On Modal Logic, IF Logic, and IF Modal Logic

Tero Tulenheimo and Merlijn Sevenster

ABSTRACT. 'Structurally determined IF modal logic' ($\mathcal{L}_{\mathbf{SD}}$) is introduced, and a study of its computational properties is undertaken. We argue that the results we prove have a general interest: The properties of our modal-like logic go against various tenets typically held with regard to modal logic. Notably, $\mathcal{L}_{\mathbf{SD}}$ has **PSPACE**-complete satisfiability problem but does not admit of a truth-preserving translation into the guarded fragment of first-order logic. Beyond merely laying down a number of technical results, we also wish to address the general issue of what are the core properties of modal logics.

1 Introduction

In the literature several logics have been presented under the heading *independence friendly modal logic* [4, 5, 13, 17, 18]. None of them, however, makes any claim of being *the* independence friendly modal logic. We will not do so either. Instead, we introduce a framework where one can play around, developing one's modal logic. As an elaborate illustration, we take up the development of one particular independence friendly modal logic within that framework, and study some of its metalogical properties.

The idea underlying our framework is the following: We have quite good understanding of modal logics ('modal languages' in the sense of [3, Def. 1.12]), first-order logic (**FO**), and independence friendly first-order logic (**IF**). Also their relationships are well understood: modal logics can be translated into **FO** via a *standard translation*, and one obtains the language of **IF** by applying some kind of *IF-procedure* to the formulas of **FO**. Now modal logics are often seen as fragments of **FO**. Similarly, independence friendly modal logic can be viewed as a language (a fragment of **IF**) that results from applying an IF-procedure to a standard translation of some modal logic. Crucially, the latter statement is underspecified regarding the IF-procedure, the standard translation and the specific modal logic $\mathbf{ML}(\tau, \Phi)$, with τ a modal similarity type and Φ a set of propositional atoms. All variations that one can think of, span the framework that we mean to introduce. Graphically, for any modal logic **ML** and any corresponding standard translation ST, an independence friendly modal logic **IFML** that fits within our framework is subject to the following relationships:

ML	is equally expressive as	$ST(\mathbf{ML})$	subset of	**FO**
		$\Downarrow_{\mathrm{IF}}$		$\Downarrow_{\mathrm{IF}}$
IFML	is equally expressive as	$\mathbf{IF}(ST(\mathbf{ML}))$	subset of	**IF**

where $\Downarrow_{\mathrm{IF}}$ is some kind of IF-procedure. Let us go through a small example, to get intuitions straight. Fix a formula $\phi := \Box(\Diamond p \vee q)$ from modal logic. The truth condition of ϕ at a state w is specified in **FO** by the following formula (obtained by the standard translation explained in any standard textbook on modal logic): $\forall x(Rx_0x \rightarrow (\exists y(Rxy \wedge Py) \vee Qx))$, assigning the state w to x_0. Applying Hintikka's IF-procedure [11] we get, among others, the formula $\forall x(Rx_0x \rightarrow ((\exists y/x)(Rxy \wedge Py) \vee Qx))$. This formula will, then, belong to $\mathbf{IF}(ST(\mathbf{ML}))$. Reading such formulas can be tedious; in fact, it is not clear at first sight whether a given string is a formula of $\mathbf{IF}(ST(\mathbf{ML}))$. For this reason, it will be convenient to introduce an independence friendly modal logic, **IFML**, that is equally expressive as $\mathbf{IF}(ST(\mathbf{ML}))$, or, to put it otherwise, whose 'standard translation' into IF first-order logic yields $\mathbf{IF}(ST(\mathbf{ML}))$. As will be seen, in the specific IF modal logic we will introduce, the truth condition of the sample formula is captured by the formula $\blacksquare(\blacklozenge p \vee q)$.

Plan of the paper. *Section* 2 introduces the basic notions, and *Section* 3 sets the scene for the specific IF modal logical framework introduced in *Section* 4. In *Section* 5 the expressive power of the resulting *structurally determined* IF modal logic ($\mathcal{L}_{\mathbf{SD}}$) is discussed, and it is shown *not* to lie within the guarded fragment of **FO**. In *Sections* 6, 7 and 8 the model checking, validity and satisfiability problems of $\mathcal{L}_{\mathbf{SD}}$ are each shown to be decidable. The first turns out to be **NP**-complete, and the latter two **PSPACE**-complete. *Section* 9 closes the paper by discussing the conceptual interest of the results reached.

Note on notation. If L is a logic for which syntax and semantics is defined, and ϕ is a formula of L, we write $\phi \in L$ to say that ϕ is among the formulas of L. That is, when no confusion may arise, we do not notationally distinguish a logic from its set of formulas. By L-formula we mean formula of L.

2 Preliminaries

We begin by introducing various notions that will be needed subsequently.

IF-procedure. A mechanical procedure of *IF-ing* first-order logic was introduced by Hintikka [11]. (For a mathematical exposition of IF first-order logic, see [21].) The procedure, applied to an **FO**-formula ϕ in negation normal form, yields a string $\mathbf{IF}(\phi)$ and goes as follows. If in a string produced by the procedure an existential quantifier (disjunction symbol) is followed by a slash sign (/), it is called 'slashed', otherwise 'non-slashed'. If $\phi \in \mathbf{FO}$, we will write '$\prec$' for the relation of *syntactic superordination* among tokens of logical operators in ϕ: if θ is a proper subformula of ψ (which itself is a subformula of ϕ), the outmost form of ψ is $o \in \{\exists, \forall, \vee, \wedge\}$ and the outmost form of θ is $o' \in \{\exists, \forall, \vee, \wedge\}$, then we write $o \prec o'$. As we proceed, we extend the notation to formulas of **IF** in the obvious way.

- Put $\psi = \phi$.
- For every non-slashed existential quantifier $\exists x$ (disjunction sign $\vee$) in ψ, execute the following clause:
  - If $\exists x\, \theta(x)$ is a subformula of ψ occurring subordinate to a number of universal quantifiers which include $\forall y_1, \ldots, \forall y_n$, then it may be replaced by $(\exists x/\{y_1, \ldots, y_n\})\, \theta(x)$, provided that all variables $x, y_1, \ldots, y_n$ are pairwise distinct, and for no i, $\forall y_i \prec \exists y_i \prec \exists x$.
  - If $(\theta \vee \theta')$ is a subformula of ψ occurring subordinate to a number of universal quantifiers which include $\forall y_1, \ldots, \forall y_n$, then it may be replaced by $(\theta\ (\vee/\{y_1, \ldots, y_n\})\ \theta')$, provided that all variables $y_1, \ldots, y_n$ are pairwise distinct, and for no i, $\forall y_i \prec \exists y_i \prec \vee$.
  - Let ψ be the result of the above substitution.
- Return $\mathbf{IF}(\phi) = \psi$.

The set of formulas of $\mathbf{IF}(\tau)$, or *IF first-order logic* (of vocabulary τ), are the strings generated by the above procedure from $\mathbf{FO}(\tau)$-formulas. In any formula $\psi \in \mathbf{IF}$, we allow writing $\exists x$ and $\vee$ for eventual occurrences of $(\exists x/\varnothing)$ and $(\vee/\varnothing)$, respectively.

Skolemization. Hintikka considers **IF**-formulas in which no quantifier Qx_i with a given variable x_i has more than one occurrence. He formulates their semantics in terms of semantic games, and points out that doing so yields an explicit translation of IF first-order logic into existential second-order logic [11, p. 31]. The translation utilizes a notion of *Skolemization*: $\mathcal{M} \models \phi$ iff $\mathcal{M} \models \mathit{Sk}(\phi)$, the Skolemization $\mathit{Sk}(\phi)$ of ϕ being a Σ^1_1-formula. We phrase the idea of Skolemization slightly differently, and will obtain a semantics for the whole language **IF**. It should be noted that we do not thereby deviate from the semantic mechanism Hintikka has in mind for representing logical (in)dependencies: we could reformulate our definition in game-theoretical terms, and most importantly, we continue to operate with 'choice functions' to represent logical (in)dependencies. Our definition is chosen to serve our present purposes, which are entirely connected with the metalogical properties of the logics considered. We will write $ar(f)$ for the arity of a second-order function variable f. Below, if f, g are such variables, and χ involves occurrences of f, 'replacing f by g in χ' means substituting g for *every* occurrence of f. E.g., if $\chi := (R(f, x) \wedge (\exists f/X)\ S(x, f(y, z)))$, the result of the relevant replacement is $(R(g, x) \wedge (\exists g/X)\ S(x, g(y, z)))$. If $\phi \in \mathbf{IF}$, let the *stepwise Skolemization of* ϕ, denoted $\mathit{SSk}(\phi)$, be obtained by applying the following procedure:

- Put $\psi = \phi$.
- Replace every slashed first-order existential quantifier $(\exists x_i/X)$ by the expression $(\exists f_i/X)$, where f_i is a nullary function variable, and replace all variables x_i bound by $(\exists x_i/X)$ in ψ by the nullary function term f_i. Let ψ be the result of this substitution.

- As long as there exists an occurrence of $(\exists f/X)$ in ψ, with $X \neq \varnothing$, execute the following routine:
  - Pick a subformula $(\exists f/X)\ \theta[f(\mathbf{u})]$ of ψ, such that $X \neq \varnothing$.
  - If $\circ \in \{\wedge, \vee\}$ and $((\exists f/X)\theta[f(\mathbf{u})] \circ \theta')$ is a subformula of ψ, introduce a fresh function variable g with $ar(g) = ar(f)$ not appearing anywhere in ψ, replace f by g in $\theta[f(\mathbf{u})]$ to obtain $\theta[g(\mathbf{u})]$, and finally replace $((\exists f/X)\theta[f(\mathbf{u})] \circ \theta')$ by $(\exists g/X)(\theta[g(\mathbf{u})] \circ \theta')$.[1] Proceed similarly if the subformula is $(\theta' \circ (\exists f/X)\theta[f(\mathbf{u})])$.
  - If $(\exists g/\varnothing)(\exists f/X)\ \theta[g(\mathbf{u}), f(\mathbf{v})]$ is a subformula of ψ, then replace it by $(\exists f/X)(\exists g/\varnothing)\ \theta[g(\mathbf{u}), f(\mathbf{v})]$.
  - If $(\forall x)(\exists f/X)\ \theta[x, f(\mathbf{u})]$ is a subformula of ψ and $x \in X$, then replace it by $(\exists f/X - \{x\})(\forall x)\ \theta[x, f(\mathbf{u})]$.
  - If $(\forall x)(\exists f/X)\ \theta[x, f(\mathbf{u})]$ is a subformula of ψ and $x \notin X$, introduce a fresh function variable g with $ar(g) = ar(f) + 1$, and replace $(\forall x)(\exists f/X)\ \theta[x, f(\mathbf{u})]$ by $(\exists g/X)(\forall x)\ \theta[x, g(\mathbf{u}, x)]$, where $\theta[x, g(\mathbf{u}, x)]$ results from replacing f by g in $\theta[x, f(\mathbf{u})]$ and filling in the remaining argument place of g by x.
  - Let ψ be the result of the above substitution.
- Replace all expressions $(\exists f_i/\varnothing)$ by the second-order function quantifier $\exists f_i$, and return $SSk(\phi) = \psi$.

The string $SSk(\phi)$ returned by the procedure is, syntactically, a formula of *second-order logic*. (Quantification over nullary functions can be construed as quantification over individuals.) Its only second-order quantifiers are positive occurrences of the function quantifier $\exists f$, so $SSk(\phi)$ is in fact equivalent to a Σ^1_1-formula. For second-order logic, we apply its standard interpretation [10]: set variables range over arbitrary subsets (of appropriate arity) of the domain. The operation $SSk(\cdot)$ is defined for every $\phi \in \mathbf{IF}$: the above clauses allow proceeding until all sets X in expressions $(\exists f/X)$ are empty. Further, no matter in which order the non-empty sets of variables are resolved in ϕ, the resulting formulas $SSk(\phi)$ are logically equivalent.

EXAMPLE 1. If $\phi := \forall x_1(\exists x_2/\varnothing)\forall x_3(\exists x_4/\{x_1\})\ R(x_1, x_2, x_3, x_4)$, then the first step of the above process returns the formula $\psi := \forall x_1(\exists f_2/\varnothing)\forall x_3(\exists f_4/\{x_1\})\ R(x_1, f_2, x_3, f_4)$. In this case the run of the second step happens to be uniquely determined:

1. $\forall x_1(\exists f_2/\varnothing)(\exists f_4/\{x_1\})\forall x_3\ R(x_1, f_2, x_3, f_4(x_3))$
2. $\forall x_1(\exists f_4/\{x_1\})(\exists f_2/\varnothing)\forall x_3\ R(x_1, f_2, x_3, f_4(x_3))$
3. $(\exists f_4/\varnothing)\forall x_1(\exists f_2/\varnothing)\forall x_3\ R(x_1, f_2, x_3, f_4(x_3))$.

The returned formula $SSk(\phi)$ then is $\exists f_4\forall x_1\exists f_2\forall x_3\ R(x_1, f_2, x_3, f_4(x_3))$. Another example: if we take $\phi := \forall x_1(\exists x_2/\varnothing)\forall x_3(R(x_1, x_2) \wedge \forall x_1(\exists x_2/\{x_1\})\ Q(x_1, x_2, x_3))$, then the first step yields $\psi := \forall x_1(\exists f_2/\varnothing)\forall x_3(R(x_1, f_2) \wedge \forall x_1(\exists f_2/\{x_1\})\ Q(x_1, f_2, x_3))$, and the second $\forall x_1(\exists f_2/\varnothing)\forall x_3(R(x_1, f_2) \wedge (\exists f_2/\varnothing)\forall x_1\ Q(x_1, f_2, x_3))$, whereby $SSk(\phi)$ is returned as $\forall x_1\exists f_2\forall x_3(R(x_1, f_2) \wedge \exists f_2\forall x_1\ Q(x_1, f_2, x_3))$. ■

[1] Observe that by construction g does not appear in θ'.

Neither of the two second-order formulas $SSk(\phi)$ of the above example is, syntactically, in Σ_1^1, but each is obviously equivalent to a Σ_1^1-formula.

DEFINITION 2. Let $\mathcal{M}$ be a τ-structure, and γ a variable assignment. For every $\phi \in \mathbf{IF}(\tau)$, we stipulate that $\mathcal{M}, \gamma \models \phi$, if $\mathcal{M}, \gamma \models SSk(\phi)$. ■

Finite variable fragments. The fragment of first-order logic (of vocabulary τ) consisting of formulas that employ only variables x and y is called its *2-variable fragment*, and denoted $\mathbf{FO^2}(\tau)$. So if $\phi \in \mathbf{FO^2}$, all occurrences of variables in ϕ (both free and bound) are occurrences of x or y. Similarly, we define 2-variable fragment of $\mathbf{IF}(\tau)$, or $\mathbf{IF^2}(\tau)$, to be the fragment of $\mathbf{IF}(\tau)$ generated using the variables x and y only. For $n \geq 3$, we define the n-variable fragment of $\mathbf{FO}^n(\tau)$ *resp.* $\mathbf{IF}^n(\tau)$ to be the fragment of $\mathbf{FO}(\tau)$ *resp.* $\mathbf{IF}(\tau)$ whose formulas are built up using variables $x_1, \ldots, x_n$ only.

$\mathbf{FO^2}$ has the finite model property [15], and so its satisfiability problem is decidable; in fact $\mathbf{FO^2}$-satisfiability is **NEXPTIME**-complete. $\mathbf{FO^2}$ does *not* have the tree model property (not every satisfiable formula has a tree-like model), and it is *not* 'robustly decidable', i.e. has natural extensions which are not decidable. (For $\mathbf{FO^2}$, see, e.g., [9, 22].) By contrast, **ML** is decidable in **PSPACE**, has the tree model property, and is robustly decidable.

Guarded fragment. The guarded fragment of **FO** and the notion of guarded bisimulation were defined in [1]; we follow the formulations of [12]. For **FO**-formulas, we apply the usual notion of 'free variable', and write $Free(\phi)$ for the set of free variables of ϕ. Let Var be a countable set of variables. *Guarded fragment* of vocabulary τ, $\mathbf{GF}(\tau)$, is defined as follows: (A) *Atomic formulas:* (1) If $x_1, x_2 \in Var$, then $x_1 = x_2$ is a formula. (2) If $R \in \tau$ is n-ary and $x_1, \ldots, x_n \in Var$, then $Rx_1 \ldots x_n$ is a formula. (B) *Complex formulas:* (1) If φ, ψ are formulas, then so are $(\varphi \wedge \psi)$ and $\neg\varphi$. (2) Let $\bar{x}$ be a finite non-empty sequence of variables, ψ a formula, and G an atomic formula such that $Free(\psi) \subseteq Free(G)$. Then $\exists\bar{x}(G \wedge \psi)$ is a formula. The atomic formula G in $(B.2)$ is termed the 'guard' of the quantifier $\exists\bar{x}$. The semantics of $\mathbf{GF}(\tau)$ is immediate from that of $\mathbf{FO}(\tau)$, once it is stipulated that $\exists\bar{x}\phi$ with $\bar{x} = x_1 \ldots x_n$ has the same semantics as the first-order formula $\exists x_1 \ldots \exists x_n \phi$.

If $R \in \tau$ and $\mathcal{M}$ is a τ-structure, we extend the usual notation $R^{\mathcal{M}}$, denoting the interpretation of R in $\mathcal{M}$, for sets. Write $X \in R^{\mathcal{M}}$, if the elements of $X \subseteq dom(\mathcal{M})$ are R-related, in any order or multiplicity, i.e., if $(x_1, \ldots, x_n) \in R^{\mathcal{M}}$ and $X = \{x_1, \ldots, x_n\}$. E.g., if R is 6-ary and $(3, 1, 12, 3, 1, 1) \in R^{\mathcal{M}}$, then $\{1, 3, 12\} \in R^{\mathcal{M}}$. If $R \in \tau$, a non-empty finite set $Z \subseteq dom(\mathcal{M})$ is *R-live* in $\mathcal{M}$, if either it is a singleton, or there is a set X such that $Z \subseteq X \in R^{\mathcal{M}}$. Z is *τ-live* in $\mathcal{M}$ (denoted $Z \subseteq^l_\tau \mathcal{M}$), if there is $R \in \tau$ such that Z is *R-live* in $\mathcal{M}$. Hence if τ contains, say, only binary relation symbols, a τ-live set is at most of size 2. A finite *partial τ-isomorphism* between τ-structures $\mathcal{M}$ and $\mathcal{N}$ is a finite one-one partial map $f : M \rightharpoonup N$ such that for every n-ary $R \in \tau$ and all $a_1, \ldots, a_n \in$

$dom(f)$: $\langle a_1, \ldots, a_n \rangle \in R^{\mathcal{M}}$ iff $\langle f(a_1), \ldots, f(a_n) \rangle \in R^{\mathcal{N}}$. *Guarded τ-bisimulation* between $\mathcal{M}$ and $\mathcal{N}$ is a non-empty set $\mathcal{F}$ of finite partial τ-isomorphisms between $\mathcal{M}$ and $\mathcal{N}$ such that for every $f : X \to Y \in \mathcal{F}$: (1) *Zig:* For any $Z \subseteq^l_\tau \mathcal{M}$, there is $g \in \mathcal{F}$ with $dom(g) = Z$, such that g and f agree on $X \cap Z$; and (2) *Zag:* For any $W \subseteq^l_\tau \mathcal{N}$, there is $g \in \mathcal{F}$ with $Im(g) = W$, such that g^{-1} and f^{-1} agree on $Y \cap W$. If $\varphi \in \mathbf{FO}(\tau)$ and $Free(\varphi) \subseteq \{x_1, \ldots, x_k\}$, φ is *invariant for guarded bisimulations*, if for all τ-structures $\mathcal{M}$, $\mathcal{N}$, guarded τ-bisimulations $\mathcal{F}$ between $\mathcal{M}$ and $\mathcal{N}$, maps $f \in \mathcal{F}$, and $m_1, \ldots, m_k \in dom(f)$: $\langle \mathcal{M}, m_1, \ldots, m_k \rangle \models \varphi$ iff $\langle \mathcal{N}, f(m_1), \ldots, f(m_k) \rangle \models \varphi$. Now $\mathbf{GF}(\tau)$-formulas are in fact invariant for guarded τ-bisimulations. In [1] it was proven:

PROPOSITION 3. *Let $\varphi \in \mathbf{FO}(\tau)$ be arbitrary. Then φ is invariant for guarded τ-bisimulations if, and only if, φ is logically equivalent to a formula of $\mathbf{GF}(\tau)$.* ■

GF has the finite model property, and so its satisfiability problem is decidable [1]. In fact, **GF**-satisfiability is complete for deterministic double exponential time [8]. **GF** also has a certain tree model property. Adding least and greatest fixed points to **GF** yields a decidable extension, so arguably **GF** is even robustly decidable. (For a discussion, see [2, 12].)

Expressive power. If L, L' are modal logics whose semantics are defined over the same class of modal structures, we say L is *translatable into* L' (in symbols $L \leq L'$), if for every $\phi \in L$, there is $\psi_\phi \in L'$ such that for all modal structures $\mathfrak{M}$ and all $w \in dom(\mathfrak{M})$: $\mathfrak{M}, w \models \phi$ iff $\mathfrak{M}, w \models \psi_\phi$. L' is *more expressive than* L (denoted $L < L'$), if $L \leq L'$ but $L' \not\leq L$. These notions extend to a comparison between a modal logic and an abstract logic, and between two abstract logics. (Here the 'abstract logics' considered are **FO** and **IF**.)

3 Three choices

Let us specify a fragment of IF first-order logic by fixing, as follows, the three parameters mentioned in the beginning of the paper:

Modal logic. We restrict ourselves to basic modal logic whose formulas are in *negation normal form.* Formally, let **ML** be the language generated by the following grammar: $\phi ::= p \mid \neg p \mid (\phi \vee \phi) \mid (\phi \wedge \phi) \mid \Diamond\phi \mid \Box\phi$, where p is a propositional atom from some countable set Φ of atoms.

Standard translation. We restrict ourselves to the standard translation that uses two first-order variables: x, y. Let $u, v \in \{x, y\}$, with $u \neq v$. For every $\phi \in \mathbf{ML}$, let $ST_2[\phi]$ be defined as $ST_x[\phi]$, following [3]:

$$\begin{aligned}
ST_u[p] &= Pu \\
ST_u[\neg p] &= \neg Pu \\
ST_u[(\phi \circ \psi)] &= (ST_u[\phi] \circ ST_u[\psi]) \qquad \text{if } \circ \in \{\wedge, \vee\} \\
ST_u[\Diamond\phi] &= \exists v(Ruv \wedge ST_v[\phi]) \\
ST_u[\Box\phi] &= \forall v(Ruv \to ST_v[\phi]).
\end{aligned}$$

Let $ST_2[\mathbf{ML}] := \{ST_2[\psi] : \psi \in \mathbf{ML}\}$. Clearly, $ST_2[\mathbf{ML}]$ falls within $\mathbf{FO^2}$.

IF-procedure. We adhere to the aforementioned IF-procedure, yet restricting ourselves to introducing only *slashed existential quantifiers*, thereby ignoring the possibility of independent disjunctions. So essentially we use the procedure $\mathbf{IF}(\cdot)$ that lacks the rule for introducing slashed disjunctions. A quick inspection of the IF-procedure tells us that no new variables are introduced along the way. Thus, if $\phi \in \mathbf{FO}$ sits in the two variable fragment, then also $\mathbf{IF}(\phi)$ uses only two variables. Generally, every formula in $\mathbf{IF}(ST_2[\mathbf{ML}])$ only employs two variables: x, y. That is, $\mathbf{IF}(ST_2[\mathbf{ML}])$ is contained in $\mathbf{IF^2}$.

This particular setting of the parameters was not chosen randomly. The ensuing sections will show that this combination has nice, even surprising features. Observe that the IF-procedure can be applied to $ST_2[\phi]$, with arbitrary $\phi \in \mathbf{ML}$, since ϕ was assumed to be in negation normal form, whereby also $ST_2[\phi]$ is in negation normal form. Further, since ST_2 introduces only two variables, we have that if $(\exists u/X)\ \theta(x)$ is a subformula of $\mathbf{IF}(ST_2[\phi])$, then X is either empty or the singleton $\{v\}$, where $u, v \in \{x, y\}$ and $u \neq v$.

LEMMA 4. $\mathbf{IF}(ST_2[\mathbf{ML}])$ *is translatable into* $\mathbf{FO}$.

Proof. Let $\psi \in \mathbf{IF}(ST_2[\mathbf{ML}])$ be arbitrary. So for some $\phi \in \mathbf{ML}$, there is an execution of the IF-procedure yielding ψ when applied to $ST_2[\phi]$. Because $ST_2[\phi] \in \mathbf{FO^2}$, any slashed existential quantifier $(\exists u/\{v\})$ in ψ satisfies: (i) $u, v \in \{x, y\}$ with $u \neq v$; (ii) v is a universally quantified variable; (iii) there is no existential quantifier $\exists v$ such that $\forall v \prec \exists v \prec \exists u$. Consider, then, the Σ^1_1-formula $SSk(\psi)$. By the algorithm producing $SSk(\psi)$, first each occurrence of $(\exists u/\{v\})$ is replaced in ψ by $(\exists f_u/\{v\})$, and the occurrences of u bound by $(\exists u/\{v\})$ are replaced by f_u. Concentrating on one occurrence of $(\exists f_u/\{v\})$ at a time, the procedure keeps replacing the corresponding occurrences of f_u by fresh function variables g_i, until $(\exists g_i/\{v\})$ is brought to a position subordinate to $\forall v$, but superordinate to all conjunctions/disjunctions to which $(\exists f_u/\{v\})$ is subordinate in ψ. Then $(\exists g_i/\{v\})$ is flipped to the left of $\forall v$, simultaneously removing v from the set $\{v\}$, and the result is that $(\exists g_i/\varnothing)$ immediately precedes $\forall v$. Effecting the transformation for all $(\exists f_u/\{v\})$, and closing the process, results in a formula $SSk(\psi)$, whose all existential quantifiers are function quantifiers $\exists g_i$, with nullary g_i. But this means that $SSk(\psi)$ is a notational variant of a first-order formula. ■

4 Structurally determined IF modal logic

We introduce a modal-like logic, $\mathcal{L}_{\mathbf{SD}}$, which will be seen to characterize the logic $\mathbf{IF}(ST_2[\mathbf{ML}])$, structurally determined above.

Syntax. A class of formulas is generated by the two grammars A and B:

$$\begin{array}{lcl} \alpha & ::= & p \mid \neg p \mid (\alpha \vee \alpha) \mid (\alpha \wedge \alpha) \mid \Diamond\alpha \mid \Box\alpha \mid \blacksquare\beta \\ \beta & ::= & \blacklozenge\alpha \mid (\alpha \vee \beta) \mid (\beta \vee \alpha) \mid (\beta \vee \beta) \mid (\alpha \wedge \beta) \mid (\beta \wedge \alpha) \mid (\beta \wedge \beta), \end{array}$$

with $p \in \Phi$. The two grammars generate the formulas of a logic we refer to as $\mathcal{L}^+_{\mathbf{SD}}$. The α are said to be *closed*, and the β *open*. If ϕ is a formula, all tokens of $\blacklozenge$ not subordinate to a token of $\blacksquare$ in ϕ are called *free*. If ϕ is open, all its free tokens of $\blacklozenge$ become *bound by* the outmost token of $\blacksquare$ in $\blacksquare\phi$. E.g., $\blacklozenge p$ is open; and $\blacksquare(\blacklozenge p \vee \blacklozenge q)$ is closed, the two tokens of $\blacklozenge$ being bound by the unique token of $\blacksquare$. We define the formulas of $\mathcal{L}_{\mathbf{SD}}$ to be the *closed* formulas of $\mathcal{L}^+_{\mathbf{SD}}$. Grammar B contains both clauses $(\alpha \circ \beta)$ and $(\beta \circ \alpha)$ with $\circ \in \{\wedge, \vee\}$, for we wish that open and closed formulas can be conjoined and disjoined in any order. The symbols $\blacksquare$ and $\blacklozenge$ are termed 'black box' and 'black diamond', and the symbols $\Box$ and $\Diamond$ 'white box' and 'white diamond'. Intuitively, $\blacklozenge$ is the 'independent diamond', and it will by definition be independent precisely of the token of $\blacksquare$ that binds it.

Semantics. For every $\phi \in \mathcal{L}^+_{\mathbf{SD}}$, a relation $\mathfrak{M}, I, \bar{i}, w \models \phi$ is defined, where: $\mathfrak{M} = (M, R, V)$ is a modal structure, with a set M of states, $w \in M$, accessibility relation $R \subseteq M^2$, and valuation of propositional atoms $V : \Phi \longrightarrow Pow(M)$. The component I is a map $\{0,1\}^* \longrightarrow M$, called *token valuation*, and $\bar{i} \in \{0,1\}^*$ is a binary string. Here, 0 and 1 are intuitively the choices *left* resp. *right*, interpreting the connectives $\wedge$ and $\vee$. Note that given a formula ϕ, a binary string $i_1 \dots i_n$ determines a subformula, namely the one yielded by going through, outside-in, the connectives $\wedge$ and $\vee$ of ϕ, and choosing for the j-th connective encountered *left* or *right* according to whether i_j is 0 or 1.

$$
\begin{array}{rll}
\mathfrak{M}, I, \bar{i}, w \models p & \text{iff:} & w \in V(p) \\
\mathfrak{M}, I, \bar{i}, w \models \neg p & \text{iff:} & w \notin V(p) \\
\mathfrak{M}, I, \bar{i}, w \models \Diamond\psi & \text{iff:} & \text{for some } v \text{ with } R(w,v)\text{: } \mathfrak{M}, I, \bar{i}, w \models \psi \\
\mathfrak{M}, I, \bar{i}, w \models \Box\psi & \text{iff:} & \text{for every } v \text{ with } R(w,v)\text{: } \mathfrak{M}, I, \bar{i}, w \models \psi \\
\mathfrak{M}, I, \bar{i}, w \models (\psi \vee \phi) & \text{iff:} & \mathfrak{M}, I, \bar{i}0, w \models \psi \ \text{ or } \ \mathfrak{M}, I, \bar{i}1, w \models \phi \\
\mathfrak{M}, I, \bar{i}, w \models (\psi \wedge \phi) & \text{iff:} & \mathfrak{M}, I, \bar{i}0, w \models \psi \ \text{ and } \ \mathfrak{M}, I, \bar{i}1, w \models \phi \\
\mathfrak{M}, I, \bar{i}, w \models \blacksquare\phi & \text{iff:} & \text{for some } I' : \{0,1\}^* \longrightarrow M\text{: } \mathfrak{M}, I', \bar{i}, w \models \Box\phi \\
\mathfrak{M}, I, \bar{i}, w \models \blacklozenge\phi & \text{iff:} & R(w, I(\bar{i})) \text{ and } \mathfrak{M}, I, \bar{i}, I(\bar{i}) \models \phi.
\end{array}
$$

The clauses for literals and formulas of the forms $\Box$ and $\Diamond$ do not employ the components I and $\bar{i}$. By contrast, these components play a key role in the rest of the semantic clauses. Inspection of the clauses reveals that as it stands, $\blacksquare$ acts as a second-order existential quantifier over token valuations (cf., however, Remark 5a). By contrast, $\blacklozenge$ is not a quantifier at all. Instead, it operates on information already available: turns attention to the state $I(\bar{i})$ obtained by applying I to the string $\bar{i}$, claims that state to be accessible from the current state, and makes evaluation to continue relative to the state $I(\bar{i})$. Why do these clauses permit to render $\blacklozenge$ and $\blacksquare$ their desired meanings, i.e. to make the occurrences of $\blacklozenge$ bound by $\blacksquare$ logically independent of it?

Semantically, the black box has a double role: in addition to having the force of the standard box (white box), it *binds* certain subordinate tokens of the black diamond. To these two sides of the semantics of $\blacksquare$, two evaluation steps of a formula $\blacksquare\phi$ correspond. (1) The first step consists of choosing a token valuation I which fixes, at one stroke, the interpretations of *all* tokens of $\blacklozenge$ bound by the relevant occurrence of $\blacksquare$. Technically this is accomplished by choosing a function from binary strings to the domain: the relevant tokens of the black diamond are identified by such strings. (2) The second step is to go on replacing $\blacksquare$ by $\Box$, and evaluating $\Box\phi$: this adds to the semantics of the black box the force of the usual box, i.e. makes it a universal quantifier over accessible states. When the evaluation eventually reaches one of the tokens of $\blacklozenge$ bound by $\blacksquare$, *nothing* is done to interpret the token; instead, if the token is identified by the string $\bar{i}$, its interpretation has already been fixed to be $I(\bar{i})$. This two-step process guarantees precisely that *each token of the black diamond is logically independent of the black box binding it*. For instance, evaluating $\blacksquare\blacklozenge\top$ relative to $\langle\mathfrak{M}, I, \bar{i}, w\rangle$ involves *first* choosing a state u to interpret $\blacklozenge$, although $\blacklozenge$ is syntactically subordinate to $\blacksquare$. This is done by fixing a token valuation I' such that $I'(\bar{i}) = u$. Only then the universal force of $\blacksquare$ is processed: the evaluation moves to $\Box\blacklozenge\top$ and a state v accessible from w is chosen. Finally, $\blacklozenge\top$ is evaluated by checking whether u can actually be accessed from v. It is useful to observe the following features of the above semantics:

REMARK 5. (**a**) For the token valuations in $\mathcal{L}^{+}_{\mathbf{SD}}$-semantics, all that matters are the *free* occurrences of the black diamond in a formula. It is not difficult to see that if (relative to an initial string $\bar{i}$) the free occurrences of $\blacklozenge$ in ϕ are those identified by the strings in the set $S \subset \{0,1\}^*$ (which is necessarily finite), then if for *some* I, we have $\mathfrak{M}, I, \bar{i}, w \models \phi$, actually $\mathfrak{M}, I', \bar{i}, w \models \phi$ holds for any I' such that for all $\bar{j} \in S$, $I'(\bar{j}) = I(\bar{j})$. In particular, to keep the satisfaction condition intact we need not have $I'(\bar{i}) = I(\bar{i})$ unless $\bar{i} \in S$. It can also be noted that the semantic clause for the black diamond need not be phrased in terms of quantification over token valuations: to evaluate $\blacksquare\phi$, it suffices to choose a fixed finite number of states: as many states as there are free occurrences of $\blacklozenge$ in ϕ to be interpreted. (**b**) Let us write $\mathfrak{M}, w \models \phi$ to express the following condition: $\mathfrak{M}, I, \bar{i}, w \models \phi$ holds for all $I : \{0,1\}^* \longrightarrow M$ and all $\bar{i} \in \{0,1\}^*$. As a special case of (a) we have: If ϕ is closed (i.e., if $\phi \in \mathcal{L}_{\mathbf{SD}}$), and for *some* I and $\bar{i}$, we have $\mathfrak{M}, I, \bar{i}, w \models \phi$, then actually $\mathfrak{M}, w \models \phi$ holds. By (a), in the evaluation of open $\mathcal{L}^{+}_{\mathbf{SD}}$-formulas, free tokens of the black diamond (as identified by certain strings) bear resemblance to free variables, and token valuations to variable assignments: satisfaction is essentially dependent on the values assigned to free variables. And by (b), $\mathcal{L}_{\mathbf{SD}}$-formulas are like *sentences* in **FO**: if satisfied under one assignment, they are satisfied under all assignments.

EXAMPLE 6. Consider evaluating the closed formula $\phi := \blacksquare(\blacklozenge p \vee \blacklozenge q)$ at the root w of the modal structure $\mathfrak{M} = (M, R, V)$ depicted in Figure 1 below. Observe that $R(w, z)$ iff $z \in \mathbb{Z}$, and that $R(z, e)$ if $z = 2z'$ for some $z' \in \mathbb{Z}$, while otherwise $R(z, o)$. Further, note that $e \in V(p) \not\ni o$ and $o \in V(q) \not\ni e$.

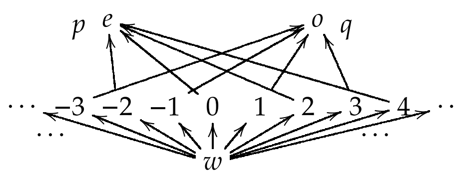

FIGURE 1

We claim $\mathfrak{M}, w \models \phi$. To prove this, it is enough to show that for any token valuation I_0: $\mathfrak{M}, I_0, \emptyset, w \models \phi$. Choose I so that[2] $I(0) = e$ and $I(1) = o$. Fixing a valuation corresponds to picking out, as it were beforehand, the interpretations of the two black diamonds $\blacklozenge$ that can come across later in the evaluation. It suffices to check that $\mathfrak{M}, I, \emptyset, w \models \Box(\blacklozenge p \;\vee\; \blacklozenge q)$. For this to hold, it must be possible to partition the set $\mathbb{Z}$ of R-successors of w into two cells (corresponding to the choice *left* or *right* for $\vee$), so that if z belongs to one of them, we have $\mathfrak{M}, I, 0, z \models \blacklozenge p$, while if z belongs to the other, $\mathfrak{M}, I, 1, z \models \blacklozenge q$ holds. Let the cells be $\{2z : z \in \mathbb{Z}\}$ for *left*, and $\{2z + 1 : z \in \mathbb{Z}\}$ for *right*. Then the former condition holds, because $I(0)$ is accessible from all states in the first mentioned cell, and $I(0) \in V(p)$; and the latter holds because $I(1)$ is accessible from all states in the other cell, and $I(1) \in V(q)$. ■

What kind of logic is $\mathcal{L}_{\mathbf{SD}}$? From one perspective, it is a modal-like characterization of $\mathbf{IF}(ST_2[\mathbf{ML}])$, cf. Proposition 9. $\mathcal{L}_{\mathbf{SD}}$ has compositional semantics, and hence it provides a particularly useful handle on $\mathbf{IF}(ST_2[\mathbf{ML}])$, whose semantics is not compositional. Both $\mathbf{IF}(ST_2[\mathbf{ML}])$ and $ST_2[\mathbf{ML}]$ employ only two variables; both admit of the same partial motivation from the computational viewpoint: seeing what can be expressed in terms of limited memory resources. The former extends the latter in the direction of informational independence just as much as it goes, given the 2-variable restriction and staying with a Hintikka-style IF-procedure. One can also view $\mathcal{L}_{\mathbf{SD}}$ as a *fragment* of the general IF modal logic **IFML** (cf. [19]), obtained by allowing to indicate a diamond as independent *only* from the closest superordinate modal operator, which furthermore must be a box. Indeed, the independence represented by placing $\blacklozenge$ subordinate to $\blacksquare$ is, of course, nothing but the logical independence that can alternatively be marked using the slash notation, indicating a diamond as independent of a superordinate box. Due to the limited independencies expressible in the framework at hand, we can, as a matter of fact, express them all without the slash notation, having instead an 'independent version' $\blacklozenge$ of $\Diamond$ available, 'bound' by the version $\blacksquare$ of $\Box$ on which the appropriate tokens of $\blacklozenge$ will, on the other hand, by definition *not* logically depend. The relevant fragment of **IFML** could certainly be singled out even without the framework of the present paper. However, here it emerges naturally: its IF first-order logic

[2] Observe that the arguments 0 and 1 of I are binary strings of length 1 from $\{0, 1\}^*$, and must not be confused with the elements $0, 1 \in \mathbb{Z}$ of the domain!

correspondence language $\mathbf{IF}(ST_2[\mathbf{ML}])$ is well-motivated on its own; what is more, its compositional semantics makes it particularly well-behaved.

FACT 7. On syntactic grounds, if $\phi \in \mathbf{ML}$, then $\phi \in \mathcal{L}_{\mathbf{SD}}$. And semantically, if $\phi \in \mathbf{ML}$, then ϕ is true in $\mathfrak{M}$ at w under the usual semantics of $\mathbf{ML}$ iff $\mathfrak{M}, w \models \phi$. It follows that $\mathbf{ML} \leq \mathcal{L}_{\mathbf{SD}}$. ■

Some definitions. Each formula $\phi \in \mathcal{L}^+_{\mathbf{SD}}$ is assigned a natural number $md[\phi]$, its *modal depth*, as follows: $md[p] = md[\neg p] = 0$; if $\circ \in \{\wedge, \vee\}$, then $md[(\phi \circ \psi)] = \max\{md[\phi], md[\psi]\}$; and if $O \in \{\Box, \Diamond, \blacksquare, \blacklozenge\}$, then $md[O\phi] = md[\phi] + 1$. Further, define an operation $Sub[\cdot]$ on formulas of $\mathcal{L}^+_{\mathbf{SD}}$: if $p \in \Phi$, $Sub[p] = \{p\}$ and $Sub[\neg p] = \{\neg p\}$; if $\circ \in \{\wedge, \vee\}$, then $Sub[(\varphi \circ \psi)] = \{(\varphi \circ \psi)\} \cup Sub[\varphi] \cup Sub[\psi]$; and if $O \in \{\Box, \Diamond, \blacksquare, \blacklozenge\}$, then $Sub[O\varphi] = \{O\varphi\} \cup Sub[\varphi]$. If $\psi \in Sub[\phi]$, ψ is a *subformula* of ϕ. The *size* of $\phi \in \mathcal{L}^+_{\mathbf{SD}}$, denoted $|\phi|$, is the number of symbol tokens appearing in ϕ, except that propositional atoms are counted as one no matter how many symbols are used for writing them down. E.g. $|(\blacklozenge p_{127} \vee \Box\neg p_1)| = 8$. Whenever $\mathfrak{M}$ is a modal structure, 'M' stands for its domain; and if $w \in M$, the pair $\langle \mathfrak{M}, w\rangle$ is termed a *model*.

Standard translation into IF first-order logic. Basic modal logic can be translated into $\mathbf{FO^2}$. The class of closed $\mathcal{L}^+_{\mathbf{SD}}$-formulas, i.e., the class $\mathcal{L}_{\mathbf{SD}}$, has an analogous property. The following map $ST^{\mathbf{IF}}_x : \mathcal{L}_{\mathbf{SD}} \longrightarrow \mathbf{IF^2}$ provides a canonical translation of $\mathcal{L}_{\mathbf{SD}}$ into $\mathbf{IF^2}$. For all $u, v \in \{x, y\}$ with $u \neq v$, put:

$$\begin{aligned}
ST^{\mathbf{IF}}_u[p] &= Pu\\
ST^{\mathbf{IF}}_u[\neg p] &= \neg Pu\\
ST^{\mathbf{IF}}_u[(\phi \circ \psi)] &= (ST^{\mathbf{IF}}_u[\phi] \circ ST^{\mathbf{IF}}_u[\psi]) \quad \text{ if } \circ \in \{\vee, \wedge\}\\
ST^{\mathbf{IF}}_u[\Diamond\phi] &= \exists v(Ruv \wedge ST^{\mathbf{IF}}_v[\phi])\\
ST^{\mathbf{IF}}_u[\Box\phi] &= \forall v(Ruv \rightarrow ST^{\mathbf{IF}}_v[\phi])\\
ST^{\mathbf{IF}}_u[\blacklozenge\phi] &= (\exists v/u)(Ruv \wedge ST^{\mathbf{IF}}_v[\phi])\\
ST^{\mathbf{IF}}_u[\blacksquare\phi] &= ST^{\mathbf{IF}}_u[\Box\phi].
\end{aligned}$$

Observe how $ST^{\mathbf{IF}}_x$ acts on black boxes and black diamonds: e.g., $ST^{\mathbf{IF}}_x[\blacksquare\blacklozenge p] = ST^{\mathbf{IF}}_x[\Box\blacklozenge p] = \forall y(Rxy \rightarrow ST^{\mathbf{IF}}_y[\blacklozenge p]) = \forall y(Rxy \rightarrow (\exists x/y)(Ryx \wedge ST^{\mathbf{IF}}_x[p])) = \forall y(Rxy \rightarrow (\exists x/y)(Ryx \wedge Px))$. The result is like $ST_2[\Box\Diamond p]$, except that the quantifier $\exists x$ translating $\blacklozenge$ is *independent of* the quantifier $\forall y$ translating $\blacksquare$. Note that the truth condition of $\blacksquare\blacklozenge p$ is actually expressible in $\mathbf{FO^3}$ by $\exists z\forall y(Rxy \rightarrow (Ryz \wedge Pz))$. Clearly if $\phi \in \mathcal{L}^+_{\mathbf{SD}}$ is closed, $ST^{\mathbf{IF}}_x[\phi]$ is an $\mathbf{IF^2}$-formula of exactly one free variable, x. In fact $ST^{\mathbf{IF}}_x$ provides a translation:

PROPOSITION 8. *For all formulas $\phi \in \mathcal{L}_{\mathbf{SD}}$ and all models $\langle \mathfrak{M}, w\rangle$: $\mathfrak{M}, w \models \phi$ iff $\mathfrak{M}^{\mathbf{FO}}, \gamma \models ST^{\mathbf{IF}}_x[\phi]$, where $\mathfrak{M}^{\mathbf{FO}}$ is the first-order structure that corresponds, in the canonical way, to the modal structure $\mathfrak{M}$; and $\gamma(x) = w$.*

Proof. The statement can be proven by induction on the structure of closed formulas. Note that if $\blacksquare\psi$ is closed, ψ is obtained by $\wedge$ and $\vee$ from closed formulas and formulas of the form $\blacklozenge\theta$, where θ is closed. ■

The following 'commutativity' result holds, establishing that $\mathcal{L}_{\mathbf{SD}}$ actually is structurally determined in the desired way. (For a proof, see [19].)

PROPOSITION 9. *Syntactically,* $ST_x^{\mathbf{IF}}[\mathcal{L}_{\mathbf{SD}}] = \mathbf{IF}(ST_x[\mathbf{ML}])$. ■

5 Expressive power

Relation to finite variable fragments of FO. $\mathcal{L}_{\mathbf{SD}}$ is *not* translatable into *any* finite variable fragment of **FO**. This is of course in contrast with the case of **ML**, translatable into $\mathbf{FO^2}$. Still $\mathcal{L}_{\mathbf{SD}}$ *is* translatable into **FO**.

THEOREM 10. (**a**) *For all* $n < \omega$, $\mathcal{L}_{\mathbf{SD}} \not\leq \mathbf{FO}^n$. *However,* (**b**) $\mathcal{L}_{\mathbf{SD}} \leq \mathbf{FO}$.

Proof. (a) Let $n \geq 2$, and think of the formula $\phi_n := \blacksquare(\blacklozenge\top \vee \ldots \vee \blacklozenge\top)$, where the number of disjuncts in $(\blacklozenge\top \vee \ldots \vee \blacklozenge\top)$ is $n-1$. In a modal structure $\mathfrak{M} = (M, R, V)$ at a state w, ϕ_n asserts that the set $\{v : R(w,v)\}$ can be partitioned into (at most) $n-1$ cells in such a way that the elements in each cell have a *common successor* along R. The truth condition of ϕ_n can be expressed by the first-order formula $\phi'_n := \exists z_1 \ldots \exists z_{n-1} \forall y(Rxy \rightarrow (Ryz_1 \vee \ldots \vee Ryz_{n-1}))$. Now $\phi'_n \in \mathbf{FO}^{n+1}$, but it is not difficult to see (by reference to a pebble game argument[3]) that $\phi'_n \notin \mathbf{FO}^n$. Hence the greater the number n is, the more variables are needed to translate the formula ϕ_n into first-order logic. (b) Follows from Propositions 8 and 9 together with Lemma 4. ■

ML is translatable into $\mathbf{FO^2}$, but $\mathcal{L}_{\mathbf{SD}}$ is not. By Fact 7 we conclude:

COROLLARY 11. $\mathbf{ML} < \mathcal{L}_{\mathbf{SD}}$. ■

Corollary 11 could be proven more straightforwardly by adapting the argument of [17, Lemma 4], reproduced in [13, Example 2].

REMARK 12. The results of Theorem 10 can be compared with what is known about other IF modal logics. The 'IF modal logic of perfect recall' studied in [13] can be shown to be translatable into $\mathbf{FO^3}$. On the other hand, giving up the perfect recall assumption, the resulting IF modal logic does *not* admit of *any* translation into **FO**, as shown in [19]. ■

Number of non-equivalent formulas. Restricting attention to a finite set Φ of propositional atoms, it is well known that for any $m < \omega$, the number of pairwise non-equivalent formulas ψ of **ML** with $md[\psi] = m$ is finite; see, e.g., [3, Prop. 2.29]. This property does not extend to $\mathcal{L}_{\mathbf{SD}}$. In fact we have:

PROPOSITION 13. *Let the size of* Φ *be finite. For any* $m \geq 2$*, the set of pairwise non-equivalent* $\mathcal{L}_{\mathbf{SD}}$*-formulas of modal depth* m *is infinite.*

Proof. If $k \geq 0$ and $n \geq 2$, let $\phi_n^k := \Box \ldots \Box\phi_n$, where ϕ_n is defined as in the proof of Theorem 10(a), and the number of boxes preceding ϕ_n is k. Whenever $n < n'$, the formulas ϕ_n^k and $\phi_{n'}^k$ are not equivalent. However,

[3] On how to use pebble games $G_m^n(\mathcal{M}, \mathbf{a}, \mathcal{N}, \mathbf{b})$ to characterize equivalence of structures up to quantifier rank $\leq m$ relative to $\mathbf{FO}^n$, see, e.g., [6, pp. 49-50].

$md[\phi_n^k] = k + 2 = md[\phi_{n'}^k]$. Hence the formulas in the set $\{\phi_n^k : n < \omega\}$ are pairwise non-equivalent, but they all have the same modal depth, $k+2$. ■

Relation to the guarded fragment. According to the influential suggestion of Andréka, van Benthem and Németi [1], the distinguishing feature of the modal fragments of **FO** is that the quantifiers they harbor appear guarded. The nice computational properties of modal fragments would be 'accounted for' by their translatability into **GF**. Our modal-like logic $\mathcal{L}_{\mathbf{SD}}$ fails to share this feature with **ML**.

Let us adopt the following notation. If $\mathcal{F}$ is a set of finite partial τ-isomorphisms between $\mathcal{M}$ and $\mathcal{N}$, $f \in \mathcal{F}$, and X is live in $\mathcal{M}$, write $\mathbf{Zig}(f, X)$ for the claim "there is $g \in \mathcal{F}$ with $dom(g) = X$ such that g and f agree on the intersection of their domains." Similarly, write $\mathbf{Zag}(f, X)$ for the claim "there is $g \in \mathcal{F}$ with $Im(g) = X$ such that g^{-1} and f^{-1} agree on the intersection of their domains." Let us prove that $\mathcal{L}_{\mathbf{SD}}$ cannot be translated into **GF**.

THEOREM 14. *There is no (truth-preserving) translation of* $\mathcal{L}_{\mathbf{SD}}$ *into* **GF**.

Proof. Let R be a binary relation symbol. Consider the $\{R\}$-structures $\mathfrak{M}$ and $\mathfrak{N}$ depicted in Figure 2 (the arrow represents R-accessibility):

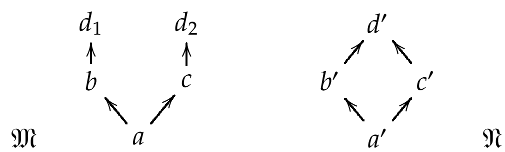

FIGURE 2

Let us define $\mathfrak{F}$ as the set consisting of the maps $\{(a,a')\}$, $\{(b,b')\}$, $\{(c,c')\}$, $\{(d_1,d')\}$, and $\{(d_2,d')\}$, together with the maps $f_1 := \{(a,a'),(b,b')\}$, $f_2 := \{(a,a'),(c,c')\}$, $f_3 := \{(b,b'),(d_1,d')\}$ and $f_4 := \{(c,c'),(d_2,d')\}$. We claim $\mathfrak{F}$ is a guarded $\{R\}$-bisimulation. $\mathfrak{F}$ is non-empty, and evidently all maps in $\mathfrak{F}$ are partial $\{R\}$-isomorphisms between $\mathfrak{M}$ and $\mathfrak{N}$. It must be checked that for all $f \in \mathfrak{F}$ and all live sets Z in $\mathfrak{M}$, $\mathbf{Zig}(f, Z)$; and that for all $f \in \mathfrak{F}$ and all live sets W in $\mathfrak{N}$, $\mathbf{Zag}(f, W)$. The live sets of size 2 in $\mathfrak{M}$ are $\{a,b\}$, $\{a,c\}$, $\{b,d_1\}$, $\{c,d_2\}$; and the live sets of size 2 in $\mathfrak{N}$ are $\{a',b'\}$, $\{a',c'\}$, $\{b',d'\}$, $\{c',d'\}$.

- If X is a singleton, it is trivial that there is a singleton $g \in \mathfrak{F}$ witnessing the claims $\mathbf{Zig}(f, X)$ and $\mathbf{Zag}(f, X)$.
- If f is a singleton and X is of size 2, it is immediate there is a singleton $g \in \mathfrak{F}$ that witnesses $\mathbf{Zig}(f, X)$, and similarly for $\mathbf{Zag}(f, X)$.
- We must still check the claims $\mathbf{Zig}(f, X)$ and $\mathbf{Zag}(f, X)$, assuming f, X both are of size 2. Now $\mathbf{Zig}(f_1, \{b, d_1\})$ is witnessed by f_3, and $\mathbf{Zig}(f_3, \{a, b\})$ by f_1. Similarly, $\mathbf{Zig}(f_2, \{c, d_2\})$ is witnessed by f_4; and $\mathbf{Zig}(f_4, \{a, c\})$ by f_2. Furthermore, $\mathbf{Zag}(f_1, \{b', d'\})$ is witnessed by f_3, and $\mathbf{Zag}(f_3, \{a', b'\})$ by f_1. Similarly, $\mathbf{Zag}(f_2, \{c', d'\})$ is witnessed by f_4; and $\mathbf{Zag}(f_4, \{a', c'\})$ by f_2.

So $\mathfrak{F}$ is a guarded $\{R\}$-bisimulation between $\mathfrak{M}$ and $\mathfrak{N}$; and $\langle\mathfrak{M}, a\rangle$ and $\langle\mathfrak{N}, a'\rangle$ satisfy precisely the same $\mathbf{GF}(\{R\})$-formulas of one free variable. Observe, then, that the **FO**-formula $\exists z\forall y(Rxy \to Ryz)$ is not satisfied in $\langle\mathfrak{M}, a\rangle$, but is satisfied in $\langle\mathfrak{N}, a'\rangle$. But $\exists z\forall y(Rxy \to Ryz)$ is a first-order translation of the $\mathcal{L}_{\mathbf{SD}}$-formula $\blacksquare\blacklozenge\top$. It follows that $\mathcal{L}_{\mathbf{SD}}$ is not translatable into **GF**. ■

Unorthodox properties of the modal-like logic $\mathcal{L}_{\mathbf{SD}}$, such as those witnessed by Proposition 13 and Theorems 10 and 14, might suggest it should be of a rather marginal interest as a modal logic; even its status as a modal logic might be thereby called into question. Now in *Sections* 7 and 8 it is shown that the validity and satisfiability problems of $\mathcal{L}_{\mathbf{SD}}$ are not only decidable, but decidable in **PSPACE**. Insofar as $\mathcal{L}_{\mathbf{SD}}$ is a modal logic, then, it is a modal logic translatable into **FO** *and* decidable in **PSPACE**. But it does *not* lie within the guarded fragment: there is no translation $\mathcal{L}_{\mathbf{SD}} \longrightarrow \mathbf{GF}$. To the extent it is a modal logic, it tells against the hypothesis that embeddability to **GF** is what 'explains' the nice computational properties of modal logics.

6 Model checking

We proceed to discuss the computational properties of $\mathcal{L}_{\mathbf{SD}}$; let us begin with the model checking problem.

THEOREM 15. *Model checking* $\mathcal{L}_{\mathbf{SD}}$ *is* **NP**-*complete.*

Proof. *Membership.* In the semantics we took the component I of the models to be a function of type $\{0,1\}^* \longrightarrow M$, but as noted in Remark 5, this is needlessly general. For simplicity and w.l.o.g., let us restrict attention to functions of type $\{1,\ldots,n\} \longrightarrow M$, where n is the number of occurrences of the black diamond in ϕ. Labeling those occurrences each with its own number $j \in \{1,\ldots,n\}$, we may suppress the component $\bar{i}$ from the models (the labels will do the job of these strings). Then, instead of evaluating $\blacklozenge\psi$ relative to $\langle\mathfrak{M}, I^*, \bar{i}, w\rangle$, we evaluate $\blacklozenge_j\psi$ relative to $\langle\mathfrak{M}, I, w\rangle$, provided that $I(j) = I^*(\bar{i})$. Let, then, $\phi \in \mathcal{L}_{\mathbf{SD}}$ and $\mathfrak{M}, I, w$ be given, the domain of $\mathfrak{M}$ being finite. We describe an algorithm deciding whether $\mathfrak{M}, I, w \models \phi$, that is, $\mathcal{L}_{\mathbf{SD}}$'s model checking algorithm.

- For every pair $\langle v, \blacksquare\psi\rangle$, where $v \in M$ and $\blacksquare\psi$ is a subformula of ϕ, non-deterministically guess a function $I_{\langle v,\blacksquare\psi\rangle} : \{1,\ldots,n\} \longrightarrow M$.
- Put $k = 1$, and for every propositional atom p occurring in ϕ, and for every state $v \in M$, label v with p if $p \in V(v)$, and with $\neg p$ otherwise.
- While $k \leq md[\phi]$, proceed as follows:
  - Repeat the routine for every $v \in M$ and every subformula $\blacksquare\psi$ of ϕ such that $md[\blacksquare\psi] = k$:
    Essentially, we must decide whether $\mathfrak{M}, I, v \models \blacksquare\psi$. To do so, it suffices to check whether $\mathfrak{M}, I_{\langle v,\blacksquare\psi\rangle}, v \models \Box\psi$, because if there exists a function I' that does the job, then so does the non-deterministically guessed function $I_{\langle v,\blacksquare\psi\rangle}$. Deciding whether

$\mathfrak{M}, I_{\langle v, \blacksquare\psi\rangle}, v \models \Box\psi$ now proceeds along the same lines as model checking basic modal logic. None of ψ's subformulas of the form $\blacksquare\chi$ causes any further problem, since $md[\blacksquare\chi] < k$ and has therefore already been taken care of in an earlier iteration of the algorithm. That is, every state $u \in M$ has been labeled with $\blacksquare\chi$ if, and only if, $\mathfrak{M}, I, u \models \blacksquare\chi$. If it turns out that $\mathfrak{M}, I_{\langle v, \blacksquare\psi\rangle}, v \models \Box\psi$, then label v with $\blacksquare\psi$, otherwise do not.

  – Increase k by one.

- If w is labeled with ϕ, accept the input, otherwise reject it.

It is not hard to see that this algorithm is correct; i.e., it accepts $\langle\phi, \mathfrak{M}, I, w\rangle$ iff $\mathfrak{M}, I, w \models \phi$. Furthermore, the algorithm runs in polynomial non-deterministic time. Namely, first, the guessing of one function $I_{\langle v, \blacksquare\psi\rangle}$ can be done in time linear in the number of black diamonds in ϕ, which is clearly bounded by ϕ's size. The non-deterministic guessing of $I_{\langle v, \blacksquare\psi\rangle}$ for every $v \in M$ and every subformula of the form $\blacksquare\psi$ takes therefore $|\phi| \cdot |M| \cdot md[\phi]$ steps. The latter term is polynomial in the size of the input, since $md[\phi] \leq |\phi|$. For the rest, the algorithm hinges on the model checking algorithm for basic modal logic that runs in polynomial deterministic time.

Hardness. We reduce from Set covering, which takes a family $F = \{S_1, \ldots, S_n\}$ of subsets of a finite set U and a budget b as instance. The problem is to decide whether there are b sets in F, whose union is U. (See [16].) Define a structure $\mathfrak{M}_F := (M, R, V)$ by putting $M := \{w\} \cup U \cup F$; $R := \{\langle w, u_i\rangle : u_i \in U\} \cup \{\langle u_i, S_j\rangle : u_i \in S_j\}$; and $V(p) := F$. Let I be any token valuation. We claim that (F, b) sits in Set covering iff $\mathfrak{M}_F, I, w \models \blacksquare(\blacklozenge_1 p \vee \ldots \vee \blacklozenge_b p)$. Abusing formal notation, the latter holds iff $(\exists T_1, \ldots, T_b \in F)(\forall u_i \in U)(u_i \in T_1 \vee \ldots \vee u_i \in T_b)$, which spells out the conditions under which (F, b) is contained in Set covering. ■

7 Validity problem

In connection with $\mathcal{L}_{\mathbf{SD}}$, we must treat the validity and satisfiability problems separately. To see this, observe first that the syntax does not allow to form the negation of an arbitrary formula $\phi \in \mathcal{L}_{\mathbf{SD}}$: negation sign appears only in front of propositional atoms. Further, $\blacklozenge\psi$ is *not* true relative to $\langle\mathfrak{M}, I, \bar{\imath}, w\rangle$ iff: either $\langle w, I(\bar{\imath})\rangle \notin R$ or $\mathfrak{M}, I, \bar{\imath}, w \not\models \psi$; and $\blacksquare\psi$ is *not* true relative to $\langle\mathfrak{M}, I, \bar{\imath}, w\rangle$ iff: for *all* valuations $I' : \{0,1\}^* \longrightarrow M$, we have $\mathfrak{M}, I', \bar{\imath}, w \not\models \Box\psi$. So $\blacklozenge$ and $\blacksquare$ are far from being duals to each other. Not only does it hence seem unlikely that for every $\phi \in \mathcal{L}_{\mathbf{SD}}$ there is $neg(\phi) \in \mathcal{L}_{\mathbf{SD}}$ such that for all $\mathfrak{M}, I, \bar{\imath}, w$ we have $\mathfrak{M}, I, \bar{\imath}, w \models neg(\phi)$ iff $\mathfrak{M}, I, \bar{\imath}, w \not\models \phi$; but it can be shown that $\mathcal{L}_{\mathbf{SD}}$ is not closed under negation. A rigorous proof would require more space than we have available here: a technique would be to define a kind of extended asymmetric n-bisimulation game $\mathfrak{B}_n(\mathfrak{M}, I, \bar{\imath}, w; \mathfrak{N}, J, \bar{\jmath}, v)$ between *Spoiler* and *Duplicator* such that for all $\phi \in \mathcal{L}^+_{\mathbf{SD}}$ with $md(\phi) \leq n$: if $\mathfrak{M}, I, \bar{\imath}, w \models \phi$ and *Duplicator* has a winning strategy in $\mathfrak{B}_n(\mathfrak{M}, I, \bar{\imath}, w; \mathfrak{N}, J, \bar{\jmath}, v)$, then $\mathfrak{N}, J, \bar{\jmath}, v \models \phi$. Validity and satisfiability problems of $\mathcal{L}_{\mathbf{SD}}$, then, are not each other's duals. The same phenomenon was observed in connection with 'IF modal logic of perfect recall' in [13, Fact 21].

Let us proceed to prove that $\mathcal{L}_{\mathbf{SD}}$-validity is decidable in **PSPACE**. Define a transformation T in $\mathcal{L}^+_{\mathbf{SD}}$ as follows:

$$\begin{aligned} T[\ell] &= \ell && \text{if } \ell \in \{p, \neg p\} \\ T[(\theta \circ \chi)] &= (T[\theta] \circ T[\chi]) && \text{if } \circ \in \{\vee, \wedge\} \\ T[\Box\psi] &= T[\blacksquare\psi] = \Box T[\psi] \\ T[\Diamond\psi] &= \Diamond T[\psi] \\ T[\blacklozenge\psi] &= \bot. \end{aligned}$$

Here are some examples of applying the transformation: $T[\blacksquare(\blacklozenge p \vee q)] = \Box(\bot \vee q)$; $T[\blacksquare\blacklozenge p] = \Box\bot$; and $T[\blacksquare(p \wedge \blacklozenge(q \wedge \Diamond\Box r))] = \Box(p \wedge \bot)$. Note that if in particular $\phi \in \mathcal{L}_{\mathbf{SD}}$ (and hence ϕ is closed), then $T[\phi]$ is, by syntactical criteria, a formula of **ML**. Further, observe that if $\phi \in \mathbf{ML}$, then $T[\phi] = \phi$.

LEMMA 16. *For every $\phi \in \mathcal{L}_{\mathbf{SD}}$, ϕ is valid if, and only if, $T[\phi]$ is valid.*

Proof. Trivially, if $T[\phi]$ is valid, then so is ϕ. For the converse direction, assume that ϕ is valid, but suppose for contradiction that $T[\phi]$ is not. So there is, by the tree model property of **ML**, cf. [3, Prop. 2.15], a tree-like model $\langle\mathfrak{M}, w\rangle$ in which $T[\phi]$ is false, but ϕ is true. In particular, then, there must be a subformula ψ of ϕ of the form $\blacksquare\theta$, and a state $v \in M$, such that ψ is true at v, but $T[\psi]$ is false at v. What is more, among such subformulas ψ of ϕ, we can choose one, $\psi_0 = \blacksquare\theta_0$, that satisfies the following further condition: $\theta_0 = \beta(\chi_1, \ldots, \chi_n, \blacklozenge\nu_1, \ldots, \blacklozenge\nu_m)$ is a boolean combination (in terms of $\wedge, \vee$) of the formulas $\chi_i, \blacklozenge\nu_j$, where all of the χ_i satisfy: for all $u \in M$ at which χ_i is true in $\mathfrak{M}$, also $T[\chi_i]$ is true at u in $\mathfrak{M}$.

Now $T[\psi_0] = \Box(\beta(T[\chi_1], \ldots, T[\chi_n], \bot, \ldots, \bot))$. Replacing conjunction with $\bot$ by $\bot$, and disjunction of a formula with $\bot$ by that formula, the boolean combination β becomes after rewriting either $\beta'(T[\chi_1], \ldots, T[\chi_n])$, or else $\bot$. Let β'' be the $\mathcal{L}_{\mathbf{SD}}$-formula for which $T[\beta''] \in \{\beta', \bot\}$ is the result of the above rewriting. By assumption, then, $\Box T[\beta'']$ is false at v. If in the tree $\mathfrak{M}$ there happens to be less than $m+1$ successors $u_1, \ldots, u_k$ of v at which $T[\beta'']$ is false, let $\langle\mathfrak{N}, u_{k+1}\rangle, \ldots, \langle\mathfrak{N}, u_{m+1}\rangle$ be isomorphic copies of the submodel of $\langle\mathfrak{M}, w\rangle$ generated (say) by u_k, with pairwise disjoint domains, each furthermore disjoint from M. Obtain a tree-like model $\langle\mathfrak{M}', w\rangle$ bisimilar to $\langle\mathfrak{M}, w\rangle$ by adding $u_{k+1}, \ldots, u_{m+1}$ as successors of v, and letting the structures $\langle\mathfrak{N}, u_{k+1}\rangle, \ldots, \langle\mathfrak{N}, u_{m+1}\rangle$ determine an extension of the domain and the accessibility relation of $\langle\mathfrak{M}, w\rangle$.

Thus we have $\mathfrak{M}', v \not\models T[\blacksquare\theta_0]$, and so, by construction, for all $1 \le i \le m+1$: $\mathfrak{M}', u_i \not\models T[\beta'']$. On the other hand, $\mathfrak{M}', v \models \blacksquare\theta_0$, and so there is I such that for all $1 \le i \le m+1$, we have: $\mathfrak{M}', I, \emptyset, u_i \models \theta_0$. It was assumed that for all $u \in M$ at which χ_i is true in $\mathfrak{M}$, also $T[\chi_i]$ is true at u in $\mathfrak{M}$. Hence if we had for some $i \in \{1, \ldots, m+1\}$ that $\mathfrak{M}', I, \emptyset, u_i \models \beta''$, it would follow that $\mathfrak{M}', u_i \models T[\beta'']$, a contradiction. So we must have for all $1 \le i \le m+1$ that $\mathfrak{M}', I, \emptyset, u_i \models \blacklozenge\chi_j$, for some $j \in \{1, \ldots, m\}$. Since there are only m distinct formulas χ_j but $m+1$ states considered, for at least two states the same $\blacklozenge\chi_j$ is chosen. But then, for the corresponding string $\bar{i}$ (i.e. the string of choices left/right leading to $\blacklozenge\chi_j$), the state $I(\bar{i})$ is the common successor of two distinct states u_r and $u_{r'}$. This is impossible, because $\mathfrak{M}'$ is tree-like. ■

THEOREM 17. *The validity problem of $\mathcal{L}_{\mathbf{SD}}$ is* **PSPACE***-complete.*

Proof. By Lemma 16 $\mathcal{L}_{\mathbf{SD}}$-validity is reducible to **ML**-validity in polynomial time. But **ML**-validity is **PSPACE**-complete (see e.g. [3, Ths. 6.47, 6.50]). ■

8 Satisfiability problem

With respect to satisfiability, too, it turns out that $\mathcal{L}_{\mathbf{SD}}$ shares the basic properties with **ML**: it has the strong finite model property, and its satisfiability problem is **PSPACE**-complete. In this section we give a loose description of the algorithm that decides satisfiability for $\mathcal{L}_{\mathbf{SD}}$.

Let S_0 be a finite set of $\mathcal{L}_{\mathbf{SD}}$-formulas. We describe an algorithm $\mathfrak{A}(S_0)$, starting out with the set S_0. First, to make describing the algorithm easier, let us suppose that the occurrences of the black diamond in all formulas $\phi \in S_0$ are labeled, each with a label (positive integer) of its own. Recall that a token of the black diamond in a formula $\psi \in \mathcal{L}^+_{\mathbf{SD}}$ is called 'free', if it is not syntactically subordinate to any black box in ψ. We extend this terminology straightforwardly to labeled tokens of the black diamond: in $\blacklozenge_1\blacksquare\blacklozenge_2\top$, $\blacklozenge_1$ is free but $\blacklozenge_2$ is not. Below, we write $Sub_m(S_0)$ for the set of those closed subformulas of S_0 whose modal depth is at most m.

The distinguishing feature of the algorithm $\mathfrak{A}(S_0)$ will be its treatment of black boxes ($\blacksquare$) and black diamonds ($\blacklozenge_i$). The black box, as in $\blacksquare\phi$, induces the algorithm to non-deterministically guess sets $T_{i_1}, \ldots, T_{i_n} \subseteq Sub_j(S_0)$ and to store them, in case ϕ has n free black diamonds $\blacklozenge_{i_1}, \ldots, \blacklozenge_{i_n}$ and $j := md[\phi] - 2$. Intuitively, the sets $T_{i_1}, \ldots, T_{i_n}$ represent n states at which at least the formulas from the respective sets are true. The black diamond, as in $\blacklozenge_k\psi$, induces the algorithm to check whether the formula ψ is in the set T_k. Since the modal depth of ψ is smaller than the modal depth of the formula $\blacksquare\phi$ whose black box binds $\blacklozenge_k$, choosing subsets from $Sub_j(S_0)$ instead of $Sub(S_0)$ goes without loss of generality. Further, we observe that by the syntax, ψ is closed, so neither does letting $Sub_j(S_0)$ only contain closed formulas affect the functionality of the algorithm.

Ladner's algorithm [14] that decides satisfiability for **ML** does not have non-deterministic choices, nor does it store sets T to perform membership checks. Interestingly, neither extension increases the computational complexity: $\mathcal{L}_{\mathbf{SD}}$-satisfiability is solvable in **PSPACE**, hence **PSPACE**-complete (because by Fact 7, $\mathbf{ML} \le \mathcal{L}_{\mathbf{SD}}$). Our algorithm is conveniently thought of as deciding the satisfiability of a stack $(S, S_1, \ldots, S_K)$ of subsets of S_0. In order to compute whether all sets on the stack are satisfiable, the algorithm considers the leftmost set on the stack, S, and computes new sets $T_1, \ldots, T_m$ that are satisfiable iff S is. Thus, the algorithm transfers

$(S, S_1, \ldots, S_K)$ into $(T_1, \ldots, T_m, S_1, \ldots, S_K)$, and iteratively processes the leftmost set on the stack until a set without modal operators in encountered.

1. When verifying the satisfiability of a set S containing only propositional atoms and their negations, if S is inconsistent (i.e. contains both p and $\neg p$ for some atom $p \in \Phi$), computation terminates and the input is rejected; otherwise S is popped from the stack.
2. When verifying the satisfiability of a set S with $(\chi \wedge \theta) \in S$, remove $(\chi \wedge \theta)$ from S, add both χ and θ to S, and continue checking if the result is satisfiable. When verifying the satisfiability of a set S with $(\chi \vee \theta) \in S$, remove $(\chi \vee \theta)$ from S, non-deterministically guess $\psi \in \{\chi, \theta\}$, put ψ in S, and continue checking if S is satisfiable.
3. If neither of the clauses 1 or 2 applies to S, it can be written as
$$\{\blacksquare\phi_1, \ldots, \blacksquare\phi_m, \Diamond\psi_1, \ldots, \Diamond\psi_n, \Box\chi_1, \ldots, \Box\chi_l\}.$$
For every $\blacksquare\phi_i$, let $\blacklozenge_{\min(i)}\zeta_{\min(i)}, \ldots, \blacklozenge_{\max(i)}\zeta_{\max(i)}$ be the subformulas of $\blacksquare\phi_i$ whose black diamonds are bound by $\blacksquare$. For every such $\blacklozenge_j\zeta_j$, guess a set $T_{i,j} \subseteq Sub_{k-2}(S_0)$ where $k = md[\blacksquare\phi_i]$. Further, for every $\Diamond\psi_t \in S$, let $U'_t = \{\psi_t\} \cup \{\phi_1, \ldots, \phi_m, \chi_1, \ldots, \chi_l\}$. Apply clause 2 to U'_t so that each of its formulas has a modal operator as its outmost form, and let the result be denoted by U''_t. Verify whether for each $\blacklozenge_r\zeta_r \in U''_t$, the formula ζ_r is contained in $T_{i,r}$, given that ϕ_i is the formula that has $\blacklozenge_r\zeta_r$ as a subformula. If not, computation terminates and the input is rejected. Otherwise let U_t be the result of removing all formulas $\blacklozenge_r\zeta_r$ from U''_t. Continue computing the stack $(U_1, \ldots, U_n, T_{1,\min(1)}, \ldots, T_{m,\max(m)}, S_1, \ldots, S_K)$.
4. If the stack is empty, then the input is accepted.

In order to see that the algorithm runs in **PSPACE**, write N for the size of the description of S_0: $N = \Sigma_{\phi \in S_0}|\phi|$. It is immediate that all sets pushed on the stack are of size less than N. We proceed to show that the number of sets on the stack is polynomially bounded by N. Given that the stack is $(S, S_1, \ldots, S_K)$ and one of the clauses 1 or 2 applies to S, the number of sets on the stack after executing the respective clause does not grow. More interesting is the case in which clause 3 applies to S. Then the sets $U_1, \ldots, U_n$ are pushed on the stack together with $T_{1,\min(1)}, \ldots, T_{m,\max(m)}$. There are n sets of the first kind: as many as there are formulas of the form $\Diamond\psi_i$ in S. Clearly, the number of those formulas is bounded above by N. Further, there are at most N^2 sets $T_{i,j}$, since every formula $\blacksquare\phi_i$ binds at most n formulas of the form $\blacklozenge_j\zeta_j$. The upper bound of N^2 follows from the fact that $m < N$.

Hence clause 3 introduces at most $2N^2$ new sets. Let $md[S] := \max\{md[\phi] : \phi \in S\}$. It is important to note that $md[S] > md[U_t]$ and that $md[S] > md[T_{i,j}]$, for all appropriate i, j and t. Thus extending the stack may go on for at most $h := md[S_0]$ cycles without popping sets from the stack. In the worst case we have h times pushed $2N^2$ sets on the stack. Since $h < N$, the stack will contain at most $2N^3$ sets. Thus an upper bound on the number of sets that are stored on the stack is $2N^3$, which is polynomial in the size of the input: N. Since **PSPACE** = **NPSPACE**, the non-deterministic choices do not increase the complexity.

In a canonical way the algorithm produces on input S_0 a model $\langle \mathfrak{M}, w \rangle = \langle M, R, V, w \rangle$ whose domain consists exactly of those sets that were introduced along the way, with $w := S_0$. The relation R is defined so as to satisfy: if the set U_r is induced by the formula $\Diamond \psi_r \in S$, then $R(S, U_r)$. Furthermore, if $\blacklozenge_j \zeta_j \in U_r$ and $\zeta_j \in T_{i,j}$, where ζ_j is a subformula of ϕ_i, then $R(U_r, T_{i,j})$. Finally, $\mathfrak{M}$'s valuation is defined to be the function V such that $S \in V(p)$ iff $p \in S$. Because distinct U_r and $U_{r'}$ can contain $\blacklozenge_j \zeta_j$ for one and the same index j, $\mathfrak{M}$ is not in general tree-like.

PROPOSITION 18. *Let $\phi \in \mathcal{L}_{\mathbf{SD}}$. ϕ is satisfiable iff $\mathfrak{A}(\{\phi\})$ accepts $\{\phi\}$.*

Proof. *Right to left.* $\mathfrak{A}(\{\phi\})$ is so constructed that it provides witnesses of all operators with existential force ($\blacksquare, \Diamond, \vee$) appearing in ϕ, when ϕ is evaluated on the model canonically produced by $\mathfrak{A}(\{\phi\})$. *Left to right.* Any model of ϕ can be used to define an accepting run of $\mathfrak{A}(\{\phi\})$; the fact that we can non-deterministically guess suitable sets in connection with clauses 2 ($\vee$) and 3 ($\blacksquare, \Diamond$) is guaranteed by the truth of ϕ in the model. ■

We are now in a position to prove what we claimed earlier:

THEOREM 19. (**a**) *$\mathcal{L}_{\mathbf{SD}}$ has the strong finite model property;* (**b**) *The satisfiability problem of $\mathcal{L}_{\mathbf{SD}}$ is* **PSPACE***-complete.*

Proof. Given the algorithm $\mathfrak{A}(\cdot)$, both (a) and the fact that $\mathcal{L}_{\mathbf{SD}}$-satisfiability is decidable in **PSPACE** are immediate from Proposition 18. $\mathcal{L}_{\mathbf{SD}}$-satisfiability is **PSPACE**-hard, because **ML** $\leq \mathcal{L}_{\mathbf{SD}}$ (Fact 7), and **ML**-satisfiability is **PSPACE**-hard (cf., e.g., [3, Th. 6.50]). ■

9 Concluding remarks

Theorems 14, 17 and 19 tell us that $\mathcal{L}_{\mathbf{SD}}$ is a computationally attractive logic that does not lie within the guarded fragment of first-order logic. In particular, for validity and satisfiability, this logic has the same computational properties as basic modal logic. Translatability into the guarded fragment is currently taken as one of the key factors − perhaps *the* key factor − in 'explaining' why modal logics have such a good computational behavior (cf., e.g., [2, 12]). An earlier influential candidate for this role was translatability into the two-variable fragment of first-order logic [7], later called into question (see, e.g., [22]). Incidentally, by Theorem 10, $\mathcal{L}_{\mathbf{SD}}$ is not translatable into any finite variable fragment of **FO**, while indeed translatable into **FO** itself. It was of course not the intention of Andréka, van Benthem and Németi [1] to suggest that *every* modal decidability result would be explained by the fact that the modal logic in question is a fragment of **GF**. However, in practice the relationship between translatability into **GF** and good computational properties of modal logics are seen as so intimately tied together, that we think that our results bring up a conceptually important point: either $\mathcal{L}_{\mathbf{SD}}$ is not a modal logic, or else the hypothesis about the importance of **GF** in accounting for modal decidability results should be taken with a grain of salt.

If $\mathcal{L}_{\mathbf{SD}}$ is not a modal logic, the operation of *IF-ing* is capable of destroying the *Eigenart* of its input: the result of *IF-ing* basic modal logic would no longer be a modal logic. If so, modal logic is intrinsically free of imperfect information, incapable of flexibly exhibiting logical independence. In practice this would mean putting an extreme stress on the locality of modal semantics; for this is perhaps the single most important thing that is destroyed by the IF modal logical frameworks. How could we argue in favor of the modal character of $\mathcal{L}_{\mathbf{SD}}$? On the list there would be the issues of Beth definability property and the existence of a characteristic bisimulation relation. (For IF first-order logic, a characteristic Ehrenfeucht-Fraïssé game is provided in [20].) We leave these questions open.

Granting, then, that $\mathcal{L}_{\mathbf{SD}}$ is a modal logic, how severely would our results tell against the hypothesis that **GF** holds the key to understanding the nice computational properties of modal logics? We have shown that there is no *truth* preserving translation of $\mathcal{L}_{\mathbf{SD}}$ into **GF**. What about an embedding of $\mathcal{L}_{\mathbf{SD}}$ into **GF** that simply preserves *satisfiability*? Now trivially, for any logic L whose satisfiability problem happens to lie in **2EXPTIME** (the complexity of **GF**-satisfiability), there exists a polynomial time reduction $t : L \longrightarrow \mathbf{GF}$ such that for any formula $\phi \in L$, ϕ is satisfiable iff $t(\phi)$ is satisfiable. Hence the mere existence of such an embedding cannot possibly have 'explanatory' value. Constructing a suitable satisfiability preserving transformation $\mathcal{L}_{\mathbf{SD}} \longrightarrow \mathbf{GF}$ might still of course be useful in getting to grips with the decidability of $\mathcal{L}_{\mathbf{SD}}$-satisfiability. But there may be another way to rehabilitate the status of the guarded fragment in an IF modal logical context. While not translatable into **GF**, $\mathcal{L}_{\mathbf{SD}}$ is trivially translatable into the result of *IF-ing* **GF**, call it **IF**(**GF**). It is an open question whether the satisfiability and validity problems of this fragment of IF first-order logic are decidable. If they are, facts about the decidability of $\mathcal{L}_{\mathbf{SD}}$ could be explained by reference to **IF**(**GF**) like the decidability of **ML** is explained by reference to **GF**. We would wish to point out that it follows from the results of [19] that **IF**(**GF**) does not fall within **FO**; it is a rather powerful language.

Acknowledgments

We wish to thank Johan van Benthem and Peter van Emde Boas for helpful discussions, and the three anonymous referees for their useful comments. We are grateful to NWO for financially supporting the work of the first author (visitor's grant B 62-608).

BIBLIOGRAPHY

[1] H. Andréka, J. van Benthem, and I. Németi. Modal languages and bounded fragments of predicate logic. *J. Phil. Logic*, 27:217–74, 1998.

[2] C. Areces, C. Monz, H. de Nivelle, and M. de Rijke. The guarded fragment: Ins and outs. In J. Gerbrandy, M. Marx, M. de Rijke, and Y. Venema, editors, *JFAK. Essays Dedicated to Johan van Benthem on the Occasion of his 50th Birthday*. Amsterdam UP, 1999.

[3] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge UP, 2002.

[4] J. Bradfield. Independence: Logics and concurrency. In *Proceedings of the 14th Workshop on Computer Science Logic*, volume 1862 of *LNCS*. Springer, 2000.

[5] J. Bradfield and S. Fröschle. Independence-friendly modal logic and true concurrency. *Nordic J. of Computing*, 9(2):102–17, 2002.

[6] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer, 1999.

[7] D. Gabbay. Expressive functional completeness in tense logic. In U. Mönnich, editor, *Aspects of Philosophical Logic*, pages 91–117. Reidel, 1981.

[8] E. Grädel. On the restraining power of guards. *J. Symbolic Logic*, 64:1719–42, 1997.

[9] E. Grädel and M. Otto. On logics with two variables. *Theor. Comp. Sc.*, 224:73–113, 1999.

[10] L. Henkin. Completeness in the theory of types. *J. Symbolic Logic*, 15(2):81–91, 1950.

[11] J. Hintikka. *Principles of Mathematics Revisited*. Cambridge UP, 1996.

[12] E. Hoogland, M. Marx, and M. Otto. Beth definability and the guarded fragment. In *Proceedings of LPAR'99*, volume 1705 of *LNAI*, pages 273–85. Springer, 1999.

[13] T. Hyttinen and T. Tulenheimo. Decidability of IF modal logic of perfect recall. In R. Schmidt, I. Pratt-Hartmann, M. Reynolds, and H. Wansing, editors, *Advances in Modal Logic*, volume 5, pages 111–31. KCL Publ., 2005.

[14] R. E. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM J. Comp.*, 6(3):467–80, 1977.

[15] M. Mortimer. On languages with two variables. *Zeitschr. f. math. Logik und Grundlagen d. Math.*, 21:135–40, 1975.

[16] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[17] T. Tulenheimo. On IF modal logic and its expressive power. In Ph. Balbiani, N.-Y. Suzuki, F. Wolter, and M. Zakharyaschev, editors, *Advances in Modal Logic*, volume 4, pages 475–98. KCL Publ., 2003.

[18] T. Tulenheimo. *Independence-Friendly Modal Logic*. PhD thesis, University of Helsinki, 2004.

[19] T. Tulenheimo and M. Sevenster. Approaches to independence-friendly modal logic. Submitted, 2006.

[20] J. Väänänen. On the semantics of informational independence. *Logic J. IGPL*, 10(3):337–50, 2002.

[21] J. Väänänen. Independence friendly logic. Dept. of Mathematics, U. of Helsinki, ms., 2006.

[22] M. Vardi. Why is modal logic so robustly decidable? In *Descriptive complexity and finite models*, volume 31 of *Series in Discr. Math. & Theor. Comp. Science*, pages 149–84. AMS, 1998.

Tero Tulenheimo
Academy of Finland; Department of Philosophy, University of Helsinki
P.O. Box 9, 00014 University of Helsinki, Finland
tero.tulenheimo@helsinki.fi

Merlijn Sevenster
Institute for Logic, Language and Computation, University of Amsterdam
Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands.
sevenstr@science.uva.nl