDash Sup(a, b)$. Then $(\mathcal{F}, w) \not\vDash Sup(a, b)$. Since $\square_a p \to \square_b p$ locally defines $R_b(x) \subseteq R_a(x)$, we know that $(\mathcal{F}, w) \not\vDash \square_a p \to \square_b p$, and since $\mathcal{F}$ is full, for some assignment V', we have $(\mathcal{F}, V', w) \not\vDash \square_a p \to \square_b p$, that is, $(\mathcal{M}, w) \not\vDash \forall p(\square_a p \to \square_b p)$.

As for the general case: $\Leftarrow$ Since $(\mathcal{M}, w) \vDash \boxdot(\vec{a})$, we have that $\Theta(\vec{a}, x)$ holds in $(\mathcal{M}, w)$, and hence in $(\mathcal{F}, w)$ (note that $\Theta \in \mathcal{L}_{fo}^1$ only talks about what is accessible from what). Since $\theta(\vec{a}, \vec{p})$ locally defines $\Theta(\vec{a}, x)$, we have $(\mathcal{F}, w) \vDash \theta(\vec{a}, \vec{p})$, and in particular $(\mathcal{F}, w) \vDash \forall \vec{p}\theta(\vec{a}, \vec{p})$. Since $\forall \vec{p}\theta(\vec{a}, \vec{p})$ is a sentence, $(\mathcal{M}, w) \vDash \forall p\theta(\vec{a}, \vec{p})$. $\Rightarrow$ Suppose that $(\mathcal{M}, w) \not\vDash \boxdot(\vec{a})$. Then, $(\mathcal{F}, w) \not\vDash \boxdot(\vec{a})$, and therefore $(\mathcal{F}, w) \not\vDash \Theta(\vec{a}, x)$. Since $\theta(\vec{a}, \vec{p})$ locally defines $\Theta(\vec{a}, x)$, we know that $(\mathcal{F}, w) \not\vDash \theta(\vec{a}, \vec{p})$, and since $\mathcal{F}$ is full, for some assignment V', we have $(\mathcal{F}, V', w) \not\vDash \theta(\vec{a}, \vec{p})$, that is, $(\mathcal{M}, w) \not\vDash \forall \vec{p}\theta(\vec{a}, \vec{p})$. $\square$

Lemma 25. *Consider formulas* $\varphi_i \in \mathcal{L}_{sopml}$ *and* $\Theta_i \in \mathcal{L}_{fo}^1$ *in Example 24, for* $i = 1, \ldots, 5$. *Let x be the only free variable in Θ_i and assume $\rho(x) = w$. Assume $\mathcal{F}$ is a full frame, then,*

$$(\mathcal{F}, w) \vDash \varphi_i \quad \textit{iff} \quad (\mathcal{F}, \rho) \vDash \Theta_i$$

Proof. All items are relatively immediate. We only proved the first and the last ones. Rather than $(\mathcal{F}, \rho) \vDash \Theta$, we will also write $\mathcal{F} \vDash \Theta(w)$, and say that Θ holds for w in $\mathcal{F}$.

1. Suppose $\mathcal{F}$ is full and irreflexive at w, that is, $\neg R_a(w, w)$, then clearly $(\mathcal{F}, w) \vDash \exists p(\square_a p \wedge \neg p)$, by considering the assignment $V(p) = R_a(w)$ for which $w \notin V(p)$. As to the converse, suppose that $\mathcal{F}, w \vDash \exists p(\square_a p \wedge \neg p)$. Hence, for every model $\mathcal{M}$ on $\mathcal{F}$, $(\mathcal{M}, w) \vDash \exists p(\square_a p \wedge \neg p)$, i.e., $(\mathcal{M}, w) \not\vDash \forall p(\square_a p \to p)$. However, by Lemma 29 below, this is the case iff $R_a(w, w)$ does not hold. Hence, $\Theta_1(w)$ holds in $\mathcal{F}$.

65

2. Suppose that $\Theta_5(w)$ holds in $\mathcal{F}$ and let V be such that $(\mathcal{F}, V, w) \vDash \Box_c p$. It is easy to check that $(\mathcal{F}, V^q_{R_a(w)}, w) \vDash \Box_a q \land \Box_b(q \to p)$. In words, if we modify V in such a way that q becomes true in exactly w's a-successors, then for every b-successor of w that satisfies q (note that this successor must then also be an a-successor), p must be true. Conversely, suppose that $\Theta_5(w)$ does not hold, i.e., for some $v \in W$, we have $R_a(w, v)$ and $R_b(w, v)$, but not $R_c(w, v)$. We now show that $(\mathcal{F}, w) \vDash \neg\varphi_5 = \exists p(\Box_c p \land \forall q(\Box_a q \to \Diamond_b(q \land \neg p)))$. The assignment V such that $V(p) = R_c(w)$ is a witness for this: if p is exactly true in the c-successors of w, then it is false in v, so whenever $\Box_a q$ is true in w, we have that $q \land \neg p$ holds in v, and hence $\Diamond_b(q \land \neg p)$ holds in w.

$\square$

Lemma 29. *For every model* $\mathcal{M} = \langle \mathcal{F}, V \rangle$, *world* $w \in W$, *and formula* $\psi \in \mathcal{L}^*_{sopml}$,

$$(\mathcal{M}, w) \vDash \psi \quad iff \quad (\mathcal{F}, \rho) \vDash ST_x(\psi)$$

whenever $\rho(x) = w$ *and* $\rho(P_i) = V(p_i)$.

Proof. The proof is by induction on the structure of ψ. Since the steps for modal logic formulas is standard, we only show the case for the quantifier. For $\psi = \forall p \phi$, $(\mathcal{M}, w) \vDash \psi$ iff for all $U \in D$, $(\mathcal{M}^p_U, w) \vDash \phi$, that is, $(\mathcal{F}, \rho') \vDash ST_x(\phi)$ by induction hypothesis, for ρ' that coincides with ρ but $\rho'(P) = U$. However, this means that $(\mathcal{F}, \rho^P_U) \vDash ST_x(\phi)$, i.e., $(\mathcal{F}, \rho) \vDash \forall P(ST_x(\phi)) = ST_x(\psi)$. $\square$

Recall that in Section 4.2, we stated the following theorem without proof:

Theorem 63. *If* $|I| \geq 2$, *then the validities in* SOPML* *over modal epistemic frames are not recursively enumerable.*

In particular, SOPML* *is not axiomatisable on the class of modal epistemic frames.*

Here, we provide the proof. The proof strategy that we use is very similar to the one used in the proof of Theorem 58: we define formulas ξ_{grid}, ξ_{sane} and ξ_T that serve the same purpose as ψ_{grid}, ψ_{sane} and ψ_T, respectively.

The main difference lies in how we define a grid, now that we use two-agent S5 as opposed to single agent K. This time, we use the following pattern: each point $(n, m) \in \mathbb{Z} \times \mathbb{Z}$ is represented not by a single world, but instead by (at least) five different worlds that are related to each other by the relation $R(a)$. One of these five worlds satisfies the propositional atom *center*, the other four worlds satisfy the atoms *left*, *right*, *up* and *down*, respectively. The *left* world of (n, m) is then related by $R(b)$ to the *right* world of $(n-1, m)$, and similarly for the other directions; see also the following diagram.
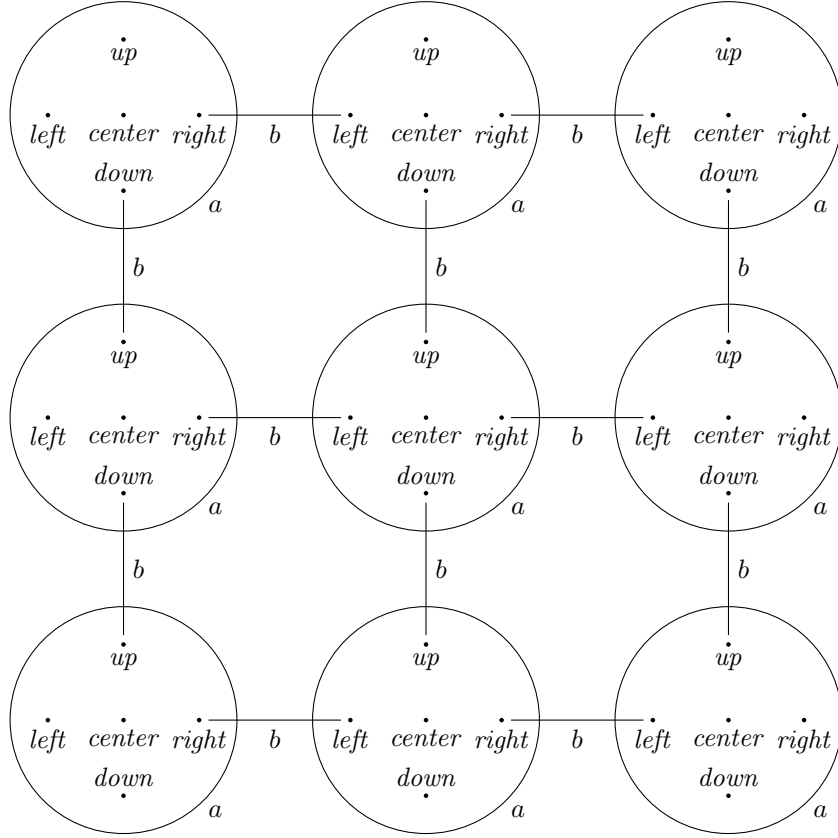
As before, this allows us to define $\Box_x$ and $\Diamond_x$ (for $x \in dir$) as abbreviations:

$$\Box_x \phi := \Box_a (x \to \Box_b (\neg x \to \Box_a (center \to \phi)))$$
$$\Diamond_x \phi := \Diamond_a (x \wedge \Diamond_b (\neg x \wedge \Diamond_a (center \wedge \phi)))$$

Then we define a grid in almost exactly the same way as the K-case:

$$\xi_{grid} := \square^* \big(labels \wedge direction \wedge same \wedge remain \wedge \bigwedge_{x \in dir} unique_x \wedge$$
$$\bigwedge_{(x,y) \in inv\_dir} inverse_{xy} \wedge \bigwedge_{(x,y) \in perp\_dir} commute_{xy} \big)$$
$$labels := \bigvee_{i \in L} (i \wedge \bigwedge_{j \in L \setminus \{i\}} \neg j) \wedge \bigwedge_{l \in L} \lozenge_a l$$
$$directions := ((left \vee right) \to (\lozenge_b left \wedge \lozenge_b right \wedge \square_b(left \vee right))) \wedge$$
$$((up \vee down) \to (\lozenge_b up \wedge \lozenge_b down \wedge \square_b(up \vee down)))$$
$$same := \bigwedge_{l \in L} \forall p((l \wedge p) \to (\square_a(l \to p) \wedge \square_b(l \to p))$$
$$remain := \bigwedge_{c \in Ag \setminus \{a,b\}} \forall p(p \leftrightarrow \square_c p)$$
$$unique_x := \forall p(\lozenge_x p \to \square_x p))$$
$$inverse_{xy} := \forall p(p \to \square_x \square_y p)$$
$$commute_{xy} := \forall p(\lozenge_x \lozenge_y p \to \square_y \square_x p)$$

where dir, inv_dir and $perp\_dir$ are as before, and $L := \{left, right, up, down, center\}$.

Note that $unique_x$, $inverse_{xy}$ and $commute_{xy}$ are identical to their counterparts in the proof of Theorem 58, but the other subformulas are different.

If $\mathcal{M}, w_0 \vDash \xi_{grid}$, then we associate the worlds of $\mathcal{M}$ with the points of $\mathbb{Z} \times \mathbb{Z}$ in the following way: (i) w_0 and all its a-successors represent $(0,0)$, (ii) if w represents (n,m), $\mathcal{M}, w \vDash left$, w' is a b-successor of w and $\mathcal{M}, w' \vDash right$, then w' represents $(n-1, m)$, (iii) similarly for the other directions and (iv) if w represents (n, m) then so does every c-successor of (n, m) for every $c \notin \{a, b\}$.

As before, we show that every point (n, m) is represented by at least one world and that each world represents at least one point. Since every point is represented by at least one *center* world, one *right* world, one *left* world, one *up* world and one *down* world, we obviously cannot guarantee that all the worlds representing a point are indistinguishable. We will show, however, that all *center* worlds representing (n, m) are modally indistinguishable from one another.

- *labels* guarantees that every world (i) satisfies exactly one of the labels from L and (ii) has a successor that satisfies l for every $l \in L$.

- *directions* guarantees that every *left* world is paired with a *right* world through the relation $R(b)$, and similarly for the other directions. Due to how we defined $\lozenge_x$ and $\square_x$ as abbreviations, this means that every *center* world has and x-successor that is also a *center* world, for every $x \in dir$. That, in turn, implies that every point (n, m) is represented by at least one world.

- *same* says that for every $U \in D$, if w is in U then so are all of its a- and b-successors that share the same label. Since we are working with a modal domain of quantification, this implies that every label is unique (up to modal indistinguishability) in its a- and b-equivalence classes. Note that, together with the fact that *left, right, up*

68

and *down* worlds occur only in pairs, this implies that whenever a world w represents some point (n,m), then every a- or b-successor of w also represents some point (n',m'). So every world in the generated submodel represents some point (n,m).

- *remain* says that for any agent c other than a and b, w is modally indistinguishable from its c-successors. This implies that if w and w' represent the same point (n,m) due to rule (iv), then w and w' are modally indistinguishable.

- The formulas $unique_x$, $inverse_{xy}$ and $commute_{xy}$, for the relevant x and y, guarantee that if w and w' represent the same point (n,m) due to rules (ii) and (iii), then w and w' are modally indistinguishable.

We have now shown that every point (n,m) is represented, that every world represents a point and that all worlds representing a single point are modally indistinguishable.

Variants ξ_{sane} and ξ_T of ψ_{sane} and ψ_T can then be defined. The only required modification is that in ξ_{sane} and ξ_T we only put requirements on *center* world, e.g., the subformula $\bigvee_{s \in states}(s \wedge \bigwedge_{s' \in states \setminus \{s\}} \neg s')$ of ψ_{sane} should be replaced with $center \rightarrow \bigvee_{s \in states}(s \wedge \bigwedge_{s' \in states \setminus \{s\}} \neg s')$ in ξ_{sane}. Since these modifications are rather trivial, we do not list them here in detail.

Overall, if we define $\zeta_T := \xi_{grid} \wedge \xi_{sane} \wedge \xi_T \wedge s_0 \wedge pos$, then $\mathcal{M}, w \vDash \zeta_T$ implies that $\mathcal{M}$ encodes the execution of T. It follows that T is non-halting if and only if $\mathcal{M}, w \vDash \zeta_T \rightarrow \Box^*(center \rightarrow \neg s_{end})$. In particular, this implies that the valid formulas of SOPML* over modal S5 frames are not recursively enumerable.

Theorem 74. *If $w \approx w'$, then for every formula $\varphi \in \mathcal{L}^*_{sopml}$,*

$$(\mathcal{F}, w) \vDash \varphi \quad iff \quad (\mathcal{F}', w') \vDash \varphi.$$

Proof. We prove the implication from right to left, the opposite direction being symmetric. If $w \approx w'$ then $\omega(w, w')$ holds for some bisimulation pair (ω, Ω). As above, we show by induction on φ that if $(\mathcal{F}, V, w) \not\vDash \varphi$ for some assignment V, then $(\mathcal{F}', \Omega(V), w') \not\vDash \varphi$, where $\Omega(V)$ is any assignment such that for every $p \in AP$, $(\Omega(V))(p) = U'$ with $\Omega(V(p), U')$. Since ω is a simulation relation in particular, the base cases for $\varphi = p$ and $\varphi = \neg p$ are as in Theorem 68, as well as the inductive cases for propositional connectives and $\varphi = \Box_a \psi$, $\varphi = \Box^* \psi$, and $\varphi = \forall p \psi$.

For $\varphi = \Diamond_a \psi$, $(\mathcal{F}', \Omega(V), w') \vDash \varphi$ iff for some $v' \in R'_a(w')$, $(\mathcal{F}', \Omega(V), v') \vDash \psi$. By bisimulation, for some $v \in R_a(w)$, $\omega(v, v')$. In particular, $(\mathcal{F}, V, v) \vDash \psi$ by induction hypothesis. That is, $(\mathcal{F}, V, w) \vDash \varphi$. The case for $\varphi = \Diamond^* \psi$ is similar.

For $\varphi = \exists p \psi$, $(\mathcal{F}', \Omega(V), w') \vDash \varphi$ iff for some $U' \in D'$, $(\mathcal{F}', (\Omega(V))^p_{U'}, w') \vDash \psi$. Now consider $U \in D$ such that $\Omega(U, U') \in D$. In particular, assignments $(\Omega(V))^p_{U'}$ and $\Omega(V^p_U)$ coincides. Hence, $(\mathcal{F}', \Omega(V^p_U), w') \vDash \psi$, and by induction hypothesis, $(\mathcal{F}, V^p_U, w) \vDash \psi$ for $U \in D$, that is, $(\mathcal{F}, V, w) \vDash \varphi$. $\qquad\square$