Undecidability of Multi-modal Hybrid Logics[⋆]

Martin Mundhenk[1] Thomas Schneider[2]

Institut für Informatik
Friedrich-Schiller-Universität
Jena, Germany

Abstract

This paper establishes undecidability of satisfiability for multi-modal logic equipped with the hybrid binder ↓, with respect to frame classes over which the same language with only one modality is decidable. This is in contrast to the usual behaviour of many modal and hybrid logics, whose uni-modal and multi-modal versions do not differ in terms of decidability and, quite often, complexity. The results from this paper apply to a wide range of frame classes including temporally and epistemically relevant ones.

Key words: Computational Complexity, Downarrow Operator, Hybrid Logic, Modal Logic

1 Introduction

The bottom line of this paper can be informally summarized by the warning

> *If you hybridize a multi-modal logic $\mathcal{L}_n$, then expect it to become undecidable — even if you only consider frame classes over which the uni-modal hybridized $\mathcal{L}$ is decidable.*

We explain this statement and formulate it more precisely.

This paper examines the effects of the interaction between the hybrid downarrow operator (↓) and multiple modalities on the decidability of the satisfiability problem of modal logics. The ↓ operator is a very powerful and desirable means of expression. It allows for binding names to points in a model (states, points in time, ...) and for referring to these points later on. But this high expressivity makes this operator dangerous in terms of computational costs. Satisfiability for modal logic equipped with ↓ is undecidable in general [1]. However, over restricted frame classes, such as transitive frames,

[⋆] This paper is a refined version of [12], refining Theorem 3.2 from that paper.

[1] Email: `mundhenk@cs.uni-jena.de`

[2] Email: `schneider@cs.uni-jena.de`

arbitrary	transitive	transitive trees	linear	equivalence relations	width $0, 1$	finite width ≥ 2
coRE [1]	NExp [13]	PSpace [7,c.]	NP [7]	NExp [13]	NP [17]	NExp [17]

Table 1

Complexity results (completeness) for the hybrid ↓-language with respect to different frame classes. A conclusion from cited work is denoted by "c".

transitive trees, linear orders, or equivalence relations, ↓ is either of no use at all, or the expressive power added does not lead to undecidability [13]. We show that for these and other frame classes, satisfiability becomes undecidable in the bi-modal case.

The consequences of the ↓ operator for the satisfiability problem of hybrid logics have been examined in many respects. It has been shown in [17] that decidability over arbitrary frames can be regained under certain syntactic restrictions concerning the interaction of ↓ and the modal operator $\Box$. In the same paper, decidability has been recovered by restricting the frame class to uni-modal frames of bounded width. Other semantic restrictions by means of temporally relevant frame classes have been shown to sustain decidability in [7] and [13]. In the case of [7], even interactions of ↓ with other hybrid operators have been allowed. The complexity results for different frame classes are summarized in Table 1. Complexity classes are used as defined in [15].

The contribution of this paper is to be seen from two points of view. On the one hand, our results will imply that many of the decidability statements from Table 1 do not carry over to the multi-modal ↓-language. On the other side, this also means that even if we restrict ourselves to frame classes over which ↓ seems to be (mostly) harmless, adding ↓ to a *multi-modal* language is much worse in terms of decidability than adding it to a *uni-modal* language. This is how the above warning shall be understood.

Precisely speaking, we prove the following results.

(1) For each frame class containing one particular linear frame, satisfiability of the bi-modal ↓-language is undecidable.

(2) For each frame class containing one particular ER[3] frame, satisfiability of the bi-modal ↓-language is undecidable.

It is worth noting that each of these two statements involves a wide range of frame classes, including temporally (in the first case) and epistemically (in the second case) relevant ones. This is in agreeable contrast to the fact that most techniques used to establish complexity results for modal and hybrid logics are not easily transferable to other frame classes. Two positive examples for results involving more than one frame class can be found in [11] and [16]. According to our understanding, the generality of our results is due to the enormous expressive power of ↓ that allows for forcing an arbitrary frame to

[3] An ER frame is a multi-modal frame in which each relation is an equivalence relation.

lang.	arbitrary	transitive	transitive trees	linear	equivalence relations (ER)
$\mathcal{ML}$	PSpace [11]	PSpace [11]	PSpace [11,c.]	NP [14]	NP [11]
$\mathcal{ML}_n$	PSpace [10]	PSpace [10]	PSpace-hard [11]	NP-hard [14]	PSpace [10]
$\mathcal{HL}$	PSpace [1]	PSpace [2]	PSpace [2]	NP [2]	NP [11,c.]
$\mathcal{HL}_n$	PSpace [1]	PSpace [2]	PSpace-hard [11]	NP-hard [14]	PSpace [10,c.]
$\mathcal{HL}^{\downarrow}$	coRE [1]	NExp [13]	PSpace [7,c.]	NP [7]	NExp [13]
$\mathcal{HL}^{\downarrow}_n$	coRE [1,c.]	**coRE** (3.1)	**coRE** (3.1)	**coRE** (3.1)	**coRE** (3.2)

Table 2
Complexity results (completeness) for modal and hybrid languages with respect to different frame classes. A conclusion from cited work is denoted by “c”. Our results are typeset in bold, accompanied by the number of the respective theorem.

have many important and very specific properties.

Furthermore, our results give another insight into the lack of robustness exhibited by ↓ languages. The term “robust” is used in a similar manner as in [9], here denoting the property that the passage from a uni-modal logic to its multi-modal version does not destroy decidability or complexity. Many, but not all, modal and hybrid logics without ↓ are robust in this sense [10,1,2,8,4], but we will show that ↓-languages lack such a robustness. This contrast becomes vivid in Table 2 which contains complexity results for modal ($\mathcal{ML}$) and hybrid ($\mathcal{HL}$) languages with respect to frame classes covered by our results, contrasting uni-modal and multi-modal versions.

This paper is organized as follows. In Section 2 we give the necessary definitions of hybrid logic and tilings, the tool used to establish undecidability. Section 3 contains our results, and Section 4 concludes the paper.

2 Preliminaries

We define the basic concepts and notations of hybrid logic and tilings. The fundamentals of hybrid logic can be found in [1,5]; tilings are defined in [18].

2.1 Hybrid Logic

Hybrid languages are extensions of the modal language allowing for explicit references to states. Here we introduce the languages relevant for our work. The definitions and notations are taken from [1,2].

Syntax. Let PROP be a countable set of *propositional atoms*, NOM be a countable set of *nominals*, SVAR be a countable set of *state variables*, ATOM $=$ PROP $\cup$ NOM $\cup$ SVAR, and $n \in \mathbb{N}_{>0}$. It is common practice to denote propositional atoms by $p, q, \dots$, nominals by $i, j, \dots$, and state variables by $x, y, \dots$ The *full n-modal hybrid language* $\mathcal{HL}^{\downarrow,@}_n$ is the set of all formulae of the form

$$\varphi ::= a \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \Diamond_\ell \varphi \mid @_t \varphi \mid {\downarrow} x.\varphi \,,$$

where $a \in \mathrm{ATOM}$, $t \in \mathrm{NOM} \cup \mathrm{SVAR}$, $x \in \mathrm{SVAR}$, and $\ell \in \{1, \dots, n\}$. We use the well-known abbreviations $\vee$, $\rightarrow$, $\leftrightarrow$, $\top$ ("true"), and $\bot$ ("false"), as well as $\Box_\ell \varphi = \neg\Diamond_\ell \neg\varphi$.

Whenever we leave $\downarrow$ or @ out of the hybrid language, we omit the according superscript of $\mathcal{HL}_n$. We call the modal language (i. e. without nominals, @, and $\downarrow$) $\mathcal{ML}_n$. In the uni-modal case, we omit the subscript 1.

Semantics for $\mathcal{HL}_n^{\downarrow,@}$ is defined in terms of *Kripke models.* A Kripke model is a triple $\mathcal{M} = \big(M, (R_1, \dots, R_n), V\big)$, where M is a nonempty set of *states*, $R_\ell \subseteq M \times M$ are binary relations — the *accessibility relations* —, and $V : \mathrm{PROP} \rightarrow \mathfrak{P}(M)$ is a function — the *valuation function.* The structure $\mathcal{F} = \big(M, (R_1, \dots, R_n)\big)$ is called a *frame.*

A *multi-modal hybrid model* is a Kripke model with the valuation function V extended to $\mathrm{PROP} \cup \mathrm{NOM}$, where for all $i \in \mathrm{NOM}$, $|V(i)| = 1$. Whenever it is clear from the context, we will omit "hybrid" and/or "multi-modal" when referring to models.

In order to evaluate $\downarrow$-formulae, an *assignment* $g : \mathrm{SVAR} \rightarrow M$ *for* $\mathcal{M}$ is necessary. Given an assignment g, a state variable x and a state m, *an* x*-variant* g_m^x *of* g is defined by $g_m^x(x) = m$ and $g_m^x(x') = g(x')$ for all $x' \neq x$. For any atom a, let $[V, g](a) = \{g(a)\}$ if $a \in \mathrm{SVAR}$, and $V(a)$, otherwise.

Given a model $\mathcal{M} = \big(M, (R_1, \dots, R_n), V\big)$, an assignment g, and a state $m \in M$, the satisfaction relation for hybrid formulae is defined by

$$\begin{array}{lll}
\mathcal{M}, g, m \models a & \text{iff} & m \in [V, g](a),\ a \in \mathrm{ATOM}, \\
\mathcal{M}, g, m \models \neg\varphi & \text{iff} & \mathcal{M}, g, m \not\models \varphi, \\
\mathcal{M}, g, m \models \varphi \wedge \psi & \text{iff} & \mathcal{M}, g, m \models \varphi \;\&\; \mathcal{M}, g, m \models \psi, \\
\mathcal{M}, g, m \models \Diamond_\ell \varphi & \text{iff} & \exists n \in M (m R_\ell n \;\&\; \mathcal{M}, g, n \models \varphi), \\
\mathcal{M}, g, m \models @_t \varphi & \text{iff} & \exists n \in M (\mathcal{M}, g, n \models \varphi \;\&\; [V, g](t) = \{n\}), \\
\mathcal{M}, g, m \models {\downarrow} x.\varphi & \text{iff} & \mathcal{M}, g_m^x, m \models \varphi.
\end{array}$$

A formula φ is *satisfiable* if there exist a model $\mathcal{M} = \big(M, (R_1, \dots, R_n), V\big)$, an assignment g for $\mathcal{M}$, and a state $m \in M$, such that $\mathcal{M}, g, m \models \varphi$.

Properties of Models and Frames. Let $\mathcal{M} = \big(M, (R_1, \dots, R_n), V\big)$ be a hybrid model with the underlying frame $\mathcal{F} = \big(M, (R_1, \dots, R_n)\big)$. By R_ℓ^+ we denote the transitive closure of R_ℓ.

If we require the accessibility relations to have certain properties, we restrict the class of relevant frames. The frame classes used in this paper are defined in Table 3, where only the uni-modal case $\mathcal{F} = (M, R)$ is considered. If we speak of a multi-modal frame having one of these properties, we mean a frame $\mathcal{F} = \big(M, (R_1, \dots, R_n)\big)$ such that each (M, R_ℓ) has this property.

frame class	abbr.	properties
arbitrary frames	—	—
trees	tree	acyclic, each point has at most one R-successor
transitive frames	trans	R is transitive
transitive trees	tt	$R = S^+$, where (M, S) is a tree
linear orders	lin	R is transitive, irreflexive, and trichotomous — trichotomy: $\big(\forall xy(xRy$ or $x{=}y$ or $yRx)\big)$
ER frames	ER	R is an equivalence relation

Table 3
Relevant frame classes, their abbreviations and definitions

Satisfiability Problems. For any hybrid language $\mathcal{HL}_n^x$ and any frame class $\mathfrak{F}$, the *satisfiability problem* $\mathcal{HL}_n^x$-$\mathfrak{F}$-SAT is defined as follows: Given a formula $\varphi \in \mathcal{HL}_n^x$, do there exist a hybrid model $\mathcal{M}$ based on a frame from $\mathfrak{F}$, an assignment g for $\mathcal{M}$, and a state $m \in M$ such that $\mathcal{M}, g, m \models \varphi$? (If $\downarrow$ is not in the considered language, the assignment g can be left out of this formulation.) For example, the satisfiability problem over transitive frames for the bi-modal hybrid $\downarrow$ language is denoted by $\mathcal{HL}_2^{\downarrow}$-trans-SAT.

2.2 Tilings

Domino tiling problems trace back to Wang [19]. A *tile* is a unit square, divided into four triangles by its diagonals. A *tile type* is a colouring of these four triangles and cannot be rotated. More formally, a tile type T is a quadruple $T = \big(\text{left}(T), \text{right}(T), \text{top}(T), \text{bot}(T)\big)$ of colours. Given a set $\mathcal{T}$ of tile types, a $\mathcal{T}$*-tiling* is a complete covering of the $\mathbb{Z} \times \mathbb{Z}$ grid with tiles having types from $\mathcal{T}$, such that each point (x, y) is covered by exactly one tile and adjacent tiles have the same colour at their common edges. Formally, a $\mathcal{T}$-tiling is a function $\tau : \mathbb{Z} \times \mathbb{Z} \to \mathcal{T}$ satisfying the following condition for all $(x, y) \in \mathbb{Z} \times \mathbb{Z}$.

$$\text{right}\big(\tau(x,y)\big) = \text{left}\big(\tau(x+1,y)\big) \quad \& \quad \text{top}\big(\tau(x,y)\big) = \text{bot}\big(\tau(x,y+1)\big). \quad (1)$$

Given a tile type $T \in \mathcal{T}$, we define $\text{RI}(T, \mathcal{T}) = \{T' \in \mathcal{T} \mid \text{right}(T) = \text{left}(T')\}$ and $\text{UP}(T, \mathcal{T}) = \{T' \in \mathcal{T} \mid \text{top}(T) = \text{bot}(T')\}$ in order to denote the sets of tile types that match T horizontally or vertically, respectively, in a $\mathcal{T}$-tiling. Condition (1), then, is equivalent to

$$\tau(x+1,y) \in \text{RI}\big(\tau(x,y)\big) \quad \& \quad \tau(x,y+1) \in \text{UP}\big(\tau(x,y)\big).$$

The *tiling problem* denotes the question whether a given set $\mathcal{T}$ of tile types admits a $\mathcal{T}$-tiling of the $\mathbb{Z} \times \mathbb{Z}$ grid. This problem is coRE-complete, hence undecidable [3]. It remains coRE-complete if the grid is restricted to the first quadrant, i. e. $\mathbb{N} \times \mathbb{N}$. We will make use of both versions in Section 3.

3 Undecidability of Multi-modal Downarrow Logic

We have observed in Section 1 that for many modal languages, as well as for the basic hybrid language, algorithms deciding their satisfiability can straightforwardly be applied to multi-modal versions of these languages without significant changes. Hence the complexity often does not increase when proceeding from uni-modal to multi-modal languages. However, concerning $\mathcal{HL}^{\downarrow}$, this is not the case because the fundamental properties that led to the proofs of the decidability results do not carry over to multi-modal versions of this language.

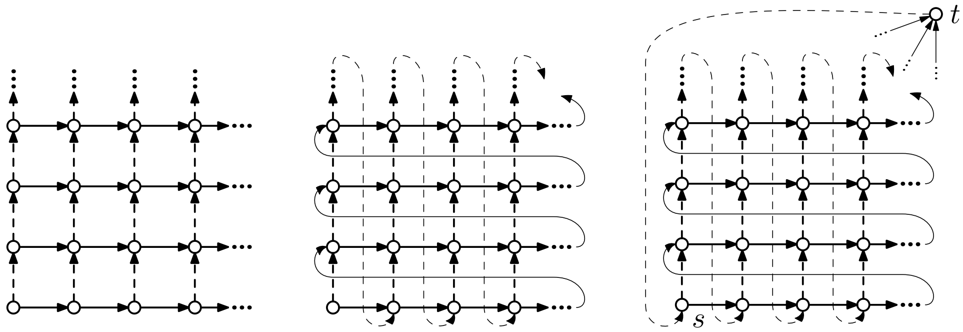
In the case of acyclic frames (linear orders or transitive trees), this "fundamental property" is the simple fact that due to the lack of cycles, we can never get back to points named by $\downarrow$. In a frame with two acyclic accessibility relations, however, cycles are possible. For transitive frames, the "fundamental property" consists of the fact that each cycle is a cluster, i. e. a complete subframe. In a transitive frame for a multi-modal language, there can be cycles consisting of edges of different accessibility relations which are not necessarily clusters. This renders the argumentation in the respective proof untransferable even to bi-modal $\mathcal{HL}^{\downarrow}$. In the case of equivalence relations, the "fundamental property" is the fact that $\mathcal{HL}^{\downarrow}$ is equivalent to the monadic class of first-order logic. This equivalence cannot be established for the bi-modal language.

The bi-modal language with $\downarrow$ is in fact strong enough to encode tilings on *any* frame class between linear and arbitrary frames. This will lead to the result in Subsection 3.1. Tilings can also be encoded on any frame class between ER frames and arbitrary frames, although three modalites are needed in this case. This result is given in Subsection 3.2. The expressive power of $\downarrow$ becomes evident in both encodings.

3.1 Between Linear Orders and Arbitrary Frames

In this subsection, we show that $\mathcal{HL}_2^{\downarrow}$ is able to encode tilings of $\mathbb{N}\times\mathbb{N}$ on *any* frame class containing one particular linear frame, which we will call Grid in the following. This ability is not too surprising if one considers the fact that $\downarrow$ is powerful enough to force the two accessibility relations to behave as the "right neighbour" and "upper neighbour" relations in the $\mathbb{N}\times\mathbb{N}$ grid. Since we are interested in a result as general as possible, we will have to insist on Grid having two linear (i. e. transitive, irreflexive, and trichotomous) accessibility relations when constructing this frame. This may seem artificial at some point, but is justified by the aim to cover as many frame classes as possible.

In order to construct Grid, we start with two accessibility relations R_h ("horizontal") and R_v ("vertical"). The frame will consist of points $(x, y) \in \mathbb{N}^2$, where $(x, y)R_h(x', y')$ whenever $x < x'$ and $y = y'$, and $(x, y)R_v(x', y')$ whenever $x = x'$ and $y < y'$. This situation is shown in Figure 1 (a), where a full line denotes an R_h edge, and a dashed line stands for an R_v edge. Note that the transitive closure of both relations is implicit. Clearly, R_h and R_v are irreflexive. For reasons just stated, we make them trichotomous by adding



(a) Transitive and irreflexive

(b) In addition, trichotomous

(c) The frame Grid with spy-/sinkpoint

Fig. 1. Simulating the $\mathbb{N} \times \mathbb{N}$ grid with two relations. The transitive closures are not drawn.

extra edges as given in Figure 1 (b) and taking the transitive closure again. More precisely speaking, we make each point on the nth row see each point on the mth row via R_h, for each $m > n$; and we make each point on the nth column see each point on the mth column via R_v, for each $m > n$.

We will need to refer to the lower left point (the "origin" of the grid) several times. For this purpose, we introduce a variant of the Spypoint Technique [6,1]. Apart from the fact that the "origin" behaves almost as a spypoint — i. e. all other points in Grid are accessible from it via some R_h-R_v-path —, we will add a sinkpoint to the model that is accessible from all other points via R_h and that sees the spypoint via R_v, cf. Figure 1 (c). Note that the spypoint-sinkpoint construction does not destroy irreflexivity or trichotomy.

Let ∞ denote the sinkpoint. We formally define $\mathsf{Grid} = \big(N, (R_h, R_v)\big)$ by

$$\begin{aligned} N &= \mathbb{N}^2 \cup \{\infty\} \qquad (\mathbb{N}^2 = \mathbb{N} \times \mathbb{N}), \\ R_h &= \big\{\big((x,y),(x',y')\big) \in (\mathbb{N}^2)^2 \mid (y = y' \text{ and } x < x') \text{ or } y < y'\big\} \cup \big(\mathbb{N}^2 \times \{\infty\}\big), \\ R_v &= \big\{\big((x,y),(x',y')\big) \in (\mathbb{N}^2)^2 \mid (x = x' \text{ and } y < y') \text{ or } x < x'\big\} \cup \big(\{\infty\} \times \mathbb{N}^2\big). \end{aligned}$$

Clearly, Grid is a linear frame. Whenever we will construct a model based on Grid, we will name the spypoint s and the sinkpoint t, where s and t are nominals. This is reflected in Figure 1 (c), too. We now formulate our result as general as possible.

Theorem 3.1 *For any bi-modal frame class $\mathfrak{F}$ with* Grid $\in \mathfrak{F}$*, $\mathcal{HL}_2^{\downarrow}$-$\mathfrak{F}$-SAT is undecidable.*

Proof. Let $\mathcal{T}$ be a set of tile types. We define a formula $\varphi_{\mathcal{T}}$ that implements the grid and expresses the tiling. This formula has to be equipped with two properties. On the one hand, it must be satisfied in some model based on Grid, given a $\mathcal{T}$-tiling. On the other side, $\varphi_{\mathcal{T}}$ must enforce that each satisfying *arbitrary* model behaves as the $\mathcal{T}$-tiled $\mathbb{N} \times \mathbb{N}$ grid. Hence, when constructing $\varphi_{\mathcal{T}}$, we will have to enforce properties like for example transitivity or convergence that hold naturally in Grid, while we do not need to enforce e. g. trichotomy.

We start with the conjuncts of $\varphi_{\mathcal{T}}$ responsible for the grid.

- *The spypoint and sinkpoint are as given in Figure 1 (c).*

$$\mathrm{SPY} = s \wedge \Diamond_h\big(t \wedge \Diamond_v s\big)$$

Before we proceed, we define a useful abbreviation that allows us to refer only to points that are not the sinkpoint.

$$\Diamond_h^{\neg t}\psi = \Diamond_h(\neg t \wedge \psi) \qquad\qquad \Box_h^{\neg t}\psi = \neg\Diamond_h^{\neg t}\neg\psi$$

Another shortcut is used for the "reflexive closure" of the modal operators.

$$\Diamond_v^*\psi = \psi \vee \Diamond_v\psi \qquad\qquad \Box_v^*\psi = \neg\Diamond_v^*\neg\psi$$
$$\Diamond_h^*\psi = \psi \vee \Diamond_h^{\neg t}\psi \qquad\qquad \Box_h^*\psi = \neg\Diamond_h^*\neg\psi$$

Note that the definition of $\Diamond_h^*$ already includes $\Diamond_h^{\neg t}$, hence we do not need to state "$\neg t$" explicitly whenever we use $\Diamond_h^*$ or $\Box_h^*$.

From now on, we will call all points other than the sinkpoint that are accessible from s via a sequence consisting of at most one R_v edge and at most one R_h edge *R_v-R_h-reachable.* Within the set of all R_v-R_h-reachable points, we can simulate the @ operator. Suppose x is bound to such a point, then we can assert $@_x\psi$ at *any* other point by going directly to the sinkpoint, from there to the spypoint and then to the point to which x is bound. This idea is captured by the following definition.

$$@_x\psi = \Diamond_h\Big(t \wedge \Diamond_v\big(s \wedge \Diamond_v^*\Diamond_h^*(x \wedge \psi)\big)\Big)$$

Note that $@_x\psi$ only works if the point to which x is bound is R_v-R_h-reachable. On the other hand, the point y at which $@_x\psi$ is satisfied, is enforced to see the sinkpoint horizontally. (As an aside, we could even simulate the "somewhere" modality E if we left out x on the right-hand side of the above definition.)

For the @ operator and subsequent conjuncts to function properly even on arbitrary frames, it will be necessary to require that every point accessible from R_v-R_h-reachable points is R_v-R_h-reachable again. This is ensured by the following formula enforcing that both relations are transitive within the grid.

- *For every R_v-R_h-reachable point x, each point accessible from x via two R_v (or R_h) edges is accessible from x in* one *R_v (or R_h) step.*

$$\mathrm{TRANS} = \Box_v^*\Box_h^*{\downarrow}x.\big(\Box_h^{\neg t}\Box_h^{\neg t}{\downarrow}y.@_x\Diamond_h y \wedge \Box_v\Box_v{\downarrow}y.@_x\Diamond_v y\big)$$

At first glance, the fact that TRANS uses the @ operator, while the @ operator seems to act on the assumption that the relations are transitive, appears to expose a cyclic definition. This is not the case because TRANS operates in an inductive manner, which will become clear further below when the tiling is constructed from a model satisfying $\varphi_{\mathcal{T}}$.

We will need to refer to *neighbours* of points. A point y is a *right neighbour* of x if xR_hy and there is no z such that xR_hzR_hy. *Upper neighbours* are defined analogously. In order to express neighbours, we define "next" operators to be the following abbreviations.

$$\begin{aligned}\bigcirc_h\psi &= {\downarrow} a.\Diamond_h^{\neg t} {\downarrow} b.\big(@_a\neg\Diamond_h\Diamond_h b \wedge \psi\big)\\ \bigcirc_v\psi &= {\downarrow} a.\Diamond_v {\downarrow} b.\big(@_a\neg\Diamond_v\Diamond_v b \wedge \psi\big)\end{aligned}$$

Whenever $\bigcirc_h$ and $\bigcirc_v$ are employed in the following, a and b must be substituted by fresh state variables. Note that these operators are diamond-style. We will not introduce an abbreviation for their duals. After we have required every R_v-R_h-reachable point to have exactly one right and one upper neighbour, the new next operators can be used box-style, as well.

- *Every R_v-R_h-reachable point has exactly one right and exactly one upper neighbour.*

$$\mathrm{NEIGH} = \Box_v^*\Box_h^* {\downarrow} x.\big(\bigcirc_h {\downarrow} y.\,@_x\neg\bigcirc_h\neg y \;\;\wedge\;\; \bigcirc_v {\downarrow} y.\,@_x\neg\bigcirc_v\neg y\big)$$

- *For every R_v-R_h-reachable point x, the unique point y that is the right neighbour of the upper neighbour of x coincides with the upper neighbour of the right neighbour of x.*

$$\mathrm{CONV} = \Box_v^*\Box_h^* {\downarrow} x.\bigcirc_v\bigcirc_h {\downarrow} y.\,@_x\bigcirc_h\bigcirc_v y$$

Having implemented the grid, it is straightforward to express the tiling on it. For this purpose, we define an atomic proposition T for each tile type in $T \in \mathcal{T}$. For the sake of short notation, we will deliberately confuse tile types with their associated atoms.

- *At each point in the grid lies exactly one tile.*

$$\mathrm{TILE} = \Box_v^*\Box_h^* \textstyle\bigvee_{T\in\mathcal{T}} \Big(T \wedge \bigwedge_{T'\neq T} \neg T'\Big)$$

- *The tiling conditions are met.*

$$\mathrm{MATCH} = \Box_v^*\Box_h^* \textstyle\bigwedge_{T\in\mathcal{T}} \Big(T \rightarrow \big(\bigvee_{T'\in\mathrm{UP}(T,\mathcal{T})} \bigcirc_v T' \;\;\wedge\;\; \bigvee_{T'\in\mathrm{RI}(T,\mathcal{T})} \bigcirc_h T'\big)\Big)$$

Let $\varphi_{\mathcal{T}} = \mathrm{SPY}\wedge\mathrm{TRANS}\wedge\mathrm{NEIGH}\wedge\mathrm{CONV}\wedge\mathrm{TILE}\wedge\mathrm{MATCH}$. In order to prove the statement of this theorem, it is sufficient to show that the following two propositions hold:

(i) If $\mathcal{T}$ admits a tiling, then $\varphi_{\mathcal{T}}$ is satisfiable in Grid.

(ii) If $\varphi_{\mathcal{T}}$ is satisfiable in an arbitrary model, then $\mathcal{T}$ admits a tiling.

Proof of (i). Suppose $\mathcal{T}$ is given and admits a tiling of $\mathbb{N}^2$. Then there exists a function $\tau : \mathbb{N}^2 \rightarrow \mathcal{T}$ such that for all $(x, y) \in \mathbb{N}^2$, Condition (1) from

Page 5 holds. We construct a model $\mathcal{M} = \big(N, (R_h, R_v), V\big)$ based on Grid, where V is defined by $V(s) = \{(0,0)\}$, $V(t) = \{\infty\}$, and $V(T) = \{(x,y) \mid \tau(x,y) = T\}$ for each $T \in \mathcal{T}$.

We claim that $\mathcal{M}, (0,0) \models \varphi_{\mathcal{T}}$ and show that each conjunct of $\varphi_{\mathcal{T}}$ is satisfied at $(0,0)$ in $\mathcal{M}$. Conjunct SPY follows directly from the definitions of R_h and R_v of Grid. Since both relations are transitive, TRANS holds. Conjuncts NEIGH and CONV are satisfied because they express basic properties of R_h and R_v that are based on $\mathbb{N}^2$. TILE and MATCH hold due to the tiling.

Proof of (ii). Let $\mathcal{M} = \big(M, (R_h, R_v), V\big)$ be an arbitrary model satisfying $\varphi_{\mathcal{T}}$. Since s, t are nominals, there exist points $m_0, m_\infty \in M$ such that $V(s) = \{m_0\}$ and $V(t) = \{m_\infty\}$. Conjunct SPY implies $m_0 R_h m_\infty$ and $m_\infty R_v m_0$. We now define a mapping $f : \mathbb{N}^2 \to M - \{m_\infty\}$ that satisfies the following conditions for all $(x,y) \in \mathbb{N}^2$.

(iii) If $x \geq 1$, then $f(x,y)$ is the right neighbour of $f(x-1,y)$.

(iv) If $y \geq 1$, then $f(x,y)$ is the upper neighbour of $f(x,y-1)$.

(v) If $x = 0$ and $y \geq 1$, then $m_0 R_v f(0,y)$.

(vi) $f(x,y)$ is R_v-R_h-reachable.

(vii) $f(x,y) R_h m_\infty$.

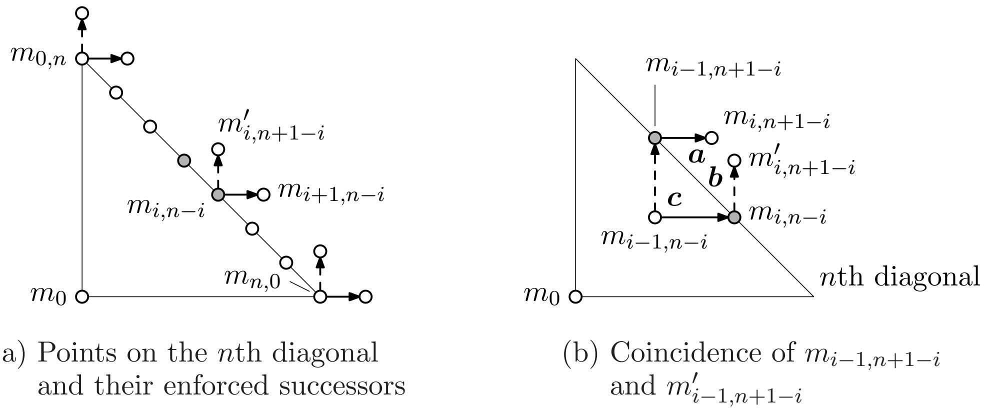
We construct f by induction on $n = x + y$, i. e. diagonal-wise with respect to $\mathbb{N}^2$. The base case consists of $n = 0, 1$. For $n = 0$, we set $f(0,0) = m_0$. Since m_0 is R_v-R_h-reachable, NEIGH together with @ implies that m_0 has a unique right neighbour $m_{1,0}$ and a unique upper neighbour $m_{0,1}$. Due to the definition of @, they both see the sinkpoint via R_h. Set $f(1,0) = m_{1,0}$ and $f(0,1) = m_{0,1}$. Now Conditions (iii)–(vii) are satisfied up to the first diagonal.

For the induction step, suppose that $f(x,y)$ has already been defined for all (x,y) with $x + y \leq n$ (i. e. from the 0th to the nth diagonal), $n \geq 1$, and Conditions (iii)–(vii) hold up to here. Consider the points on the nth diagonal, namely $m_{i,n-i} = f(i, n-i)$ for $i = 0, \ldots, n$. Because of (vi), NEIGH applies and implies that each $m_{i,n-i}$ has a unique horizontal successor $m_{i+1,n-i}$ and a unique vertical successor $m'_{i,n+1-i}$, see Figure 2 (a). Note that the @ operator works because each $m_{i,n-i}$ satisfies (vi).

Now for each $i = 1, \ldots, n-1$, the points $a = m_{i,n+1-i}$ and $b = m'_{i,n+1-i}$ coincide. To justify this claim, let $c = f(i-1, n-i)$ (lying on the $(n-1)$st diagonal). Since c has the horizontal successor $m_{i,n-i}$ which has the vertical successor b, and c has the vertical successor $m_{i-1,n+1-i}$ which has the horizontal successor a, and (vi) holds for c, CONV implies $a = b$. See also Figure 2 (b).

Let $f(0, n+1) = m'_{0,n+1}$ and $f(i, n+1-i) = m_{i,n+1-i}$, for all $i = 1, \ldots, n+1$. It follows from this construction that Conditions (iii), (iv), and (vii) are satisfied for the "new" (x,y) from the $(n+1)$st diagonal. To end the inductive construction, we have to show that the "new" (x,y) also satisfy (v) and (vi).

Condition (v) has to be shown for $(0, n+1)$. Since according to the induction hypothesis, $m_0 R_v f(0,n)$, TRANS applied to m_0 yields $m_0 R_v f(0, n+1)$.



(a) Points on the nth diagonal and their enforced successors

(b) Coincidence of $m_{i-1,n+1-i}$ and $m'_{i-1,n+1-i}$

Fig. 2. The diagonal-wise construction of the grid

Condition (vi) for $(0, n+1)$ follows from (v). For the remaining $(i, n+1-i)$, we argue as follows. Due to the induction hypothesis, $m_{i-1,n+1-i}$ is R_v-R_h-reachable. Hence there is some point a which is accessible from m_0 in at most one R_v step and from which $m_{i-1,n+1-i}$ is accessible in at most one R_h step. If the last "at most one" is in fact 0, then we are done. If it is 1, then $m_{i,n+1-i}$ is accessible from a in two R_h steps. Since a is R_v-R_h-reachable, too, TRANS applied to a yields $aR_h m_{i,n+1-i}$, hence $m_{i,n+1-i} = f(i, n+1-i)$ is R_v-R_h-reachable.

With f at our disposal, we can easily define a function $\tau : \mathbb{N}^2 \to \mathcal{T}$ as follows. Let $\tau(x,y) = T$ if and only if $f(x,y) \in V(T)$, for each $(x,y) \in \mathbb{N}^2$ and each $T \in \mathcal{T}$. The correctness of this definition is ensured by the construction of f and TILE. Because of MATCH, τ satisfies the tiling conditions. □

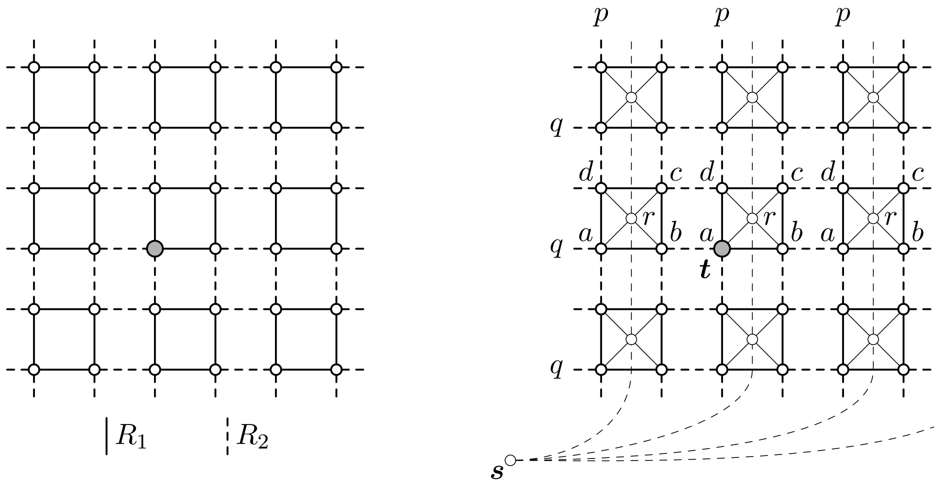
3.2 Between Equivalence Relations and Arbitrary Frames

In this subsection, we show that $\mathcal{HL}_2^{\downarrow}$ is able to encode tilings on *any* frame class containing one particular ER frame, which we will call Grid2 in the following. For the sake of an easy definition of the accessibility relations, we will consider tilings of the whole $\mathbb{Z} \times \mathbb{Z}$ grid here.

Before we state a result as general as possible, we give a construction of Grid2 and formally define this bi-modal frame to be $\mathsf{Grid2} = \big(N, (R_1, R_2)\big)$, whose components are given as follows.

- $N = (\mathbb{Z} \times \mathbb{Z}) \cup (\mathbb{Z}_2 \times \mathbb{Z}_2 \times \{+\}) \cup \{s\}$, where $\mathbb{Z}_2 = \{2z \mid z \in \mathbb{Z}\}$.
  Let N' denote $N - (\mathbb{Z} \times \mathbb{Z})$.
- $R_2 = \bigcup_{k,l\in\mathbb{Z}} \text{minicluster}(2k+1, 2l+1) \cup (N' \times N')$,
  where $\text{minicluster}(i,j) = \{(i,j), (i+1,j), (i,j+1), (i+1,j+1)\}^2$.
- $R_1 = \bigcup_{k,l\in\mathbb{Z}} \text{Minicluster}(2k, 2l)$,
  where $\text{Minicluster}(i,j) = \{(i,j), (i+1,j), (i,j+1), (i+1,j+1), (i,j,+)\}^2$.

These definitions are visualized in Figure 3, where a full line denotes an R_1 edge, and a dashed line stands for an R_2 edge. Note that due to symmetry,



(a) The grid

(b) The spypoint construction

Fig. 3. Simulating the $\mathbb{Z}\times\mathbb{Z}$ grid with two equivalence relations. Each line represents a bidirectional arrow. The transitive (and hence, reflexive) closures are not drawn.

no arrowheads appear. Furthermore, many edges implied by transitivity have not been drawn for the sake of clarity.

The idea behind the construction of N needs a detailed explanation. First, each pair $(x, y) \in \mathbb{Z} \times \mathbb{Z}$ is a state and represents the point (x, y) from the $\mathbb{Z} \times \mathbb{Z}$ grid. These points are drawn in Figure 3 (a), where R_1 and R_2, each restricted to $\mathbb{Z} \times \mathbb{Z}$, are shown as well. We call every unit square in the same drawing a *minicluster*. Second, for every R_1-minicluster, with $(2i, 2j)$ as its lower left point, there is an additional point $(2i, 2j, +)$ that belongs to the same R_1-cluster. We call these new points *local spypoints* and collect them in $\mathbb{Z}_2 \times \mathbb{Z}_2 \times \{+\}$. Finally, there is a spypoint s that sees all local spypoints via R_2. This situation is shown in Figure 3 (b).

It is easy to observe that this somewhat intricate construction is necessary. First of all, there must be a point from which all $\mathbb{Z} \times \mathbb{Z}$ points are reachable. Otherwise, the tiling conditions, for example, could not be formulated. This justifies the choice of the spypoint. Now if the spypoint were connected via each $\mathbb{Z}\times\mathbb{Z}$ point via a single R_1 or R_2 edge, this clearly would merge many R_i-miniclusters into one large cluster, which would prevent us from distinguishing between four directions in the grid. The local spypoints together with the additional R_1 and R_2 edges as drawn in Figure 3 (b), allow for accessing all $\mathbb{Z} \times \mathbb{Z}$ points from s in two steps, while keeping the miniclusters separated.

Whenever we will construct a model based on Grid2, we will name the spypoint s, where s is a nominal. The point $(0, 0)$ is named by the nominal t, and the local spypoints are labelled by the atomic proposition r. Furthermore, we will use the atomic propositions p and q to label those points that lie on an even column or row, respectively. This will enable us to distinguish between four directions. For this purpose, we define the following abbreviations.

$a = p \wedge q \wedge \neg r$	even row and even column
$b = \neg p \wedge q \wedge \neg r$	even row and odd column
$c = \neg p \wedge \neg q \wedge \neg r$	odd row and odd column
$d = p \wedge \neg q \wedge \neg r$	odd row and even column

All these settings are reflected in Figure 3 (b), too. Again, we formulate our result as general as possible, involving each class of frames containing Grid2. This includes the class of ER frames.

Theorem 3.2 *For any bi-modal frame class $\mathfrak{F}$ with* Grid2 $\in \mathfrak{F}$*,* $\mathcal{HL}^{\downarrow}_2$-$\mathfrak{F}$-SAT *is undecidable.*

Proof. Let $\mathcal{T}$ be a set of tile types. We define a formula $\varphi_{\mathcal{T}}$ that implements the grid and expresses the tiling using atomic propositions T for each $T \in \mathcal{T}$. In order to keep every part of $\varphi_{\mathcal{T}}$ short, we define two kinds of abbreviation.

First, we will have to refer to R_1-successors that are *not* local spypoints, and to R_2-successors that *are* local spypoints. This is done via new modal operators $\Diamond'_i, \Box'_i$ and $\Diamond^r_i, \Box^r_i$, where $i = 1, 2$, given by

$$\begin{aligned} \Diamond'_i\psi &= \Diamond_i(\neg r \wedge \psi), & \Diamond^r_i\psi &= \Diamond_i(r \wedge \psi),\\ \Box'_i\psi &= \neg\Diamond'_i\neg\psi, & \Box^r_i\psi &= \neg\Diamond^r_i\neg\psi. \end{aligned}$$

Second we define abbreviations that give us direct access to the left, right, upper, and lower neighbour of a given point.

$$\begin{aligned} \Diamond_l\psi &= \big(a \wedge \Diamond_2(b\wedge\psi)\big) \vee \big(b \wedge \Diamond_1(a\wedge\psi)\big) \vee \big(c \wedge \Diamond_1(d\wedge\psi)\big) \vee \big(d \wedge \Diamond_2(c\wedge\psi)\big)\\ \Diamond_r\psi &= \big(a \wedge \Diamond_1(b\wedge\psi)\big) \vee \big(b \wedge \Diamond_2(a\wedge\psi)\big) \vee \big(c \wedge \Diamond_2(d\wedge\psi)\big) \vee \big(d \wedge \Diamond_1(c\wedge\psi)\big)\\ \Diamond_u\psi &= \big(a \wedge \Diamond_2(d\wedge\psi)\big) \vee \big(b \wedge \Diamond_2(c\wedge\psi)\big) \vee \big(c \wedge \Diamond_1(b\wedge\psi)\big) \vee \big(d \wedge \Diamond_1(a\wedge\psi)\big)\\ \Diamond_d\psi &= \big(a \wedge \Diamond_1(d\wedge\psi)\big) \vee \big(b \wedge \Diamond_1(c\wedge\psi)\big) \vee \big(c \wedge \Diamond_2(b\wedge\psi)\big) \vee \big(d \wedge \Diamond_2(a\wedge\psi)\big) \end{aligned}$$

As usual, the duals are defined by $\Box_x\psi = \neg\Diamond_x\neg\psi$, $x \in \{l, r, u, d\}$. Note that we are not forced to use $\Diamond'_1$ instead of $\Diamond_1$ in these definitions, because $\neg r$ is already required in a, b, c, d.

From now on, we call a point *accessible* if it is reachable from the spypoint using an R_2 edge to a local spypoint and from there using an R_1 edge, where both edges are bidirectional. Hence, ψ is satisfied at every accessible point if $\Box^r_2\Box'_1\psi$ is satisfied at the spypoint. Now, the formula $\varphi_{\mathcal{T}}$ consists of the following conjuncts.

- *The spypoint is named s. It does not satisfy r and sees itself via R_2. The origin is accessible. It is named t and satisfies a. Every point R_2-accessible from the spypoint is labelled r.*

$$\mathrm{SPY} = s \wedge \neg r \wedge \Diamond_2 s \wedge \Diamond^r_2\Diamond'_1(t \wedge a \wedge \Diamond^r_1\Diamond'_2 s) \wedge \Box_2(\neg s \to r)$$

- *Each accessible point has a unique left, right, upper, and lower neighbour, respectively. Each of these four neighbours is accessible again. The three missing conjuncts (“. . . ”) are analogous to the first.*

$$\mathrm{NEIGH} = \Box_2^r \Box_1' {\downarrow} x. \Big[\Diamond_l {\downarrow} y. \Diamond_1^r \Diamond_2' \Big(s \wedge \Diamond_2^r \Diamond_1' \big(y \wedge \Diamond_r (x \wedge \Box_l y) \big) \Big) \wedge \ldots \Big]$$

- *Convergence holds, i. e. for each accessible point x, the (uniquely determined) point that is the right neighbour of the upper neighbour of x coincides with the upper neighbour of the right neighbour of x.*

$$\mathrm{CONV} = \Box_2^r \Box_1' \Box_u \Box_r {\downarrow} x. \Box_l \Box_d \Box_r \Box_u x$$

(Note that it suffices to replace the prefix $\Box_2^r \Box_1' \Box_u \Box_r$ by $\Box_2^r \Box_1'$, but the given definition of CONV *simplifies the considerations at the end of this proof.)*

- *At each point in the grid lies exactly one tile.*

$$\mathrm{TILE} = \Box_2^r \Box_1' \bigvee_{T \in \mathcal{T}} \Big(T \wedge \bigwedge_{T' \neq T} \neg T' \Big)$$

- *The tiling conditions are met.*

$$\mathrm{MATCH} = \Box_2^r \Box_1' \bigwedge_{T \in \mathcal{T}} \Big(T \rightarrow \big(\bigvee_{T' \in \mathrm{UP}(T,\mathcal{T})} \Box_u T' \ \wedge \ \bigvee_{T' \in \mathrm{RI}(T,\mathcal{T})} \Box_r T' \big) \Big)$$

Let $\varphi_{\mathcal{T}} = \mathrm{SPY} \wedge \mathrm{NEIGH} \wedge \mathrm{CONV} \wedge \mathrm{TILE} \wedge \mathrm{MATCH}$. Note that we only have to require certain properties of Grid2, but not all of them. For example, it is not necessary to enforce that the R_i are equivalence relations or that each four points that correspond to an R_1-minicluster from Grid2 have a *common* local spypoint. The properties enforced by $\varphi_{\mathcal{T}}$ are chosen such that they are satisfied by Grid2 on the one hand, and sufficient for a satisfying model to encode a tiling on the other hand. More precisely, it remains to prove the following two propositions.

(i) If $\mathcal{T}$ admits a tiling, then $\varphi_{\mathcal{T}}$ is satisfiable in Grid2.

(ii) If $\varphi_{\mathcal{T}}$ is satisfiable in an arbitrary model, then $\mathcal{T}$ admits a tiling.

Proposition (i) is shown as in the proof of Theorem 3.1.

Proof of (ii). Let $\mathcal{M} = \big(M, (R_1, R_2), V\big)$ be an arbitrary model satisfying $\varphi_{\mathcal{T}}$ at m_0. Because of SPY, $V(s) = \{m_0\}$. SPY also implies that there is an accessible point $m_{0,0}$ satisfying t and a. We define a mapping $f : \mathbb{Z} \times \mathbb{Z} \rightarrow M$ satisfying the following conditions for all $(x, y) \in \mathbb{Z} \times \mathbb{Z}$.

(iii) $f(x, y)$ is accessible

(iv) (a) $2 \mid x \ \Leftrightarrow \ \mathcal{M}, f(x, y) \models p$
(b) $2 \mid y \ \Leftrightarrow \ \mathcal{M}, f(x, y) \models q$

(v) (a) $x \geq 1 \ \Rightarrow \ \big(\mathcal{M}, f(x-1, y) \models p \quad \Rightarrow \ f(x-1, y) R_1 f(x, y) R_1 f(x-1, y)\big)$
(b) $x \geq 1 \ \Rightarrow \ \big(\mathcal{M}, f(x-1, y) \models \neg p \ \Rightarrow \ f(x-1, y) R_2 f(x, y) R_2 f(x-1, y)\big)$

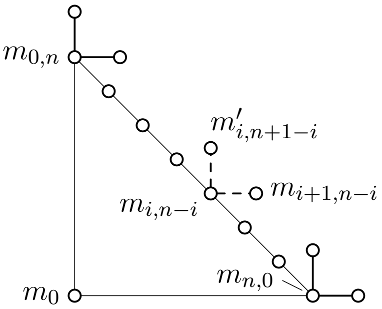


Fig. 4. Points on the nth diagonal and their enforced successors

(c) $y \geq 1 \Rightarrow \big(\mathcal{M}, f(x,y-1) \models q \quad \Rightarrow f(x,y-1) R_1 f(x,y) R_1 f(x,y-1)\big)$
(d) $y \geq 1 \Rightarrow \big(\mathcal{M}, f(x,y-1) \models \neg q \Rightarrow f(x,y-1) R_2 f(x,y) R_2 f(x,y-1)\big)$

We construct f by induction on $n = |x| + |y|$. For a given $n \in \mathbb{N}$, all points (x,y) satisfying $|x| + |y| = n$ lie on a square that is rotated by 45 degrees and whose corners are $(n,0)$, $(-n,0)$, $(0,n)$, and $(0,-n)$. In the considerations to follow, we restrict ourselves to the first quadrant, i. e. $\mathbb{N} \times \mathbb{N}$. The arguments for the other three quadrants are analogous. Note that we cannot restrict the *whole* proof to $\mathbb{N} \times \mathbb{N}$ since this would cause more intricate definitions of R_1, R_2, and NEIGH owing to an extra treatment of the margins of the grid.

The base case consists of $n = 0, 1$. Set $f(0,0) = m_{0,0}$. Now NEIGH implies that there exist accessible $m_{1,0}, m_{0,1} \in M$ such that $\mathcal{M}, m_{1,0} \models b$; $\mathcal{M}, m_{0,1} \models d$; and there exist R_1-edges in both directions between $m_{0,0}$ and each of these two new points. Set $f(1,0) = m_{1,0}$ and $f(0,1) = m_{0,1}$. Clearly, Conditions (iii)–(v) hold for all x, y with $x + y \leq 1$.

For the induction step, suppose that $f(x,y)$ has already been defined and satisfies Conditions (iii)–(v) for all (x,y) with $x+y \leq n$. Consider the points on the nth diagonal, namely $m_{i,n-i} = f(i, n-i)$ for $i = 0, \ldots, n$. Because of (iii), NEIGH applies, hence each $m_{i,n-i}$ has a unique right neighbour $m_{i+1,n-i}$ and a unique upper neighbour $m'_{i,n+1-i}$, see Figure 4.

By an argumentation analogous to that in the proof of Theorem 3.1, we conclude from CONV that $m_{i,n+1-i}$ and $m'_{i,n+1-i}$ coincide for each $i = 1, \ldots, n$. Set $f(0, n+1) = m'_{0,n+1}$ and $f(i, n+1-i) = m_{i,n+1-i}$, $i = 1, \ldots, n+1$. Now this construction and NEIGH imply (iii)–(v) for all x, y with $x + y \leq n + 1$.

Now we define $\tau : \mathbb{Z} \times \mathbb{Z} \to \mathcal{T}$ as follows. Let $\tau(x,y) = T$ if and only if $f(x,y) \in V(T)$, for each $(x,y) \in \mathbb{Z} \times \mathbb{Z}$ and each $T \in \mathcal{T}$. The correctness of this definition is ensured by the construction of f and TILE. Because of MATCH, τ satisfies the tiling conditions. □

4 Conclusion

We have shown that the interaction between multiple modalities and the ↓ operator leads to undecidability over a wide range of frame classes. This justifies the warning we gave in the Introduction. Corollary 4.1 provides evidence of the fact that our results cover frame classes well-known from temporal (1) and epistemic (2) logic. Statement (2) refers to many important frame classes whose accessibility relations are generalizations of equivalence relations.

Corollary 4.1

(1) *For any bi-modal frame class* $\mathfrak{F} \in \{\mathsf{lin}, \mathsf{tt}, \mathsf{trans}\}$*,* $\mathcal{HL}_2^{\downarrow}$-$\mathfrak{F}$-SAT *is undecidable.*

(2) *For any bi-modal frame class* $\mathfrak{F}$ *containing* ER*,* $\mathcal{HL}_2^{\downarrow}$-$\mathfrak{F}$-SAT *is undecidable.*

Let us make a technical remark concerning the results stated in Theorems 3.1 and 3.2. A closer look at the formulae $\varphi_{\mathcal{T}}$ occurring in the proofs reveals that only two nominals s and t are used. They can as well be replaced by two more bound state variables. Furthermore, the $\varphi_{\mathcal{T}}$ do not contain any free state variables. Hence, both statements do in fact hold for the nominal-free fragments of all sentences (formulae without free state variables) of $\mathcal{HL}_2^{\downarrow}$.

Acknowledgement

We thank Ulrike Sattler for providing us with the idea to combine hybrid ↓ logics with multi-modal logics, and Volker Weber for comments on this work.

References

[1] Areces, C., P. Blackburn and M. Marx, *A road-map on complexity for hybrid logics*, in: *Proc. of the 13th CSL, 1999*, LNCS **1683** (1999), pp. 307–321.

[2] Areces, C., P. Blackburn and M. Marx, *The computational complexity of hybrid temporal logics*, Logic Journal of the IGPL **8** (2000), pp. 653–679.

[3] Berger, R., *The undecidability of the domino problem*, Memoirs of the American Mathematical Society **66** (1966), pp. 1–72.

[4] Bezhanishvili, N. and B. ten Cate, *Transfer results for hybrid logic. Part 1: the case without satisfaction operators*, Journal of Logic and Computation **16** (2006), pp. 177–197.

[5] Blackburn, P., *Representation, reasoning, and relational structures: a hybrid logic manifesto*, Logic Journal of the IGPL **8** (2000), pp. 339–365.

[6] Blackburn, P. and J. Seligman, *Hybrid languages*, Journal of Logic, Language and Information **4** (1995), pp. 251–272.

[7] Franceschet, M., M. de Rijke and B.-H. Schlingloff, *Hybrid logics on linear structures: Expressivity and complexity*, in: *Proc. of the 10th TIME, 2003* (2003), pp. 166–173.

[8] Ghilardi, S. and L. Santocanale, *Algebraic and model theoretic techniques for fusion decidability in modal logics.*, in: M. Y. Vardi and A. Voronkov, editors, *LPAR*, Lecture Notes in Computer Science **2850** (2003), pp. 152–166.

[9] Grädel, E., *Why are modal logics so robustly decidable?*, Bulletin of the EATCS **68** (1999), pp. 90–103.

[10] Halpern, J. Y. and Y. Moses, *A guide to completeness and complexity for modal logics of knowledge and belief*, Artificial Intelligence **54** (1992), pp. 319–379.

[11] Ladner, R. E., *The computational complexity of provability in systems of modal propositional logic*, SIAM Journal on Computing **6** (1977), pp. 467–480.

[12] Mundhenk, M. and T. Schneider, *Undecidability of multi-modal hybrid logics*, in: *International Workshop on Hybrid Logic (HyLo 2006)*, 2006, accepted for publication.

[13] Mundhenk, M., T. Schneider, T. Schwentick and V. Weber, *Complexity of hybrid logics over transitive frames*, in: H. Schlingloff, editor, *M4M*, Informatik-Berichte **194** (2005), pp. 62–78.

[14] Ono, H. and A. Nakamura, *On the size of refutation Kripke models for some linear modal and tense logics*, Studia Logica **34** (1980), pp. 325–333.

[15] Papadimitriou, C. H., "Computational Complexity," Addison-Wesley, 1994.

[16] Spaan, E., "Complexity of modal logics," Ph.D. thesis, ILLC, University of Amsterdam (1993).

[17] ten Cate, B. and M. Franceschet, *On the complexity of hybrid logics with binders*, in: *Proc. of the 19th CSL, 2005*, LNCS **3634** (2005), pp. 339–354.

[18] van Emde Boas, P., *The convenience of tilings*, in: A. Sorbi, editor, *Complexity, Logic, and Recursion Theory*, Lecture Notes in Pure and Applied Mathematics, Marcel Dekker Inc., 1997 pp. 331–363.

[19] Wang, H., *Proving theorems by pattern recognition II*, Bell Systems Technical Journal **40** (1961), pp. 1–42.