Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science

Electronic Notes in Theoretical Computer Science 273 (2011) 3–19

www.elsevier.com/locate/entcs

Hybrid Metric Propositional Neighborhood Logics with Interval Length Binders

Della Monica, Dario[1]

University of Udine, Udine, Italy, and
University of Salerno, Salerno, Italy

Goranko, Valentin[2,5]

Technical University of Denmark, Lyngby, Denmark

Sciavicco, Guido[3,4]

University of Murcia, Murcia, Spain, and
University for Information Science and Technology, Ohrid, Macedonia

Abstract

We investigate the question of how much hybrid machinery can be added to the interval neighbourhood logic PNL and its metric extension MPNL without losing the decidability of their satisfiability problem in $\mathbb{N}$. In particular, we consider the natural hybrid extension of MPNL obtained by adding binders on integer variables ranging over lengths of intervals, thus enabling storage of the length of the current interval and further references to it. We show that even a very weak natural fragment of such extensions becomes undecidable, which is somewhat surprising, being in contrast with the decidability of MPNL, which can be seen as a hybrid language with length constraints only involving constants over interval lengths. These results show that MPNL itself is, in this sense, a maximal decidable (weakly) hybrid extension of PNL.

Keywords: hybrid logics, interval neighbourhood logics, interval length binders, undecidability, tiling.

[1] Email: dario.dellamonica@dimi.uniud.it

[2] Email: vfgo@imm.dtu.dk

[3] Email: guido@um.es

[4] This author has been partially supported by the Spanish/South-African Project HS2008-0006, and by the Spanish project TIN2009-14372-C03-01.

[5] This author has been supported by the HYLOCORE project funded by the Danish Natural Science Research Council.

doi:10.1016/j.entcs.2011.06.009

1 Introduction

1.1 Background

Interval temporal logics are based on temporal structures over (usually) linearly ordered domains, where time intervals, rather than time instants, are the primitive ontological entities. The problem of representing and reasoning about time intervals arises naturally in various other fields of computer science, artificial intelligence, and temporal databases, such as theories of action and change, natural language processing, and constraint satisfaction problems. In particular, temporal logics with interval-based semantics have been proposed as a useful formalism for the specification and verification of hardware [22] and of real-time systems [12].

A systematic analysis of the variety of relations between two intervals in a linear order was initiated by Allen [1], who proposed the use of interval reasoning in systems for time management and planning. Allen identified the thirteen different binary relations between intervals on linear orders, hereafter referred to as Allen's relations. In [16], Halpern and Shoham introduced a multi-modal logic, hereafter called HS, involving modal operators corresponding to all Allen's interval relations and showed that such a logic is undecidable under very weak assumptions on the class of interval structures in which it is interpreted. One of the few known cases of decidable interval logics with truly interval semantics (not reducible to point-based semantics) is the Propositional Neighborhood Interval Logic (PNL) [5,15]. PNL is a fragment of HS with only two modal operators, corresponding to Allen's relations *meets* and its inverse *met by*. Its satisfiability problem has been shown to be decidable (NEXPTIME-complete) when interpreted over various classes of linearly ordered sets, in particular, over domains based on natural numbers [6]; the results presented in the same paper and in [21] showed that all possible extensions of PNL with Allen's modal operator make the logic undecidable, which means that PNL is maximal in terms of decidability (as a matter of fact, there are extensions of PNL that are non-elementarly decidable only if interpreted over finite prefixes of $\mathbb{N}$ and undecidable in most of the other cases [21]). In [4,7], authors proposed a 'metric' extension of PNL, called *Metric* PNL (MPNL, for short), which involves special propositional letters expressing equality or inequality constraints on the length of the current interval with respect to fixed integer constants. The satisfiability problem for MPNL interpreted in the interval structure over $\mathbb{N}$ is proved to be decidable in [4], with complexity between EXPSPACE and 2NEXPTIME when the integer constraints in formulae are represented in binary, and with complexity in NEXPTIME-complete when the integer constraints in formulae are constant or represented in unary.

In the present paper we have investigated the question of how much hybrid machinery can be added to PNL and MPNL without losing the decidability of the satisfiability problem in $\mathbb{N}$. Since the difference modality is definable in PNL[6] [15], nominals can be simulated there, so adding them alone is unproblematic with

[6] The definition in the strict semantics is given further; in the non-strict semantics it needs using the modal constant π for point intervals.

regards to decidability. However, it is very easy to see that adding binders over state variables immediately leads to undecidability. On the other hand, in the framework of Metric PNL it is quite natural to use binders not on state variables ranging over intervals, but on integer variables ranging over *lengths of intervals*, thus enabling storage of the length of the current interval and further references to it.

1.2 A brief comparison with Duration Calculus

The length variables and binders introduced in this paper bear natural resemblance with the interval length variables used in Duration Calculus (DC) [11,18] – an interval logic introduced by Zhou Chaochen, C.A.R. Hoare, and A.P. Ravn [12], extending Moszkowski's ITL [22]. The original version of ITL involves only one, binary, modal operator C, called *chop*, where pCq states that the current interval $[a, b]$ can be split (chopped) into two consecutive intervals $[a, c]$ and $[c, b]$ such that $[a, c]$ satisfies p and $[c, b]$ satisfies q. DC is a real-time version of ITL that adds to the language *state expressions*, representing states of the system and assigned a *duration* – the length of the time period during which the system remains in the given state. In [10], a version of DC based on Neighborhood Logic, denoted as DC/NL, has been introduced; it features the two interval neighborhood modalities $\Diamond_r$ and $\Diamond_l$ and subsumes the original DC. Both the interval logic ITL and DC are undecidable over almost all interesting classes of linear orders.

Several versions of ITL and DC have been studied, aiming to obtain decidability for the satisfiability/validity problem. For example, in [23], Pandya considers the interval-based version of DC, called *Interval Duration Logic* (*IDL*), that is generally undecidable; however, it has been proved in [23] that a specific fragment of IDL, denoted by *LIDL*−, is decidable, by using an automata-theoretic argument. The problem of deciding the validity of IDL formulae has been further investigated in [9], where the authors propose a syntactic characterization of a subset of IDL-formulae that share a property called *Strong Closure under Inverse Digitalization* that allows one to effectively check the validity of formulae that belong to such a subset by reducing it to the validity problem of formulae of *Discrete Time Duration Calculus* (*QDDT*), that is, the corresponding discrete-time logic, whose validity problem is decidable [20]. Subsequently, M. Fränzle and M. R. Hansen obtain a decidability result for a rich fragment of DC [14], extending the work of Zhou Chaochen et al. [13] on decidability of linear duration invariants to a much wider fragment of DC. In the Restricted Duration Calculus [17] denoted by RDC$_1$, the length of the current intervals can be referred to by using the equality constraint $\mathsf{len}_{=\mathsf{k}}$ (which can be expressed using C and $\mathsf{len}_{=1}$). This fragment turns out to be decidable over discrete time, but undecidable over dense time. On the other hand, the subset denoted by RDC$_3$, where lengths of current intervals can also be referred to by means of variables and quantifiers, turns out to be undecidable for both discrete and dense time interpretations. On the other hand, under the assumption of *locality* (that is, the truth of formulae over intervals is reduced to the truth of it over the points inside the interval), not only DC but also a hybrid extension of it studied in [2], that allows binders for intervals (not lengths) and nominals referring to specific intervals, is decidable, because the locality assumption essentially reduces the interval logic to a point-based one, and eventually reduces the satisfiability problem over $\mathbb{N}$ to the one for MSO over $\mathbb{N}$.

1.3 Our contributions

The main part of this paper is devoted to analyze extensions of PNL and MPNL with interval length variables and binders with respect to decidability. Eventually, we show that even very weak extensions become undecidable, which in some cases was not obvious at all, and even somewhat surprising, being in sharp contrast with the decidability of MPNL, which can be seen as a hybrid language only involving nominals over interval lengths. These results show that MPNL itself is, in this sense, a maximal decidable (weakly) hybrid extension of PNL. Finally, we also note the contrast between the strongly prevailing undecidability of hybrid interval logics with truly interval-based semantics, demonstrated here, and the much more robust decidability of even very expressive hybrid extensions of interval logics with essentially point-based semantics, incl. the hybrid DC mentioned above.

Structure of the paper: in Section 2 we recall the syntax and semantics of PNL and MPNL; in Section 3 we discuss hybrid extensions of MPNL, and present the *Weakly Hybrid* MPNL; in Section 4 we prove the main undecidability results, and in Section 5 we present some ideas of how to regain decidability, before concluding.

2 Metric Propositional Neighborhood Logic

The language of the Propositional Neighborhood Logic (PNL) consists of a set $\mathcal{AP}$ of atomic propositions, the propositional connectives $\neg$ and $\vee$, and the modal operators $\Diamond_r$ and $\Diamond_l$, corresponding to the Allen's interval relations *meets* and its inverse *met-by* [1]. The other propositional connectives, as well as the logical constants $\top$ (*true*) and $\bot$ (*false*), and the dual modal operators $\Box_r$ and $\Box_l$, are defined as usual. PNL has been studied both in the so-called *strict semantics*, which excludes point-intervals, and in the *non-strict* one which includes them. In the latter case, it is natural to include in the language a special atomic proposition (modal constant), usually denoted by π, that is true over all and only the point-intervals. The expressive powers of the various cases have been studied and compared in [15].

The formulae of PNL, denoted by $\varphi, \psi, \ldots$, are generated by the following grammar:

$$\varphi ::= \pi \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \Diamond_r\varphi \mid \Diamond_l\varphi.$$

Given a linearly ordered domain $\mathbb{D} = \langle D, < \rangle$, a (*non-strict*) *interval* over $\mathbb{D}$ is any ordered pair $[a, b]$ such that $a \leq b$. An *interval structure* is a pair $\langle \mathbb{D}, \mathbb{I}(\mathbb{D}) \rangle$, where $\mathbb{I}(\mathbb{D})$ is the set of all intervals over $\mathbb{D}$. The semantics of PNL is given in terms of models of the form $M = \langle \mathbb{D}, \mathbb{I}(\mathbb{D}), V \rangle$, where $\langle \mathbb{D}, \mathbb{I}(\mathbb{D}) \rangle$ is an interval structure and $V : \mathcal{AP} \to 2^{\mathbb{I}(\mathbb{D})}$ is a valuation function assigning to every propositional letter the set of those intervals over which it is true. Note that no conditions, such as locality, homogeneity, etc. on the valuation are imposed. We recursively define the satisfiability relation $\Vdash$ as follows:

- $M, [a, b] \Vdash \pi$ iff $a = b$;
- $M, [a, b] \Vdash p$ iff $[a, b] \in V(p)$, for any $p \in \mathcal{AP}$;
- $M, [a, b] \Vdash \neg\psi$ iff it is not the case that $M, [a, b] \Vdash \psi$;
- $M, [a, b] \Vdash \psi \vee \tau$ iff $M, [a, b] \Vdash \psi$ or $M, [a, b] \Vdash \tau$;
- $M, [a, b] \Vdash \Diamond_r\psi$ iff there exists $c \geq b$ such that $M, [b, c] \Vdash \psi$;
- $M, [a, b] \Vdash \Diamond_l\psi$ iff there exists $c \leq a$ such that $M, [c, a] \Vdash \psi$.

The satisfiability problem of the various version of PNL has been shown to be decidable in [5,15]. In [8], a tableau-based method has been presented for PNL.

From now on, we only consider PNL and its extensions interpreted in the interval structure on $\mathbb{N}$. We denote by $\delta : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ the *distance* function on $\mathbb{N}$, defined as $\delta(a, b) = |a - b|$. Most of the claims and results in this paper hold not only on $\mathbb{N}$, but also on $\mathbb{Z}$ and many other linear orders on which distance between points is definable. The metric extension of PNL, called MPNL, was introduced and studied in [7,4]. MPNL extends PNL with *atomic propositions for length constraints.* These are pre-interpreted propositional letters referring to the length of the current interval, which can be seen as the metric generalizations of the modal constant π. From now on, let $\mathbb{C} = \{<, \leq, =, \geq, >\}$. For each $\mathcal{C} \in \mathbb{C}$, we introduce the length constraint $\mathsf{len}_{\mathcal{C}\mathsf{k}}$, with the following semantics, where $k \in \mathbb{N}$:

$$M, [a, b] \Vdash \mathsf{len}_{\mathcal{C}\mathsf{k}} \text{ iff } \delta(a, b)\mathcal{C}k.$$

The satisfiability problem for MPNL has been shown in [4] to be decidable; in particular the following theorem holds.

Theorem 2.1 *The complexity of the satisfiability problem for MPNL interpreted over $\mathbb{N}$ is between EXPSPACE and 2NEXPTIME if the length constraints are represented in binary, and NEXPTIME complete when k is a constant or represented in unary.*

3 Hybrid Propositional Neighborhood Logics

Some operators of hybrid flavor can be defined in the language of PNL. For instance, the *universal modality* $[G]$, referring to all intervals in the model, is definable in all variants of PNL. As an example, if we consider the non-strict semantics, it may be defined as follows: $[G]\varphi \equiv \Box_l\Box_l\Box_r\Box_r\varphi$. Moreover, in the strict semantics, as shown in [15], the *difference modality* $[\neq]$ can be defined as:

$$[\neq]\varphi \equiv \Box_l\Box_l\Box_r\varphi \wedge \Box_l\Box_r\Box_r\varphi \wedge \Box_r\Box_l\Box_l\varphi \wedge \Box_r\Box_r\Box_l\varphi.$$

Thus, nominals over intervals can be simulated in PNL, and therefore the *Basic Hybrid extension of PNL* (BHPNL) remains decidable over a large family of linear orders, including $\mathbb{N}$. However, it is quite easy to see that adding stronger hybrid machinery, such as binders or quantifiers over intervals, immediately leads to undecidability even under very week assumptions about the class of linear orders.

On the other hand, in the framework of MPNL it is quite natural to use binders not on state variables ranging over intervals, but on integer variables ranging over *lengths of intervals.* In its "classical" version, MPNL allows metric constraints expressed by explicit numbers; for example, $\Diamond_r(\mathsf{len}_{=5} \wedge p \rightarrow \Diamond_l \Diamond_r q)$ is a well-formed MPNL formula, while $\Diamond_r(\mathsf{len}_{=\mathsf{x}} \wedge p)$ is not. This means that, despite the fact that MPNL can be considered very expressive (as shown by several examples in [4]), there are simple and natural properties that we are not able to express in this language, such as e.g.: *the right neighbor interval with length equal to the length of the current interval satisfies the property q.* Thus, it is natural to extend the language of MPNL with a sort of hybrid machinery which allows one to store the length of the current interval and to use it further in formulae.

Here we introduce such a hybrid extension of MPNL, that we call *Weakly Hybrid Metric Proposition Neighborhood Logic*, or WHMPNL, for (not so) short [7]. In fact, we are considering the weakest natural hybrid extension of MPNL that would allow us to store the length of current interval in a variable and unrestrictedly refer to it in sub-formulae in order to express metric properties of intervals. To this end, we introduce a special sort of a binder $\downarrow$, called *length binder*, a countable set of *length variables* $DVar = \{x, y, \ldots\}$ (where $DVar \cap \mathcal{AP} = \emptyset$) and a corresponding set of *hybrid metric constraints* of the kind $\mathsf{len}_{\mathcal{C}\mathsf{x}}$ for each $\mathcal{C} \in \mathbb{C}$ and $x \in DVar$. Formulae of WHMPNL will be evaluated with respect to a suitable valuation functions over length variables and the classical valuation function over propositional letters. Thus, as in classical hybrid logics, we define a *length assignment* $g : DVar \rightarrow \mathbb{N}$. A model for WHMPNL over $\mathbb{N}$ is defined as a quadruple $M = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g \rangle$, where $\mathbb{I}(\mathbb{N})$ is the interval structure on $\mathbb{N}$, $V : \mathcal{AP} \rightarrow 2^{\mathbb{I}(\mathbb{N})}$ is a valuation function for propositional letters, and g is a length assignment. For two length assignments g, g' and a variable x we write $g' \sim_x g$ to mean that g' possibly differs from g only on the value of x. WHMPNL-formulae are defined by the following grammar:

$$\varphi ::= p \mid \mathsf{len}_{\mathcal{C}\mathsf{k}} \mid \mathsf{len}_{\mathcal{C}\mathsf{x}} \mid \neg\varphi \mid \psi \vee \varphi \mid \Diamond_r \varphi \mid \Diamond_l \varphi \mid \downarrow_x \varphi,$$

where $k \in \mathbb{N}$ and $x \in DVar$.

The semantic rules of WHMPNL extend those of MPNL with the clauses:

- $M, [a, b] \Vdash \mathsf{len}_{\mathcal{C}\mathsf{x}}$ iff $\delta(a, b)\mathcal{C}g(x)$;
- $M, [a, b] \Vdash \downarrow_x \varphi$ iff $M', [a, b] \Vdash \varphi$ for $M' = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g' \rangle$, where g' is a length assignment such that $g' \sim_x g$ and $g'(x) = \delta(a, b)$.

Note that a universal (and, respectively, existential) analogue of the hybrid operator @ is definable, too, by $@_x\varphi := [G](\mathsf{len}_{=\mathsf{x}} \rightarrow \varphi)$, with respective semantics:

- $M, [a, b] \Vdash @_x\varphi$ iff for any interval $[c, d]$ such that $\delta(c, d) = g(x)$ it is the case that $M, [c, d] \Vdash \varphi$.

We denote by WHPNL the fragment of WHMPNL not involving atomic propositions for length constraints (that is, sub-formulae of the kind $\mathsf{len}_{=\mathsf{k}}$).

[7] The qualifier 'weakly' indicates that we do not add a full hybrid language over interval logics – as we have noted above, that would immediately lead to irreparable undecidability.

We note that, while the different types of atomic propositions for length constraints (i.e., involving $<$, $=$, $>$, $\leq$, and $\geq$, but only comparing with explicit numbers) are definable in terms of each other, not all of these inter-definitions work for hybrid metric constraints. For example, $\mathsf{len}_{\geq \mathsf{x}}$ is equivalent to $\neg\mathsf{len}_{< \mathsf{x}}$ but it is not possible to define $\mathsf{len}_{\leq \mathsf{x}}$ or $\mathsf{len}_{< \mathsf{x}}$ in terms of $\mathsf{len}_{= \mathsf{x}}$. Therefore, it makes sense to consider also sub-languages of WHMPNL not including all hybrid metric constraints. If L is any of WHMPNL and WHPNL, for any $C \subseteq \mathbb{C}$ we denote by $\mathrm{L}(C)$ the language that includes only hybrid metric constraints of the kind $\mathsf{len}_{\mathcal{C}\mathsf{x}}$, with $\mathcal{C} \in C$. For any such language $\mathrm{L}^{'}$, we denote by $\mathrm{L}^{'}_{n}$ its sub-language containing only n length variables.

4 Undecidability of WHMPNL and Fragments

In this section we show that even the weakest syntactically unrestricted fragment of WHMPNL that still retains $\downarrow_x$ for *only one* length variable x is already undecidable, even when atomic propositions for length constraints are not allowed. That result, while disappointing, is interesting because it shows how dangerously close is the non-hybrid MPNL to undecidability, and raises the question of searching for yet weaker and still meaningful decidable extensions of MPNL.

4.1 Undecidability of WHMPNL

We begin by noting that, while PNL is a strict fragment of the Halpern-Shoham modal logic of Allen's relations HS (whose satisfiability problem is undecidable under very general assumptions for the class of linear orders underlying the interval structures, see [16]), its hybrid metric version is, in fact, at least as expressive as the whole HS. To show that it suffices to define the operators $\langle B \rangle$, corresponding to the interval relation *begins*, and $\langle E \rangle$, corresponding to the interval relation *ends*, and their inverses $\langle \overline{B} \rangle$ and $\langle \overline{E} \rangle$, as all other modal operators in HS are definable in terms of these plus $\Diamond_l$ and $\Diamond_r$ [16]. Indeed:

$$\langle B \rangle p := \downarrow_x \Diamond_l \Diamond_r (p \wedge \mathsf{len}_{< \mathsf{x}}),$$

$$\langle E \rangle p := \downarrow_x \Diamond_r \Diamond_l (p \wedge \mathsf{len}_{< \mathsf{x}}).$$

$$\langle \overline{B} \rangle p = \downarrow_x \Diamond_l \Diamond_r (p \wedge \mathsf{len}_{> \mathsf{x}}),$$

$$\langle \overline{E} \rangle p = \downarrow_x \Diamond_r \Diamond_l (p \wedge \mathsf{len}_{> \mathsf{x}}).$$

Thus, we have the following result.

Theorem 4.1 *The satisfiability problem for WHPNL, and hence for WHMPNL, interpreted over* $\mathbb{N}$ *is undecidable.*

4.2 Undecidability of the fragments

It is known [3] that any of the pairs of HS-modalities $\langle B \rangle$, $\langle E \rangle$ and $\langle \overline{B} \rangle$, $\langle \overline{E} \rangle$ is sufficient to establish undecidability. Thus, even very small fragments of WH-

PNL, namely WHPNL$(<)_1$ and WHPNL$(>)_1$, are expressive enough to render undecidability. Since $\mathsf{len}_{>x}$ is definable in WHPNL$(\leq)_1$, and $\mathsf{len}_{<x}$ is definable in WHPNL$(\geq)_1$, these two languages are undecidable too.

It remains to be seen what happens when only $\mathsf{len}_{=x}$ is allowed. This case turns out rather more difficult, but we will show here that it is undecidable, too. For better understanding, we first show that WHPNL$(\pi,=)_1$, where the modal constant π is added, is undecidable, and then we show how the proof can be adapted to the fragment WHPNL$(=)_1$. We will make use of the undecidability of the *Finite Tiling Problem* [19]. It is the problem of establishing whether, for a given set of tile types $\mathcal{T} = \{t_1, \ldots, t_k\}$, there exists a finite rectangle $\mathcal{R} = [0, X] \times [0, Y] = \{(i, j) : i, j \in \mathbb{N} \wedge 0 \leq i \leq X \wedge 0 \leq j \leq Y\}$ for some $X, Y \in \mathbb{N}$, such that $\mathcal{T}$ can correctly tile $\mathcal{R}$ with the entire border colored by the same designated color \$, also called *side color*. To be more precise, for every tile type $t_i \in \mathcal{T}$, let $right(t_i)$, $left(t_i)$, $up(t_i)$, and $down(t_i)$ be the colors of the corresponding sides of t_i. To solve the Finite Tiling Problem for $\mathcal{T}$ one must find two natural numbers X and Y, and a mapping $f : \mathcal{R} \to \mathcal{T}$ such that

$$\begin{array}{lll} right(f(i,j)) = left(f(i+1,j)) & \text{for each } 0 \leq i < X, 0 \leq j \leq Y, \\ up(f(i,j)) = down(f(i,j+1)) & \text{for each } 0 \leq i \leq X, 0 \leq j < Y, \end{array}$$

and that satisfies, in addition, the following constraints:

$$\begin{array}{llll} left(f(0,j)) = \$ & \text{and} & right(f(X,j)) = \$ & \text{for each } 0 \leq j \leq Y, \\ down(f(i,0)) = \$ & \text{and} & up(f(i,Y)) = \$ & \text{for each } 0 \leq i \leq X. \end{array}$$

where \$ is the side color of $\mathcal{R}$.

In order to perform the reduction from the Finite Tiling Problem for the set of tiles $\mathcal{T} = \{t_1, \ldots, t_k\}$ to the satisfiability problem for WHPNL$(\pi,=)_1$, we will make use of some special propositional letters, namely u, Id, $*$, tile, Start, Stop, $\mathsf{IdStart}$, IdStop, $\mathsf{up\_rel}$, Ltile, Rtile, $\mathtt{t}_1, \mathtt{t}_2, \ldots, \mathtt{t}_k$. For every propositional letter p, by p-*interval* we mean an interval satisfying p. The reduction consists of three main steps:

(i) the encoding of the rectangle by means of a suitable chain of so-called ‘unit’ intervals (u-intervals, for short);

(ii) the encoding of the ‘above-neighbor’ relation by means of a suitable family of so-called up_rel-intervals; and

(iii) the encoding of the ‘right-neighbor’ relation.

Here is a sketch of the encoding. First, we set our framework by forcing the existence of a unique finite chain of u-intervals on the linear ordering (u-*chain*, for short). The u-intervals are used as cells to arrange the tiling. Next, we define a chain of Id-intervals (Id-*chain*, for short), each of them representing a row of the rectangle; the additional propositional letters IdStart and IdStop are used to encode, respectively, the bottom and the top rows of the rectangle. Any Id-interval consists of a sequence of u-intervals. Using the length binder, we force each Id-interval to contain exactly the same number of u-intervals. Each u-interval is used either to represent a part of the plane or to separate two Id-intervals. In the former case, it is labeled with the propositional letter tile, in the latter case, it is labeled with the propositional letter $*$. Then, we use the propositional letter $\mathsf{up\_rel}$ to encode the relation that connects each tile with its above neighbor in $\mathcal{R}$. Finally, we introduce a set of propositional letters $\mathtt{T} = \{\mathtt{t_1}, \mathtt{t_2}, \ldots, \mathtt{t_k}\}$ corresponding to the set of tile types $\mathcal{T} = \{t_1, t_2, \ldots, t_k\}$ and define a formula $\Phi_{\mathcal{T}}$ which is satisfiable if and only if there exists a finite rectangle $\mathcal{R}$ for some $X, Y \in \mathbb{N}$ and a proper tiling of $\mathcal{R}$ by $\mathcal{T}$, i.e., a tiling that satisfies the color constraints on the border tiles and between vertically- and horizontally-adjacent tiles.

To define the u-chain we use the following formulae:

$$\mathsf{u} \wedge \mathsf{Start} \wedge \Box_r \neg \mathsf{Start} \wedge \Diamond_r \Diamond_r (\mathsf{Stop} \wedge \mathsf{u}) \tag{1}$$
$$[G](\mathsf{u} \vee \mathsf{Start} \vee \mathsf{Stop} \rightarrow \mathsf{len}_{=\mathsf{x}}) \tag{2}$$
$$[G](\Diamond_r \mathsf{Start} \rightarrow \Box_r(\neg \pi \rightarrow \Box_r \neg \mathsf{Start})) \tag{3}$$
$$[G](\Diamond_r \mathsf{Stop} \rightarrow \Box_r(\neg \pi \rightarrow \Box_r \neg \mathsf{Stop})) \tag{4}$$

starts the u*-chain*

u, Start, Stop *are equally long*

Start *is unique*

Stop *is unique*

$$[G](\mathsf{u} \wedge \neg \mathsf{Stop} \rightarrow \Diamond_r \mathsf{u}) \tag{5}$$
$$[G]((\mathsf{Start} \rightarrow \Box_l \Box_l \neg \mathsf{u}) \wedge (\mathsf{Stop} \rightarrow \Box_r \Box_r \neg \mathsf{u})) \tag{6}$$
$$(1) \wedge \ldots \wedge (6) \tag{7}$$

u*-chain to the right*

no u *out of the chain*

Lemma 4.2 *Let $M = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g \rangle$ be a WHPNL-model such that*

$$M, [a, b] \Vdash \downarrow_x (7).$$

Then there exists a finite sequence of points $b_0 < b_1 < \ldots < b_k$, with $k > 0$, such that $b_0 = a$, $b_1 = b$, and:

(i) *All intervals $[b_i, b_{i+1}]$, for $0 \leq i \leq k-1$, have the same length $b - a > 0$.*

(ii) *$M, [b_i, b_{i+1}] \Vdash \mathsf{u}$ for each $0 \leq i \leq k-1$.*

(iii) *$M, [c, d] \Vdash \mathsf{u}$ holds for no other interval $[c, d]$.*

Proof First of all, notice that the interval satisfying Start is unique, and same applies to Stop, due to (3) and (4). Indeed, by (1), one interval satisfying Start is $[a, b] = [b_0, b_1]$ and it is not a point-interval because it also satisfies $\Box_r \neg\mathsf{Start}$. Moreover, by (1) and (2), all u-, Start-, and Stop-intervals have the same length $b - a$, hence no two different Start-intervals can start from the same point. Then, by (5), the interval $[b_0, b_1]$ starts a finite chain of u-intervals $[b_i, b_{i+1}]$, with $i \geq 0$. The finiteness follows from the fact that, by (1), some future u-interval satisfies Stop and no interval starting to the right of it is a u-interval, by (6). That interval must belong to the chain, otherwise it will be overtaken by the chain, by (5), which would be a contradiction. Furthermore, any u-interval that is not in that chain may not be to the left of Start, (6), and must start another chain of u-intervals, all of the same lenght $b - a$, and therefore overlapping the u-intervals of the first chain. However, the unique interval satisfying Stop cannot belong to the second chain, and therefore must be overtaken by it – a contradiction with (6).

□

We now define the Id-chain with the following formulae:

$[G]((\mathsf{u} \leftrightarrow \mathsf{tile} \vee *) \wedge (* \rightarrow \neg\mathsf{tile}))$ (8) u *is either* tile *or* $*$

$[G]((\Diamond_r \mathsf{Start} \leftrightarrow \Diamond_r \mathsf{IdStart}) \wedge (\Diamond_l \mathsf{Stop} \leftrightarrow \Diamond_l \mathsf{IdStop}))$ (9) *first and last* Id

$[G](\mathsf{IdStart} \vee \mathsf{IdStop} \rightarrow \mathsf{Id}) \wedge (\mathsf{IdStart} \rightarrow \neg\mathsf{IdStop})$ (10) IdStart, IdStop *def*

$[G](\mathsf{Id} \rightarrow \mathsf{len}_{=\mathsf{x}} \wedge \Diamond_r \Diamond_l \mathsf{tile})$ (11) Id*s same length*

$[G](\Diamond_r \mathsf{Id} \leftrightarrow \Diamond_r *)$ (12) Id*s start with* $*$

$[G]((\mathsf{Id} \wedge \neg\mathsf{IdStop} \rightarrow \Diamond_r \mathsf{Id}) \wedge (\mathsf{Id} \wedge \neg\mathsf{IdStart} \rightarrow \Diamond_l \mathsf{Id}))$ (13) Id*-chain*

$(8) \wedge \ldots \wedge (13) \wedge \mathsf{IdStart} \wedge \Box_r \neg\mathsf{IdStart}$ (14)

Note that the last formula above ensures that the interval satisfying IdStart, and hence any Id-interval, is not a point-interval.

Lemma 4.3 *Let* $M = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g \rangle$ *be a WHPNL-model such that*

$$M, [a, b] \Vdash \downarrow_x (7) \wedge \Diamond_l \Diamond_r \downarrow_x (14).$$

Then there exist positive integers h, v *and a finite sequence of points* $a = b_1^0 < b_1^1 < \ldots < b_1^h = b_2^0 < \ldots < b_2^h = b_3^0 < \ldots < b_{v-1}^h = b_v^0 < \ldots < b_v^h$ *such that for each* $1 \leq j \leq v$*, we have:*

(i) $M, [b_j^0, b_j^1] \Vdash *$.

(ii) $M, [b_j^i, b_j^{i+1}] \Vdash \mathsf{tile}$ *for each* $0 < i < h$.

(iii) $M, [b_j^0, b_j^h] \Vdash \mathsf{Id}$.

Moreover, no other interval satisfies respectively $*$, tile, *and* Id, *but those indicated above.*

Proof First of all, by Lemma 4.2, there is a finite sequence of points $a = b_0 < b_1 < \ldots < b_k$, defining a finite chain of u-intervals. Each of these u-intervals is either a

$*$-interval or a tile-interval and no other interval is a $*$-interval or a tile-interval, by (8). Furthermore, by (12), every $*$-interval starts an Id-interval, every Id-interval starts with a $*$-interval and ends with a tile-interval. Thus, every Id-interval spans several u-intervals. therefore, there are finitely many Id-intervals. Let their number be v. The first u-interval $[a, b]$, which is also the only Start-interval, starts an Id-interval $[a, b_h]$ for some $h < k$, satisfying IdStart, by (9). The only Stop-interval, which is the last u-interval, ends the last Id-interval, satisfying IdStop, again by (9). Since all u-intervals have the same length $b_1 - a$ and all Id-intervals have the same length $b_h - a$, then every Id-interval spans exactly h u-intervals. Hence, the sequence $b_0 < b_1 < \ldots < b_k$ can be written as $b_1^0 < b_1^1 < \ldots < b_1^h = b_2^0 < \ldots < b_2^h = b_3^0 < \ldots < b_{v-1}^h = b_v^0 < \ldots < b_v^h$, as required. Now, the first 3 claims of the lemma are immediate. As for the last one, it suffices to notice that every Id-interval starts a chain of Id-intervals which must terminate, hence it must end with the only IdStop-interval, itself ending with the only Stop-interval. Furthermore, the first possible Id-interval starts with the first possible tile-interval, which is the only Start-interval. Thus, no other Id-intervals exist in M, but those of the type $[b_j^0, b_j^h]$. The rest of claim 4 is now immediate.

□

The above lemma guarantees the existence of an Id-chain. Now we want to force the propositional letter up_rel to correctly encode the relation that connects pairs of tiles of the rectangle $\mathcal{R}$ that are vertically adjacent.

$$[G](\mathsf{up\_rel} \to \mathsf{len}_{=\mathsf{x}} \wedge \Diamond_l \Diamond_r \mathsf{tile}) \qquad (15) \qquad \mathsf{up\_rel}\ \textit{and}\ \mathsf{Id}\ \textit{are equally long}$$

$$[G](\mathsf{tile} \to (\Diamond_r \Diamond_r \mathsf{IdStop} \leftrightarrow \Diamond_l \Diamond_r \mathsf{up\_rel})) \qquad (16) \qquad \mathsf{tile}\ \textit{starts}\ \mathsf{up\_rel}$$

$$(15) \wedge (16) \qquad (17)$$

Lemma 4.4 *Let $M = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g \rangle$ be a WHPNL-model such that*

$$M, [a, b] \Vdash \downarrow_x (7) \wedge \Diamond_l \Diamond_r \downarrow_x ((14) \wedge (17))$$

and let $a = b_1^0 < b_1^1 < \ldots < b_1^h = b_2^0 < \ldots < b_2^h = b_3^0 < \ldots < b_v^0 < \ldots < b_v^h$ be the sequence of points guaranteed by Lemma 4.3. Then, for each $1 \leq j < v$, the interval $[b_j^i, b_{j+1}^i]$ satisfies up_rel, *and no other interval satisfies* up_rel.

Proof By the first conjunct of (15), we have that up_rel-intervals have the same length of Id-intervals. By (16), each tile-interval, but the ones belonging to the last Id-interval, starts a up_rel-interval. Finally, by the second conjunct of (15), each up_rel-interval is started by a tile-interval. Given that all u-intervals are equally long and every Id-interval spans the same number of u-intervals, the claim follows immediately from Lemma 4.3. □

Finally, we can force all tile-matching conditions to be respected, by using the following formulae:

$$[G](((\mathsf{tile} \wedge \Diamond_r *) \vee (\mathsf{tile} \wedge \mathsf{Stop})) \leftrightarrow \mathsf{Rtile}) \qquad (18) \qquad \textit{right side } \mathsf{tile}\textit{s}$$

$$[G](\mathsf{tile} \wedge \Diamond_l * \leftrightarrow \mathsf{Ltile}) \qquad (19) \qquad \textit{left side } \mathsf{tile}\textit{s}$$

$$[G]((\mathsf{tile} \leftrightarrow \bigvee_{\mathtt{t_q} \in \mathcal{T}} \mathtt{t_q}) \wedge \bigwedge_{\mathtt{t_q}, \mathtt{t_u} \in \mathcal{T}, \mathtt{t_q} \neq \mathtt{t_u}} \neg(\mathtt{t_q} \wedge \mathtt{t_u})) \qquad (20) \qquad \mathsf{tile}\textit{s are tiles}$$

$$[G](\mathsf{tile} \wedge \Diamond_r \mathsf{tile} \rightarrow \bigvee_{right(\mathtt{t_q}) = left(\mathtt{t_u})} (\mathtt{t_q} \wedge \Diamond_r \mathtt{t_u})) \qquad (21) \qquad \textit{right-left constraint}$$

$$[G](\mathsf{up\_rel} \rightarrow \bigvee_{up(\mathtt{t_q}) = down(\mathtt{t_u})} (\Diamond_l \Diamond_r \mathtt{t_q} \wedge \Diamond_r \mathtt{t_u})) \qquad (22) \qquad \textit{up-down constraint}$$

Moreover, being $\mathcal{T}_r$ (resp., $\mathcal{T}_l$, $\mathcal{T}_u$, $\mathcal{T}_d$) the subset of $\mathcal{T}$ in which all tiles have the right (resp., left, up, down) side colored with \$, we can force the side constraints:

$$[G](\Diamond_l \mathsf{IdStart} \rightarrow \Box_l \Box_l (\mathsf{tile} \rightarrow \bigvee_{\mathtt{t_q} \in \mathcal{T}_d} \mathtt{t_q})) \qquad (23) \qquad \textit{down side constraint}$$

$$[G](\Diamond_r \mathsf{IdStop} \rightarrow \Box_r \Box_r (\mathsf{tile} \rightarrow \bigvee_{\mathtt{t_q} \in \mathcal{T}_u} \mathtt{t_q})) \qquad (24) \qquad \textit{up side constraint}$$

$$[G](\mathsf{Ltile} \rightarrow \bigvee_{\mathtt{t_q} \in \mathcal{T}_l} \mathtt{t_q}) \qquad (25) \qquad \textit{left side constraint}$$

$$[G](\mathsf{Rtile} \rightarrow \bigvee_{\mathtt{t_q} \in \mathcal{T}_r} \mathtt{t_q}) \qquad (26) \qquad \textit{right side constraint}$$

$$(18) \wedge \ldots \wedge (26) \qquad (27)$$

The following theorem implies the undecidability of the logic WHPNL$(\pi, =)_1$.

Theorem 4.5 *Given any finite set of tiles $\mathcal{T}$ and a side color \$, the formula*

$$\Phi := \downarrow_x (7) \wedge \Diamond_l \Diamond_r \downarrow_x ((14) \wedge (17) \wedge (27))$$

is satisfiable in $\mathbb{N}$ if and only if $\mathcal{T}$ can tile some finite rectangle $\mathcal{R}$ with side color \$.

Proof (Only if:): Suppose that $M, [a,b] \Vdash \Phi$. Then, by Lemma 4.3, there is a sequence of points $b_0 = b_1^0 < b_1^1 < \ldots < b_1^h = b_2^0 < \ldots < b_2^h = b_3^0 < \ldots < b_v^0 < \ldots < b_v^h = b_k$. We put $X = h - 1$ and $Y = v$. We have that $M, [b_s^r, b_{s+1}^r] \Vdash \mathsf{tile}$ if and only if $s > 0$, which implies $M, [b_s^r, b_{s+1}^r] \Vdash \mathtt{t_q}$ for a unique $\mathtt{t_q}$. Now, for all s, r, where $1 \leq s \leq X$, $1 \leq r \leq Y$, define $f(s,r) = t_g$ if and only if $M, [b_s^r, b_{s+1}^r] \Vdash \mathtt{t_q}$. From Lemma 4.2, 4.3, and 4.4 it follows that the function $f : \mathcal{R} \rightarrow \mathcal{T}$ defines a correct tiling of $\mathcal{R}$, where X and Y are defined as above.

(If:) Let $f : \mathcal{R} \mapsto \mathcal{T}$ be a correct tiling function of the rectangle $\mathcal{R} = [0, X] \times [0, Y]$ for some X, Y, and a given border color \$. For convenience, we will identify the tile-variables with their corresponding tiles from $\mathcal{T}$. We will show that there exist a model M and an interval $[a,b]$ such that $M, [a,b] \Vdash \Phi$. Let $n = (X+1) \cdot Y$, we define a model $M = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g \rangle$ such that $M, [0,1] \Vdash \Phi$. We must provide a valuation function V. Since the only length variable occurring in Φ is x and it has no free occurrences there, any valuation of x would be as good as any other, so we put $g(x) = 1$. Then, for each $i, j \in \mathbb{N}$, we put:

$$\begin{aligned} V(\mathsf{u}) &:= \{[i, i+1] \mid 0 \leq i < n\}; \\ V(\mathsf{Start}) &:= \{[0, 1]\}; \\ V(\mathsf{Stop}) &:= \{[n-1, n]\}. \end{aligned}$$

This guarantees that (7) is satisfied. Now, in order to satisfy the remaining part of Φ on $[0, 1]$ it suffices to show that the formula $\Diamond_l \Diamond_r \downarrow_x ((14) \wedge (17) \wedge (27))$ can be satisfied on the interval $[0, X+1]$, i.e., $(14) \wedge (17) \wedge (27)$ can be satisfied on $[0, X+1]$ by a valuation assigning value $X + 1$ to the length variable x. With the following, we define the valuation for the remaining propositional letters:

$$\begin{aligned} V(\mathsf{Id}) &:= \{[i \cdot (X+1), (i+1) \cdot (X+1)] \mid 0 \leq i < Y\}; \\ V(*) &:= \{[i \cdot (X+1), i \cdot (X+1) + 1] \mid 0 \leq i < Y\}; \\ V(\mathsf{tile}) &:= V(\mathsf{u}) \setminus V(*); \\ V(\mathsf{IdStart}) &:= \{[0, X+1]\}; \\ V(\mathsf{IdStop}) &:= \{[(X+1) \cdot (Y-1), (X+1) \cdot Y]\}; \\ V(\mathsf{up\_rel}) &:= \{[i, j] \mid \delta(i, j) = X+1, [i, j] \notin V(\mathsf{Id}), 0 \leq i, j < n\}; \\ V(\mathsf{Ltile}) &:= \{[i \cdot (X+1) + 1, i \cdot (X+1) + 2] \mid 0 \leq i < Y\}; \\ V(\mathsf{Rtile}) &:= \{[i \cdot (X+1) - 1, i \cdot (X+1)] \mid 0 < i \leq Y\}. \end{aligned}$$

Finally, we evaluate the tile-variables as follows. For each $\mathtt{t} \in \mathtt{T}$:

$$V(\mathtt{t}) := \{[i + (j-1) \cdot (X+1), i + (j-1) \cdot (X+1) + 1)] \mid f(i, j) = t\}.$$

It is now straightforward to check that $M, [0, 1] \Vdash \Phi$, hence the claim. □

Now, we will sketch how we can eliminate the modal constant π from the formulae in the construction above. Note that the modal constant π is involved only in formulae (3) and (4), forcing uniqueness of the u-intervals Start and Stop, and consequently uniqueness of the u-chain. Without the modal constant π in the language, it seems that we cannot force that uniqueness anymore. Still, we can ensure that any two Start intervals must overlap, by replacing the formula (3) with

$$[G](\mathsf{Start} \rightarrow \mathsf{u} \wedge \Box_r \Box_r \neg \mathsf{Start}).$$

Likewise for the Stop-intervals, by replacing (4) with

$$[G](\mathsf{Stop} \rightarrow \mathsf{u} \wedge \Box_r \Box_r \neg \mathsf{Stop}).$$

Thus, other u-chains may be possible in a model satisfying the resulting modification of $\downarrow_x$ (7), but they all would have the same length d and number k of u-intervals and may differ from each other by a translation to the left or right at a distance less than d. By a similar argument, the same would happen to the ld-chains in every model of the modified formula $\downarrow_x$ (7) $\wedge \Diamond_l \Diamond_r \downarrow_x$ ((14) $\wedge$ (17) $\wedge$ (27)). Eventually, every model of the modified formula Φ would have possibly several isomorphic copies of the encoding of the required finite tiling; conversely, every correct finite tiling would generate a model satisfying the modified formula Φ. We leave the further details to the reader. Thus, we obtain the following result.

Theorem 4.6 *The satisfiability problem for WHPNL(=)$_1$ interpreted over $\mathbb{N}$ is undecidable.*

Another minor modification of some of the formulae above can reduce the Finite Tiling Problem to the satisfiability problem of the logic WHPNL(=)$_1$ interpreted over $\mathbb{N}$ with *strict semantics*, thus excluding point intervals [8], hereafter denoted WHPNL(=)$_1^-$. Essentially the only necessary changes in the formulae used in the encoding of the tiling problem are to replace formulae of the type $\Box_r \Box_r \psi$ with $\Box_r \psi \wedge \Box_r \Box_r \psi$, likewise $\Box_l \Box_l \psi$ with $\Box_l \psi \wedge \Box_l \Box_l \psi$, and, respectively, $\Diamond_r \Diamond_r \psi$ with $\Diamond_r \psi \wedge \Diamond_r \Diamond_r \psi$, likewise $\Diamond_l \Diamond_l \psi$ with $\Diamond_l \psi \wedge \Diamond_l \Diamond_l \psi$. The rest should be essentially the same, save for the fact that the complications coming from the point intervals will now disappear.

5 Regaining Decidability: PNL with Restricted Interval Length Binders

As we have seen so far, adding even a single length variable and a binder over it to PNL leads to undecidability. The natural question that arises is whether there is any natural decidable extension of PNL or MPNL that still allows a restricted use of length binders, i.e. memory. So far, our proposal to regain decidability is to limit the range of the binders over length variables, by replacing the binder $\downarrow_x$ with a hierarchy of restricted versions $\{\downarrow_x^k \mid k \in \mathbb{N}\}$ and modifying suitably the semantics when the length of the current interval exceeds the limit of the binder. We consider two such versions of the semantics of $\downarrow_x^k$:

1. *Restricted semantics*: $M, [a, b] \Vdash^r \downarrow_x^k \varphi$ iff
   - i) $\delta(a, b) \leq k$ and $M', [a, b] \Vdash \varphi$ for $M' = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g' \rangle$, where g' is the assignment such that $g' \sim_x g$ and $g'(x) = \delta(a, b)$, or
   - ii) $\delta(a, b) > k$ and $M', [a, b] \Vdash \varphi$ for $M' = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g' \rangle$, and for *some* assignment g' such that $g' \sim_x g$ and $g'(x) > k$.
2. *Truncated semantics*: $M, [a, b] \Vdash^t \downarrow_x^k \varphi$ iff
   - i) $\delta(a, b) \leq k$ and $M', [a, b] \Vdash \varphi$ for $M' = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g' \rangle$, where g' is the assignment such that $g' \sim_x g$ and $g'(x) = \delta(a, b)$, or

[8] In the strict semantics the operators $\Diamond_r$ and $\Diamond_l$ are usually denoted as in HS, that is, by $\langle A \rangle$ and $\langle \overline{A} \rangle$, respectively.

ii) $\delta(a, b) > k$ and $M', [a, b] \Vdash \varphi$ for $M' = \langle \mathbb{N}, \mathbb{I}(\mathbb{N}), V, g' \rangle$, where g' is the assignment such that $g' \sim_x g$ and $g'(x) = k + 1$.

The intuition is clear: the binder $\downarrow_x^k$ can only store the length of the current interval if it does not exceed k, otherwise in the truncated semantics it stores $k + 1$ and in the restricted semantics it only stores the constraint $\mathsf{len}_{>\mathsf{k}}$.

Now, let us consider the *restricted fragment* WHMPNL^r(=) of the logic WHMPNL(=), where only restricted length binders may occur and the variable length constraints of the type $\mathsf{len}_{=\mathsf{x}}$ may only occur *positively* (i.e., we do not allow sub-formulae of the kind $\neg\mathsf{len}_{=\mathsf{x}}$, after the formula is transformed to a negation normal form), interpreted with the restricted semantics. The *truncated fragment* WHMPNL^t(=) of WHMPNL(=) is defined likewise, but without the polarity restriction on the occurrences of $\mathsf{len}_{=\mathsf{x}}$, and interpreted with the truncated semantics. It is not obvious whether any of these fragments subsumes the other one in terms of expressiveness with respect to models on $\mathbb{N}$, and it is possible that they are incomparable in that sense.

Theorem 5.1 *The following hold:*

(i) *Every formula ψ of WHMPNL^r(=) can be effectively translated to a formula $\tau^r(\psi)$ of MPNL which is equisatisfiable with ψ when interpreted over $\mathbb{N}$ and has length at most exponential in the length of ψ.*

(ii) *Every formula ψ of WHMPNL^t(=) can be effectively translated to a formula $\tau^t(\psi)$ of MPNL which is equisatisfiable with ψ when interpreted over $\mathbb{N}$ and has length at most exponential in the length of ψ.*

Proof Both translations, for WHMPNL^r(=) and WHMPNL^t(=), into MPNL, distribute over all logical connectives except the length binders, on which they act respectively as follows, where $A[Z/\mathsf{len}_{=\mathsf{x}}]$ is the result of simultaneous substitution of all *free* occurrences of $\mathsf{len}_{=\mathsf{x}}$ (i.e., not in the scope of a $\downarrow_x^k$) by Z in A:

$$\tau^r(\downarrow_x^k \psi) := (\mathsf{len}_{>\mathsf{k}} \wedge \tau^r(\psi)[\mathsf{len}_{>\mathsf{k}}/\mathsf{len}_{=\mathsf{x}}]) \vee \bigvee_{j=0}^{k} (\mathsf{len}_{=\mathsf{j}} \wedge \tau^r(\psi)[\mathsf{len}_{=\mathsf{j}}/\mathsf{len}_{=\mathsf{x}}]).$$

$$\tau^t(\downarrow_x^k \psi) := (\mathsf{len}_{>\mathsf{k}} \wedge \tau^t(\psi)[\mathsf{len}_{=\mathsf{k+1}}/\mathsf{len}_{=\mathsf{x}}]) \vee \bigvee_{j=0}^{k} (\mathsf{len}_{=\mathsf{j}} \wedge \tau^t(\psi)[\mathsf{len}_{=\mathsf{j}}/\mathsf{len}_{=\mathsf{x}}]).$$

We claim that each of these translations, when applied to formulae in the respective languages, produces a formula equisatisfiable with the original one. For lack of space, we leave the details to the reader. □

Note that the translation τ^r does not work correctly when a variable length constraint $\mathsf{len}_{=\mathsf{x}}$ occurs negatively, because $\neg\mathsf{len}_{=\mathsf{x}}$ is not equivalent to $\neg\mathsf{len}_{>\mathsf{k}}$ when $x > k$. For instance, $\downarrow_x^k (\mathsf{len}_{>\mathsf{k}} \wedge \Diamond_r(\neg\mathsf{len}_{=\mathsf{x}} \wedge \mathsf{len}_{>\mathsf{k}}))$ is satisfiable in the restricted semantics but $\tau^r(\downarrow_x^k (\mathsf{len}_{>\mathsf{k}} \wedge \Diamond_r(\neg\mathsf{len}_{=\mathsf{x}} \wedge \mathsf{len}_{>\mathsf{k}}))) = (\mathsf{len}_{>\mathsf{k}} \wedge \Diamond_r(\neg\mathsf{len}_{>\mathsf{k}} \wedge \mathsf{len}_{>\mathsf{k}})) \vee \bigvee_{j=0}^{k}(\mathsf{len}_{=\mathsf{j}} \wedge (\mathsf{len}_{>\mathsf{k}} \wedge \Diamond_r(\neg\mathsf{len}_{=\mathsf{j}} \wedge \mathsf{len}_{>\mathsf{k}})))$ is not. That problem does not arise for the translation τ^t.

Now, the following is immediate from Theorem 5.1 and Theorem 2.1.

Corollary 5.2 *The satisfiability problem for both the fragments* $WHMPNL^r(=)$ *and* $WHMPNL^t(=)$*, interpreted over* $\mathbb{N}$*, is decidable in 3NEXPTIME, when length constraints in the formulae are represented in binary, and in 2NEXPTIME, when length constraints in the formulae are constant or represented in unary.*

6 Conclusions and Future Directions

In this paper we have demonstrated that extending (M)PNL with variables and binders over interval lengths is natural, but generally leads to undecidability even in very restricted fragments. While these results are somewhat disappointing, they show that strong restrictions must be imposed on the application of length binders in order to retain the decidability of the non-hybrid fragment. The restrictions proposed here render the resulting languages no more expressive than their non-hybrid fragments, so the question whether an essential gain of expressiveness can be obtained by adding some hybrid machinery to interval logic and retaining decidability remains open.

References

[1] Allen, J. F., *Maintaining knowledge about temporal intervals*, Communications of the ACM **26** (1983), pp. 832–843.

[2] Bolander, T., J. Hansen and M. R. Hansen, *Decidability of a hybrid duration calculus*, Electronic Notes in Theoretical Computer Science **174** (2007), pp. 113–133.

[3] Bresolin, D., D. Della Monica, V. Goranko, A. Montanari and G. Sciavicco, *Decidable and Undecidable Fragments of Halpern and Shoham's Interval Temporal Logic: Towards a Complete Classification*, in: *Proc. of 15th Int. Conf. on Logic for Programming, Artificial Intelligence, and Reasoning*, LNCS **5330** (2008), pp. 590–604.

[4] Bresolin, D., D. Della Monica, V. Goranko, A. Montanari and G. Sciavicco, *Metric propositional neighborhood logic: Expressiveness, decidability, and undecidability*, in: *Proc. of the 19th European Conference on Artificial Intelligence (ECAI 2010)*, 2010, pp. 695–700.

[5] Bresolin, D., V. Goranko, A. Montanari and G. Sciavicco, *On Decidability and Expressiveness of Propositional Interval Neighborhood Logics*, in: *Proc. of the International Symposium on Logical Foundations of Computer Science*, LNCS **4514** (2007), pp. 84–99.

[6] Bresolin, D., V. Goranko, A. Montanari and G. Sciavicco, *Propositional interval neighborhood logics: Expressiveness, decidability, and undecidable extensions*, Annals of Pure and Applied Logic **161** (2009), pp. 289 – 304.

[7] Bresolin, D., V. Goranko, A. Montanari and G. Sciavicco, *Right propositional neighborhood logic over natural numbers with integer constraints for interval lengths*, in: *Proc. of the 7th IEEE Conf. on Software Engeneering and Formal Methods*, 2009, pp. 240–249.

[8] Bresolin, D., A. Montanari and P. Sala, *An optimal tableau-based decision algorithm for Propositional Neighborhood Logic*, in: *Proc. of STACS 2007*, LNCS **4393** (2007), pp. 549–560.

[9] Chakravorty, G. and P. K. Pandya, *Digitizing Interval Duration Logic*, in: *CAV*, 2003, pp. 167–179.

[10] Chaochen, Z. and M. Hansen, *An adequate first order interval logic*, in: *Proc. of the 1997 International Symposium on Compositionality: The Significant Difference*, LNCS **1536** (1998), pp. 584–608.

[11] Chaochen, Z. and M. R. Hansen, "Duration Calculus: A Formal Approach to Real-Time Systems," EATCS Monographs in Theoretical Computer Science, Springer, 2004, 257 pp.

[12] Chaochen, Z., C. A. R. Hoare and A. P. Ravn, *A calculus of durations*, Information Processing Letters **40** (1991), pp. 269–276.

[13] Chaochen, Z., Z. Jingzhong, Y. Lu and L. Xiaoshan, *Linear Duration Invariants*, in: H. Langmaack, W.-P. de Roever and J. Vytopil, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems (FTRTFT 2004)*, Lecture Notes In Computer Science **863** (1994), pp. 86–109.

[14] Fränzle, M. and M. R. Hansen, *Deciding an interval logic with accumulated durations*, in: *TACAS*, 2007, pp. 201–215.

[15] Goranko, V., A. Montanari and G. Sciavicco, *Propositional interval neighborhood temporal logics*, Journal of Universal Computer Science **9** (2003), pp. 1137–1167.

[16] Halpern, J. and Y. Shoham, *A propositional modal logic of time intervals*, Journal of the ACM **38** (1991), pp. 935–962.

[17] Hansen, M. and D. V. Hung, *A theory of duration calculus with application*, in: *Domain Modeling and the Duration Calculus*, LNCS **4710** (2007), pp. 119–176.

[18] Hansen, M. R. and Z. Chaochen, *Duration calculus: Logical foundations*, Formal Aspects of Computing **9** (1997), pp. 283–330.

[19] Kari, J., *Reversibility and surjectivity problems of cellular automata*, Journal of Computer Systems and Science **48** (1994), pp. 149–182.

[20] Krishna, S. N. and P. K. Pandya, *Modal Strength Reduction in Quantified Discrete Duration Calculus*, in: *FSTTCS*, 2005, pp. 444–456.

[21] Montanari, A., G. Puppis and P. Sala, *Maximal Decidable Fragments of Halpern and Shoham's Modal Logic of Intervals*, in: *Proc. of the 37th International Colloquium on Automata, Languages and Programming (ICALP)*, 2010.

[22] Moszkowski, B., "Reasoning about digital circuits," Tech. rep. stan-cs-83-970, Dept. of Computer Science, Stanford University, Stanford, CA (1983).

[23] Pandya, P. K., *Interval duration logic: Expressiveness and decidability*, Electronic Notes in Theoretical Computer Science **65** (2002), pp. 254 – 272, theory and Practice of Timed Systems (Satellite Event of ETAPS 2002).