On modal products with the logic of 'elsewhere'

Christopher Hampson

Department of Informatics
King's College London
Strand, London, WC2R 2LS, U.K.

Agi Kurucz

Department of Informatics
King's College London
Strand, London, WC2R 2LS, U.K.

Abstract

The finitely axiomatisable and decidable modal logic **Diff** of 'elsewhere' (or 'difference operator') is known to be quite similar to **S5**. Their validity problems have the same coNP complexity, and their Kripke frames have similar structures: equivalence relations for **S5**, and 'almost' equivalence relations, with the possibility of some irreflexive points, for **Diff**. However, their behaviour may differ dramatically as components of two-dimensional logics. Here we consider the decision problems of modal product logics of the form $L \times \mathbf{Diff}$. We present some cases where the transition from $L \times \mathbf{S5}$ to $L \times \mathbf{Diff}$ not only increases the complexity of the validity problem, but in fact introduces undecidability, sometimes even non-recursive enumerability.

Keywords: difference operator, products of modal logics, decision problems

1 Introduction

Von Wright's 'logic of elsewhere' [24] is the set **Diff** of propositional modal formulas that are valid in all *difference frames*, that is to say, relational structures $\mathfrak{F} = (W, R)$ where for all $u, v \in W$, uRv iff $u \neq v$. Segerberg [20] gives a complete axiomatisation of **Diff**: He shows that **Diff** is the smallest set of modal formulas (having $\Box$ and $\Diamond$ as modal operators) that is closed under the rules of Substitution, Modus Ponens and Necessitation $\varphi/\Box\varphi$, and contains all propositional tautologies and the formulas

$$\Box(p \to q) \to (\Box p \to \Box q)$$
$$p \to \Box\Diamond p$$
$$\Diamond\Diamond p \to (p \vee \Diamond p)$$

So an arbitrary Kripke frame for **Diff** may contain both reflexive and irreflexive points, but it is always symmetric and *pseudo-transitive*:

$$\forall x, y, z\ \big(R(x, y) \wedge R(y, z) \rightarrow (x = z \vee R(x, z))\big). \tag{1}$$

Note that it is not hard to see [3] that

$$\text{every rooted frame for } \mathbf{Diff} \text{ is a p-morphic image of a difference frame.} \tag{2}$$

One can express the *universal modality* and the '*precisely one*' modality with the help of a difference diamond:

$$\forall\psi = \psi \wedge \neg\Diamond\neg\psi, \qquad\qquad \Diamond^{=1}\psi = (\psi \vee \Diamond\psi) \wedge \neg\Diamond(\psi \wedge \Diamond\psi).$$

Then, for any model $\mathfrak{M}$ over some difference frame (W, R), and any $x \in W$,

$$\begin{array}{rcl} \mathfrak{M}, x \models \forall\psi & \text{iff} & \mathfrak{M}, y \models \psi, \text{ for all } y \in W, \\ \mathfrak{M}, x \models \Diamond^{=1}\psi & \text{iff} & |\{y \in W : \mathfrak{M}, y \models \psi\}| = 1. \end{array}$$

In this paper we take the first steps in investigating decision problems of two-dimensional product logics with **Diff**. We find them intriguing because of the following reason. It is known that in general the existence of a polynomial reduction of a logic L_1 to a logic L_2 does not imply that $L_1 \times L$ is polynomially reducible to $L_2 \times L$ (see e.g. [5, Remark 6.19]). However, in cases when there exist so called 'model level' reductions between L_1 and L_2, such reductions may be 'lifted' to the products (see Sections 2.8, 6.3 and 6.5 in [5]). Both **Diff** and the well-known modal logic **S5** of equivalence relations not only share the same coNP-complete validity problems [3,13], but in fact their frames closely resemble one another. So one might have hoped for such a 'liftable' reduction. However, here we present some cases where the transition from $L \times \mathbf{S5}$ to $L \times \mathbf{Diff}$ not only increases the complexity of the validity problem, but in fact introduces undecidability, sometimes even non-recursive enumerability.

The product construction as a combination method on modal logics was introduced in [19,21,6], and has been extensively studied ever since. Modal products are connected to several other multi-dimensional logical formalisms, see [5,12] for surveys and references. Here we discuss the following special case of the general construction: Given a bimodal frame $\mathfrak{F}_h = (W_h, R_h^1, R_h^2)$ and a unimodal frame $\mathfrak{F}_v = (W_v, R_v)$, their *product* is defined to be the 3-modal frame

$$\mathfrak{F}_h \times \mathfrak{F}_v = (W_h \times W_v, \bar{R}_h^1, \bar{R}_h^2, \bar{R}_v),$$

where $W_h \times W_v$ is the Cartesian product of W_h and W_v and, for all $x, x' \in W_h$, $y, y' \in W_v$, $i = 1, 2$,

$$\begin{array}{rcl} (x, y)\bar{R}_h^i(x', y') & \text{iff} & xR_h^i x' \text{ and } y = y', \\ (x, y)\bar{R}_v(x', y') & \text{iff} & yR_v y' \text{ and } x = x'. \end{array}$$

Frames of this form will be called *product frames* throughout. Now let L_h be a Kripke complete bimodal logic in the language with boxes $\Box_h^1$, $\Box_h^2$ and diamonds $\Diamond_h^1$, $\Diamond_h^2$. Let L_v be a Kripke complete unimodal logic in the language with box $\Box_v$ and diamond $\Diamond_v$. Their product $L_h \times L_v$ is then the set of all 3-modal formulas, in the language having $\Box_h^1$, $\Box_h^2$, $\Box_v$ and $\Diamond_h^1$, $\Diamond_h^2$, $\Diamond_v$, that are valid in all product frames $\mathfrak{F}_h \times \mathfrak{F}_v$, where $\mathfrak{F}_h$ is a frame for L_h, and $\mathfrak{F}_v$ is a frame for L_v. (Here we assume that $\Box_h^i$ and $\Diamond_h^i$ are interpreted by $\bar{R}_h^i$, while $\Box_v$ and $\Diamond_v$ are interpreted by $\bar{R}_v$.) It is easy to see that in fact it is enough to consider *rooted* frames for both component logics [5, Prop.3.7]:

$$\begin{aligned} L_h \times L_v = \{\varphi : \ & \varphi \text{ is valid in every } \mathfrak{F}_h \times \mathfrak{F}_v, \\ & \text{where } \mathfrak{F}_h \text{ is a rooted frame for } L_h, \\ & \text{and } \mathfrak{F}_v \text{ is a rooted frame for } L_v.\} \end{aligned} \tag{3}$$

Our notation and terminology is mostly standard. However, we assume that the reader is familiar with basic notions of propositional multi-modal logic and its possible world (or relational) semantics, and we use these without explicit references. For concepts and statements not defined or proved here, consult, for example, [1,2].

2 Results

In this section we illustrate that the transition from $L \times \mathbf{S5}$ to $L \times \mathbf{Diff}$ might introduce undecidability, and, in some cases, even non-recursively enumerability.

Our first example of such an L is the logic $\mathbf{K}_u$, the bimodal logic of all Kripke frames of the form $\mathfrak{F} = (W, R, W \times W)$, that is, the first relation is arbitrary, and the second is the universal relation on W. By a standard unravelling argument, it can be shown that an arbitrary rooted frame for $\mathbf{K}_u$ is always a p-morphic image of some frame $(W, R, W \times W)$, where (W, R) is a disjoint union of irreflexive, intransitive trees. As the product construction on frames commutes with taking p-morphic images [5, Prop.3.10], by (2) and (3) we obtain that

$\mathbf{K}_u \times \mathbf{Diff}$ is determined by product frames $\mathfrak{F}_h \times \mathfrak{F}_v$, where $\mathfrak{F}_h = (W_h, R_h, W_h \times W_h)$ is such that (W_h, R_h) is a disjoint union of irreflexive, intransitive trees, and $\mathfrak{F}_v = (W_v, R_v)$ is a difference frame. (4)

The validity problem of $\mathbf{K}_u$ is EXPTIME-complete [23,10]. The following theorem is in contrast with the decidability of $\mathbf{K}_u \times \mathbf{S5}$ (see [25] and [5, Thm.6.58]):

Theorem 2.1 $\mathbf{K}_u \times \mathbf{Diff}$ *is undecidable.*

Proof. We reduce the undecidable *non-halting problem* for two-counter Minsky machines [15] to the $\mathbf{K}_u \times \mathbf{Diff}$-satisfiability problem.

A *two-counter Minsky machine* is a finite sequence of instructions $M = (I_0, \dots, I_T)$, where each I_t, for $t < T$, is from the set

$$\{\mathsf{zero}_i, \mathsf{inc}_i, \mathsf{dec}_i(j) : i = 0, 1,\ j \le T\},$$

and $I_T = \mathsf{halt}$. A *configuration* of M is a triple (k, ℓ, m) of natural numbers, with k being the index of the current instruction, and ℓ, m the current contents of the two registers. The (unique) *computation* of M is the function $f_M : \omega \to (\omega \times \omega \times \omega)$ defined by taking $f_M(0) = (0, 0, 0)$, and if $f_M(n) = (k, \ell, m)$ then $k \leq T$ and

$$f_M(n+1) = \begin{cases} (k+1, 0, m), & \text{if } I_k = \mathsf{zero}_0, \\ (k+1, \ell, 0), & \text{if } I_k = \mathsf{zero}_1, \\ (k+1, \ell+1, m), & \text{if } I_k = \mathsf{inc}_0, \\ (k+1, \ell, m+1), & \text{if } I_k = \mathsf{inc}_1, \\ (k+1, \ell-1, m), & \text{if } I_k = \mathsf{dec}_0(j) \text{ and } \ell > 0, \\ (k+1, \ell, m-1), & \text{if } I_k = \mathsf{dec}_1(j) \text{ and } m > 0, \\ (j, 0, m), & \text{if } I_k = \mathsf{dec}_0(j) \text{ and } \ell = 0, \\ (j, \ell, 0), & \text{if } I_k = \mathsf{dec}_1(j) \text{ and } m = 0, \\ (k, \ell, m) & \text{if } I_k = \mathsf{halt}. \end{cases}$$

We write $f_M(n) = \big(\mathsf{i}^M(n), \mathsf{c}_0^M(n), \mathsf{c}_1^M(n)\big)$ to indicate the role of the numbers in the configurations. We define the halting number H_M of M as

$$H_M = \begin{cases} n+1, & \text{if } n \text{ is the smallest number with } I_{\mathsf{i}^M(n)} = \mathsf{halt}, \\ \omega, & \text{if there is no } n < \omega \text{ with } I_{\mathsf{i}^M(n)} = \mathsf{halt}. \end{cases}$$

We say that M *halts* if and only if $H_M < \omega$.

Now, given a Minsky machine M as above, we will define a 3-modal formula φ_M, whose length is recursive (in fact, linear) in T. We will use the language having $\Diamond_h, \Box_h$ for the 'horizontal' **K**-modalities, $\forall_h$ for the 'horizontal' universal modality, and $\Diamond_v$ for the 'vertical' difference operator. We will also use the following 'vertical' abbreviations:

$$\begin{aligned} \exists_v \psi &= \psi \vee \Diamond_v \psi, \\ \forall_v \psi &= \neg \exists_v \neg \psi, \\ \Diamond_v^{=1} \psi &= (\psi \vee \Diamond_v \psi) \wedge \neg \Diamond_v (\psi \wedge \Diamond_v \psi). \end{aligned}$$

The idea is to encode the configuration of M as M evolves over time; with each **K**-succession representing one time-step in the computation of M. We take two propositional variables c_0 and c_1 that will emulate the counters in each of the two registers of M: the number of points in each vertical **Diff**-cluster satisfying c_i will represent the contents of the ith register.

We introduce the following abbreviations, for $i = 0, 1$, that will dictate how the counters in each register are manipulated:

$$\begin{aligned} \psi_{\mathsf{inc}}(i) &= \Diamond_v^{=1} (\neg c_i \wedge \Diamond_h c_i) \wedge \forall_v (c_i \to \Box_h c_i) \\ \psi_{\mathsf{dec}}(i) &= \Diamond_v^{=1} (c_i \wedge \Diamond_h \neg c_i) \wedge \forall_v (\neg c_i \to \Box_h \neg c_i) \\ \psi_{\mathsf{fix}}(i) &= \forall_v (c_i \leftrightarrow \Diamond_h c_i) \\ \psi_{\mathsf{zero}}(i) &= \forall_v \Box_h \neg c_i \end{aligned}$$

For example, $\psi_{\mathsf{inc}}(i)$ stipulates that there is *exactly one* vertically accessible point that evolves from satisfying $\neg c_i$ to satisfying c_i, while every vertically accessible point that satisfies c_i remains satisfying c_i in its **K**-successors; hence the number of points satisfying c_i is incremented by exactly one from one vertical **Diff**-cluster to the next.

We take a propositional variable s_t, for each $t \leq T$, to encode the internal state of M as M evolves over time, and define φ_M to be the conjunction of the following formulas:

$$s_0 \wedge \forall_v \neg c_0 \wedge \forall_v \neg c_1 \tag{5}$$

$$\forall_h \bigvee_{t \leq T} s_t \wedge \bigwedge_{t \neq t' \leq T} \forall_h \neg(s_t \wedge s_{t'}) \tag{6}$$

$$\bigwedge_{t \leq T} \forall_h(\Diamond_h s_t \to \Box_h s_t) \tag{7}$$

$$\bigwedge_{i=0,1} \forall_h \forall_v(\Diamond_h c_i \to \Box_h c_i) \tag{8}$$

$$\bigwedge_{\substack{t<T,\, i=0,1 \\ I_t=\mathsf{zero}_i}} \forall_h\big(s_t \to \Diamond_h s_{t+1} \wedge \psi_{\mathsf{zero}}(i) \wedge \psi_{\mathsf{fix}}(1-i)\big) \tag{9}$$

$$\bigwedge_{\substack{t<T,\, i=0,1 \\ I_t=\mathsf{inc}_i}} \forall_h\big(s_t \to \Diamond_h s_{t+1} \wedge \psi_{\mathsf{inc}}(i) \wedge \psi_{\mathsf{fix}}(1-i)\big) \tag{10}$$

$$\bigwedge_{\substack{t<T,\, i=0,1 \\ I_t=\mathsf{dec}_i(j)}} \forall_h\big((s_t \wedge \exists_v c_i) \to \Diamond_h s_{t+1} \wedge \psi_{\mathsf{dec}}(i) \wedge \psi_{\mathsf{fix}}(1-i)\big) \tag{11}$$

$$\bigwedge_{\substack{t<T,\, i=0,1 \\ I_t=\mathsf{dec}_i(j)}} \forall_h\big((s_t \wedge \forall_v \neg c_i) \to \Diamond_h s_j \wedge \psi_{\mathsf{fix}}(0) \wedge \psi_{\mathsf{fix}}(1)\big) \tag{12}$$

The first formula (5) encodes the initial configuration of M, while (6) stipulates that every point horizontally accessible from the root must always satisfy exactly one state variable s_t, for $t \leq T$. Formulas (7) and (8) ensure that any two distinct **K**-chains encode the same sequence of configurations. The remaining formulas (9)–(12) specify the behaviour of the machine, depending on the sequence of instructions set down by M.

Now suppose that φ_M is $\mathbf{K}_u \times \mathbf{Diff}$-satisfiable. By (4), we may assume that $\mathfrak{M}, (x_0, y_0) \models \varphi_M$ for some model $\mathfrak{M}$ based on a product frame $\mathfrak{F}_h \times \mathfrak{F}_v$, where $\mathfrak{F}_h = (W_h, R_h, W_h \times W_h)$ is such that (W_h, R_h) is a disjoint union of irreflexive, intransitive trees, and $\mathfrak{F}_v = (W_v, R_v)$ is a difference frame. The following notion is then well-defined, for every $x \in W_h$:

$$d(x_0, x) = \begin{cases} 0, & \text{if } x = x_0, \\ n, & \text{if there exist } w_0, \ldots, w_n \text{ with } x_0 = w_0 R_h \ldots R_h w_n = x, \\ \omega, & \text{otherwise.} \end{cases}$$

Now we have the following claim (where we use $(x, y) \models \psi$ as a shorthand for $\mathfrak{M}, (x, y) \models \psi$):

CLAIM **2.1.1** *If $(x_0, y_0) \models \varphi_M$, then for all $n < H_M$, for all $x \in W_h$ with $d(x_0, x) = n$, and for all $t \leq T$, the following hold:*

(i) $(x, y_0) \models s_t$ *if and only if* $t = \mathsf{i}^M(n)$,

(ii) $|\{y \in W_v : (x, y) \models c_0\}| = \mathsf{c}_0^M(n)$,

(iii) $|\{y \in W_v : (x, y) \models c_1\}| = \mathsf{c}_1^M(n)$.

(Here $|U|$ denotes the cardinality of set U.)

Proof. We prove this by induction on n. If $n = 0$, then the statements hold by (5) and (6). Now suppose for induction that the statements (i), (ii) and (iii) hold for all $x' \in W_h$ with $d(x_0, x') = n$, and let $x \in W_h$ be such that $d(x_0, x) = n + 1$. Then there is a unique $x' \in W_h$ such that $d(x_0, x') = n$ and $x' R_h x$. Let $t = \mathsf{i}^M(n)$. Then, by the IH, $(x', y_0) \models s_t$. There are six cases, depending on the form of I_t. Let us consider two examples:

- $I_t = \mathsf{zero}_0$:
  Then $\mathsf{i}^M(n+1) = \mathsf{i}^M(n) + 1 = t + 1$, $\mathsf{c}_0^M(n+1) = 0$, and $\mathsf{c}_1^M(n+1) = \mathsf{c}_1^M(n)$. By (9), we have $(x', y_0) \models \Diamond_h s_{t+1}$, and so by (6) and (7), (i) holds. Also by (9), we have $(x', y_0) \models \psi_{\mathsf{zero}}(0)$, and so $(x, y) \models \neg c_0$, for all $y \in W_v$, as required in (ii). Further, by (9), we have $(x', y_0) \models \psi_{\mathsf{fix}}(1)$, and so by (8), for all $y \in W_v$, $(x', y) \models c_1$ if and only if $(x, y) \models c_1$, as required in (iii).
- $I_t = \mathsf{dec}_1(j)$:
  Suppose first that $\mathsf{c}_1^M(n) > 0$. Then $(x', y_0) \models \exists_v c_1$, by the IH, and we have $\mathsf{i}^M(n+1) = \mathsf{i}^M(n) + 1 = t + 1$, $\mathsf{c}_0^M(n+1) = \mathsf{c}_0^M(n)$, and $\mathsf{c}_1^M(n+1) = \mathsf{c}_1^M(n) - 1$. By (11), we have $(x', y_0) \models \Diamond_h s_{t+1}$, and so by (6) and (7), (i) holds. Also by (11), we have $(x', y_0) \models \psi_{\mathsf{dec}}(1)$. Therefore, there is $y^* \in W_v$ with $(x', y^*) \models c_1 \wedge \Diamond_h \neg c_1$, and for all $y \in W_v$, $y \neq y^*$, $(x', y) \models c_1$ if and only if $(x, y) \models c_1$. By (8), we also have $(x, y^*) \models \neg c_1$, as required in (iii). Further, by (11), we have $(x', y_0) \models \psi_{\mathsf{fix}}(0)$, and so by (8), for all $y \in W_v$, $(x', y) \models c_0$ if and only if $(x, y) \models c_0$, as required in (ii).

  Now suppose that $\mathsf{c}_1^M(n) = 0$. Then $(x', y_0) \models \forall_v \neg c_1$, by the IH, and we have $\mathsf{i}^M(n+1) = j$, $\mathsf{c}_0^M(n+1) = \mathsf{c}_0^M(n)$, and $\mathsf{c}_1^M(n+1) = \mathsf{c}_1^M(n) = 0$. By (12), we have $(x', y_0) \models \Diamond_h s_j$, and so by (6) and (7), (i) holds. Also by (12), we have $(x', y_0) \models \psi_{\mathsf{fix}}(i)$, for $i = 0, 1$, and so by (8), for all $y \in W_v$, $(x', y) \models c_i$ if and only if $(x, y) \models c_i$, as required in (ii) and (iii).

The other cases are similar and are left to the reader. $\square$

As a consequence of Claim 2.1.1, we obtain the following:

$$\text{if } \varphi_M \wedge \forall_h \neg s_T \text{ is } \mathbf{K}_u \times \mathbf{Diff}\text{-satisfiable, then } M \text{ does not halt.} \tag{13}$$

On the other hand, we also have that

$$\text{if } M \text{ does not halt, then } \varphi_M \wedge \forall_h \neg s_T \text{ is } \mathbf{K}_u \times \mathbf{Diff}\text{-satisfiable.} \tag{14}$$

Indeed, suppose that M does not halt. Let $\mathfrak{F}_h = (\omega, +1, \omega \times \omega)$, and let $\mathfrak{F}_v$ be the difference frame on ω. We define a model $\mathfrak{M}$ on $\mathfrak{F}_h \times \mathfrak{F}_v$ by taking, for all

$n, m < \omega$, $t \leq T$, and $i = 0, 1$,

$$\mathfrak{M}, (n,m) \models s_t \quad \text{iff} \quad \mathsf{i}^M(n) = t \text{ and } m = 0,$$
$$\mathfrak{M}, (n,m) \models c_i \quad \text{iff} \quad m < \mathsf{c}_i^M(n).$$

It is then straightforward to check that $\mathfrak{M}, (0,0) \models \varphi_M \wedge \forall_h \neg s_T$. The theorem now follows from (13) and (14). $\Box$

Note that the class of all frames, for each of $\mathbf{K}_u$ and **Diff**, can be defined by a recursive set of first-order sentences in the frame-correspondence language. Therefore, the product logic $\mathbf{K}_u \times \mathbf{Diff}$ is *recursively enumerable* [6]. So Theorem 2.1 implies that $\mathbf{K}_u \times \mathbf{Diff}$ *lacks the effective* (or *bounded*) *finite model property:* The size of a frame necessary to falsify any given formula φ that does not belong to $\mathbf{K}_u \times \mathbf{Diff}$ cannot be bound by a function recursive in the length of φ. However, as $\mathbf{K}_u \times \mathbf{Diff}$ is not finitely axiomatisable [9], in principle it can happen that we cannot enumerate the finite frames for $\mathbf{K}_u \times \mathbf{Diff}$, and so $\mathbf{K}_u \times \mathbf{Diff}$ might have the (abstract) finite model property. It is easy to see that it does not have the finite model property w.r.t. product frames: For example, take the formula φ_M defined in the proof above for the two-counter Minsky machine $M = (\mathsf{inc}_0, \mathsf{dec}_1(0), \mathsf{halt})$.

Next, instead of frames with a universal modality as first components, we consider frames of the form (W, R, R^*), where (W, R) is an irreflexive, intransitive tree, and R^* is the *reflexive and transitive closure* of R. The modal operator corresponding to R^* is sometimes called *master modality*, or *common knowledge operator* in epistemic logics. Examples of logics determined by frames of this kind are

- $\mathbf{K}_C$, the bimodal logic of all frames of the form (W, R, R^*),
- $\mathbf{PTL}_{X\Box}$, the 'next-time, future' fragment of Propositional Temporal Logic over $(\omega, +1, \leq)$ as time-line, and
- $\mathbf{PDL}_1^-$, test-free Propositional Dynamic Logic with just one atomic program and its Kleene star closure.

Both $\mathbf{K}_C$ and $\mathbf{PDL}_1^-$ have the same EXPTIME-completeness as $\mathbf{K}_u$ [8,4,16], while $\mathbf{PTL}_{X\Box}$ is PSPACE-complete [22]. Furthermore, each of these logics (and, indeed, $\mathbf{K}_u$) are polynomially reducible to full **PDL** using 'model level' reductions that can be 'lifted' to products, see [5, Sections 6.3,6.5]. As is shown in [5, Thm.6.49], the validity problem of $\mathbf{PDL} \times \mathbf{S5}$ is decidable in CON2EXPTIME, and so $\mathbf{K}_C \times \mathbf{S5}$, $\mathbf{PTL}_{X\Box} \times \mathbf{S5}$, and $\mathbf{PDL}_1^- \times \mathbf{S5}$ are also decidable. (Note that all these logics are EXPSPACE-hard, and $\mathbf{PTL}_{X\Box} \times \mathbf{S5}$ is in fact EXPSPACE-complete, see [5, Thms.6.65,6.66].)

Theorem 2.2 *Let $\mathcal{C}$ be any class of frames such that*

- *every frame in $\mathcal{C}$ is of the form $\mathfrak{F}_h \times \mathfrak{F}_v$, where $\mathfrak{F}_h = (W, R, R^*)$ with (W, R) being an irreflexive, intransitive tree, and $\mathfrak{F}_v$ is a difference frame;*
- $(\omega, +1, \leq) \in \mathcal{C}$.

Let L be the set of all 3-modal formulas that are valid in all frames in $\mathcal{C}$. Then

L is not recursively enumerable.

Proof. We reduce the undecidable but recursively enumerable *halting problem* for two-counter Minsky machines to the L-satisfiability problem.

Let M be a Minsky machine, and let φ_M be the formula defined in the proof of Theorem 2.1. It is straightforward to see that $\varphi_M \land \neg\forall_h \neg s_T$ is satisfiable in a frame in $\mathcal{C}$ if and only if M halts. (Here the notation $\forall_h$ is a bit misleading, as the corresponding relation is not 'horizontally' universal any more, but the reflexive and transitive closure of the relation corresponding to $\Box_h$.) $\Box$

Corollary 2.3 *The logics $\mathbf{K}_C \times \mathbf{Diff}$, $\mathbf{PTL}_{X\Box} \times \mathbf{Diff}$, and $\mathbf{PDL}_1^- \times \mathbf{Diff}$ are not recursively enumerable.*

3 Discussion

We conclude the paper with a few remarks on related formalisms and further research.

- As **Diff** can be regarded as a fragment of hybrid logic, our Theorems 2.1 and 2.2 imply the undecidability of some hybrid product logics (see [18]).
- We gave examples of bimodal logics L where $L \times \mathbf{Diff}$ is undecidable. Products of standard decidable unimodal logics and **Diff** have not been investigated. Is there an example for undecidability among them? In particular, is any of $\mathbf{K} \times \mathbf{Diff}$, $\mathbf{K4} \times \mathbf{Diff}$, $\mathbf{S5} \times \mathbf{Diff}$, or $\mathbf{Diff} \times \mathbf{Diff}$ decidable? Each of these products with **S5** in place of **Diff** is known to be decidable, logics like $\mathbf{K} \times \mathbf{S5}$ and $\mathbf{S5} \times \mathbf{S5}$ are even coNEXPTIME-complete. Even if these kinds of products with **Diff** turn out to be decidable, we cannot always hope for proofs that are completely analogous to the **S5**-cases: in contrast to the product finite model property of, say, $\mathbf{K} \times \mathbf{S5}$ or $\mathbf{S5} \times \mathbf{S5}$, it turns out that $\mathbf{Diff} \times \mathbf{Diff}$ has no (abstract) finite model property [9]. Concerning attempts at filtration arguments, note that no logic of the form $L \times \mathbf{Diff}$ is finitely axiomatisable, whenever L is between **K** and **S5** [9].
- We proved Theorems 2.1 and 2.2 using reductions of the halting problem for two-counter Minsky machines. It appears that this technique is slightly different to other proofs of undecidability results about product logics, which use reductions of the halting problem for Turing machines or $\omega \times \omega$-tilings. One might think that these latter undecidable problems are tailor-made for product logics: product frames are by definition grid-like, so it should not be hard to encode these 'grid-based' problems into them. Indeed, if we have both next-time and universal or master modalities in both dimensions, then this is rather straightforward (see, for example, the case of $\mathbf{K}_u \times \mathbf{K}_u$ in [5, Thm.5.37]). However, if some of this machinery is missing, then often the grid needs to be encoded by 'diagonal' points, and some other tricks may be needed [7,14,17,11]. We failed to apply these kinds of tricks in the undecidability proofs given here. In order to understand the boundaries of each technique, it would be interesting to know whether there is a natural way to fully encode the $\omega \times \omega$-grid in frames for $L \times \mathbf{Diff}$-logics.

References

[1] Blackburn, P., M. de Rijke and Y. Venema, "Modal Logic," Cambridge University Press, 2001.

[2] Chagrov, A. and M. Zakharyaschev, "Modal Logic," Oxford Logic Guides **35**, Clarendon Press, Oxford, 1997.

[3] de Rijke, M., *The modal logic of inequality*, Journal of Symbolic Logic **57** (1992), pp. 566–584.

[4] Fischer, M. and R. Ladner, *Propositional dynamic logic of regular programs*, Journal of Computer and System Sciences **18** (1979), pp. 194–211.

[5] Gabbay, D., A. Kurucz, F. Wolter and M. Zakharyaschev, "Many-Dimensional Modal Logics: Theory and Applications," Studies in Logic and the Foundations of Mathematics **148**, Elsevier, 2003.

[6] Gabbay, D. and V. Shehtman, *Products of modal logics. Part I*, Journal of the IGPL **6** (1998), pp. 73–146.

[7] Gabelaia, D., A. Kurucz, F. Wolter and M. Zakharyaschev, *Products of 'transitive' modal logics*, Journal of Symbolic Logic **70** (2005), pp. 993–1021.

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

[9] Hampson, C. and A. Kurucz, *Axiomatisation and decision problems of modal product logics with the difference operator* (2012), (manuscript).

[10] Hemaspaandra, E., *The price of universality*, Notre Dame Journal of Formal Logic **37** (1996), pp. 174–203.

[11] Kikot, S. and A. Kurucz, *Undecidable two dimensional modal product logics with diagonal constant* (2012), (submitted).

[12] Kurucz, A., *Combining modal logics*, in: P. Blackburn, J. van Benthem and F. Wolter, editors, *Handbook of Modal Logic*, Studies in Logic and Practical Reasoning **3**, Elsevier, 2007 pp. 869–924.

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

[14] Marx, M. and M. Reynolds, *Undecidability of compass logic*, Journal of Logic and Computation **9** (1999), pp. 897–914.

[15] Minsky, M., "Finite and infinite machines," Prentice-Hall, 1967.

[16] Pratt, V., *Models of program logics*, in: *Proceedings of the 20th IEEE Symposium on Foundations of Computer Science*, 1979, pp. 115–122.

[17] Reynolds, M. and M. Zakharyaschev, *On the products of linear modal logics*, Journal of Logic and Computation **11** (2001), pp. 909–931.

[18] Sano, K., *Axiomatizing hybrid products: How can we reason many-dimensionally in hybrid logic?*, Journal of Applied Logic **8** (2010), pp. 459–474.

[19] Segerberg, K., *Two-dimensional modal logic*, Journal of Philosophical Logic **2** (1973), pp. 77–96.

[20] Segerberg, K., *A note on the logic of elsewhere*, Theoria **46** (1980), pp. 183–187.

[21] Shehtman, V., *Two-dimensional modal logics*, Mathematical Notices of the USSR Academy of Sciences **23** (1978), pp. 417–424, (Translated from Russian).

[22] Sistla, A. and E. Clarke, *The complexity of propositional linear temporal logics*, Journal of the Association for Computing Machinery **32** (1985), pp. 733–749.

[23] Spaan, E., "Complexity of Modal Logics," Ph.D. thesis, Department of Mathematics and Computer Science, University of Amsterdam (1993).

[24] von Wright, G., *A modal logic of place*, in: E. Sosa, editor, *The philosophy of Nicolas Rescher*, Dordrecht, 1979 pp. 65–73.

[25] Wolter, F. and M. Zakharyaschev, *Modal description logics: modalizing roles*, Fundamenta Informaticae **39** (1999), pp. 411–438.