Available online at http://scik.org
J. Math. Comput. Sci. 3 (2013), No. 6, 1405-1429
ISSN: 1927-5307

TRANSLATION OF AN ALLEN'S TEMPORAL LOGIC VARIANT TO RPNL

MUHAMMAD ARZAKI* AND ANGGHA SATYA NUGRAHA

Formal Methods in Software Engineering Laboratory, Faculty of Computer Science,

Universitas Indonesia, Depok (16424), Indonesia

Abstract. In this paper, we investigate the relationship between Allen's temporal logic (ATL) and the right propositional neighborhood logic (RPNL). We introduce ATL_B^+ as a variant of ATL that includes Holds and Occurs predicates and its models are interpreted over arbitrary bounded below linearly ordered sets. We show that any ATL_B^+ formula can be translated in linear time into an equisatisfiable RPNL formula. This translation makes the verification techniques of a system whose specification is described using ATL_B^+ can be performed using RPNL.

Keywords: Allen's temporal logic (ATL), translation, right propositional neighborhood logic (RPNL).

2000 AMS Subject Classification: 03B44, 03D15

*Corresponding author

Received August 22, 2013

1. Introduction

Allen's interval algebra, also known as Allen's temporal logic or ATL [9], is a temporal logic which is characterized by the relationship of *events*. Since it was first introduced in 1983 [1], it is frequently used in artificial intelligence, particularly in planning [9]. One reason is because the syntax and semantics of ATL are both intuitively easy to understand.

Right propositional neighborhood logic (RPNL) is the *future-time* fragment of propositional neighborhood logic (PNL). It was first proposed by Goranko et al. in 2003 [8]. Satisfiability problems for RPNL and PNL over discrete and dense linearly ordered sets is decidable [5, 6]. These results mean that RPNL and PNL are suitable for *continuous-time* systems verification.

In 2006, Roşu and Bensalem [9] investigated the relationship between ALTL (Allen linear temporal logic) and LTL (linear temporal logic). ALTL is a variant of ATL whose models are interpreted over linear domains that are order isomorphic to natural numbers. It is shown that any ALTL formula can be translated into an *equisatisfiable*[1] LTL formula using a linear-time algorithm. This translation enables LTL verification techniques on a system whose specification is described using ALTL. Yet, since any ALTL model is interpreted over a discrete domain, ALTL is not always appropriate for *continuous-time* systems formalization.

In this paper we give a formal and systematic investigation of the relationship between ATL over arbitrary (discrete or dense) linearly ordered domains and the temporal logic RPNL. To have a semantic basis for such a relationship, we first define ATL_B^+ as a variant of ATL whose models can be interpreted over arbitrary bounded below linearly ordered sets. We adapt some of our definitions, lemmas, and theorems from [9].

[1]In logic, two formulas are *equisatisfiable* if both are satisfiable or both are not.

2. PRELIMINARIES

2.1. **Some Basic Notations.** Readers are assumed to be familiar with linear order. We recall here only some basics and introduce our notations.

Definition 2.1. Let $(T, \leq)$ be a linearly ordered set, i.e. every two elements of T are comparable. Suppose A and B are two non-empty subsets of T. We define $A \leq B$ iff $a \leq b$ for all $a \in A$ and $b \in B$. Again by abuse of notation, for $t \in T$, we define $t \leq A$ (resp. $t \geq A$) iff $t \leq a$ (resp. $t \geq a$) for all $a \in A$. The notations $A < B$, $t < A$, and $t > A$ are defined analogously. In addition, the boundedness, minimal element, maximal element, infimum (*greatest lower bound*), and supremum (*least upper bound*) for any non-empty subset of T are defined as usual.

A linearly ordered set $(T, \leq)$ is: *discrete* iff for all $a, b \in T$ such that $a < b$ there are $c, d \in T$ such that $a < c \leq b$ and $a \leq d < b$, and there is no $e \in T$ satisfying $a < e < c$ or $d < e < b$; *dense* iff for all $a, b \in T$ such that $a < b$ there is $c \in T$ satisfying $a < c < b$; and *bounded below* iff there is $m \in T$ such that $m \leq T$. The element m is called the *minimal element* of T. The following definition gives the formal description of an interval over a linearly ordered set.

Definition 2.2. Let $(T, \leq)$ be a linearly ordered set. An interval I over T is a non-empty subset of T with a *convex property*, i.e. for all $i_1, i_2 \in I$ and $t \in T$, we have $i_1 < t < i_2$ implies $t \in I$. We say I is closed iff $\inf(I), \sup(I) \in I$. A closed interval with $\inf(I) = a$ and $\sup(I) = b$ is denoted by $[a, b]$. An interval is a strict interval iff $a < b$, while it is a point interval if $a = b$.

In this paper, the term interval refers to a closed interval unless otherwise specified. We denote the set of all closed intervals over $(T, \leq)$ by $\mathbb{I}(T)$ or simply $\mathbb{I}$ when T is understood from the context.

2.2. RPNL**: Syntax, Semantics, and Satisfiability Problem.** We give a brief introduction to the syntax and semantics of right propositional neighborhood logic (RPNL) interpreted over arbitrary linearly ordered sets. More detailed explanation regarding RPNL's variants, syntaxes, semantics, proof systems, and satisfiability problems can be found in [5, 6, 8] and the references therein. There are three common variants of RPNL, namely: RPNL$^-$, RPNL$^+$, and RPNL$^{\pi+}$. Bresolin et al. show that RPNL$^{\pi+}$ is strictly more expressive than RPNL$^-$ and RPNL$^+$ [3, 4]. Due to this reason, we only use RPNL$^{\pi+}$ throughout this paper. From now on, the term RPNL here refers to RPNL$^{\pi+}$.

The language of RPNL consists of a set $\mathscr{P}$ of atomic propositions; the propositional connectives $\neg$ and $\vee$; the modal constant π; and the right neighborhood modality $\Diamond_r$. The other propositional connectives, as well as the logical constants $\top$ (true) and $\bot$ (false), are defined as usual. The formulas of RPNL are defined recursively by the following grammar:

$$\varphi ::= p \mid \pi \mid \neg\varphi \mid \varphi \vee \varphi \mid \Diamond_r \varphi,$$

where $p \in \mathscr{P}$. We use the modality $\Box_r$ as a shorthand for $\neg\Diamond_r\neg$. Henceforth, we identify $\neg\Diamond_r\neg\varphi$ with $\Box_r\varphi$. A formula of the form $\Diamond_r(\Diamond_r\varphi)$, $\Diamond_r(\Box_r\varphi)$, $\Box_r(\Diamond_r\varphi)$, or $\Box_r(\Box_r\varphi)$, will be written without parentheses, i.e. $\Diamond_r\Diamond_r\varphi$, $\Diamond_r\Box_r\varphi$, $\Box_r\Diamond_r\varphi$, or $\Box_r\Box_r\varphi$, respectively.

We will sometimes write RPNL$_{\mathscr{P}}$ to emphasize RPNL which relies exclusively on a set $\mathscr{P}$ of atomic propositions. The set of all well-formed RPNL$_{\mathscr{P}}$ formulas is denoted by $\mathfrak{F}(\text{RPNL}_{\mathscr{P}})$ or simply $\mathfrak{F}(\text{RPNL})$ when $\mathscr{P}$ is understood from the context. The size of an RPNL formula φ, denoted by $|\varphi|$, is the number of operators and modalities in φ.

An RPNL$_{\mathscr{P}}$ *model* is a tuple $\mathbf{M} = (\mathbb{I}(T), \mathscr{V})$, where $\mathbb{I}(T)$ is the set of all intervals over $(T, \leq)$, and $\mathscr{V} : \mathbb{I}(T) \to 2^{\mathscr{P}}$ is a *valuation function* that assigns to every interval the set of atomic propositions true on it. Given a model $\mathbf{M} = (\mathbb{I}(T), \mathscr{V})$ and an interval $[a,b] \in \mathbb{I}(T)$, the semantics of RPNL$_{\mathscr{P}}$ is defined recursively by the *satisfiability relation* $\Vdash$ as follows: for every $p \in \mathscr{P}$, $\mathbf{M}, [a,b] \Vdash p$ iff $p \in \mathscr{V}([a,b])$; $\mathbf{M}, [a,b] \Vdash \pi$ iff $a = b$; $\mathbf{M}, [a,b] \Vdash \neg\varphi$ iff $\mathbf{M}, [a,b] \nVdash \varphi$; $\mathbf{M}, [a,b] \Vdash \varphi_1 \vee \varphi_2$ iff $\mathbf{M}, [a,b] \Vdash \varphi_1$ *or* $\mathbf{M}, [a,b] \Vdash \varphi_2$; and $\mathbf{M}, [a,b] \Vdash \Diamond_r\varphi$ iff there is $c \geq b$ such that $\mathbf{M}, [b,c] \Vdash \varphi$.

An RPNL formula φ is *satisfiable* iff there is a model $\mathbf{M} = (\mathbb{I}(T), \mathscr{V})$ and $[a,b] \in \mathbb{I}(T)$ such that $\mathbf{M}, [a,b] \Vdash \varphi$. The satisfiability problem for RPNL is the problem of determining whether an RPNL formula is satisfiable of not. The satisfiability problem for RPNL over any dense linearly ordered set is decidable in NEXPTIME [6].

2.3. HRPNL**: Syntax and Semantics.** There is no constraint on the valuation function in RPNL. In particular, given an interval $[a,b]$, it may happen that $p \in \mathscr{V}([a,b])$ but $p \notin \mathscr{V}([c,d])$ for some $[c,d] \subset [a,b]$. In other words, the truth of an atomic formula at an interval does not necessarily imply the truth of that atomic formula at every subinterval of it. Due to this reason, we now construct a variant of RPNL that is imposed to a homogeneity restriction for all atomic propositions.

The homogeneity restriction is applied in such a way that for any $p \in \mathscr{P}$, the satisfiability of p at an interval $[a,b]$ implies the satisfiability of p at every subinterval of $[a,b]$. We define HRPNL as a variant of RPNL whose syntax is exactly the same as RPNL's, but with slightly different semantics. A version of HRPNL which relies exclusively on a set $\mathscr{P}$ of atomic propositions will sometimes denoted by HRPNL$_{\mathscr{P}}$. The set of all well-formed HRPNL$_{\mathscr{P}}$ formulas is denoted by $\mathfrak{F}(\text{HRPNL}_{\mathscr{P}})$ or simply $\mathfrak{F}(\text{HRPNL})$ when $\mathscr{P}$ is understood from the context. The following two definitions formalize the semantics of HRPNL.

Definition 2.3. A *model* for an HRPNL$_{\mathscr{P}}$ formula is a tuple $M = (\mathbb{I}(T), V)$, where $\mathbb{I}(T)$ is the set of all intervals over $(T, \leq)$ and $V : T \to 2^{\mathscr{P}}$ is a *valuation function* that assigns to every element of T the set of atomic propositions true on it.

Definition 2.4. Let $M = (\mathbb{I}(T), V)$ be an HRPNL$_{\mathscr{P}}$ model and let $[a,b] \in \mathbb{I}(T)$. Given an HRPNL$_{\mathscr{P}}$ formula φ, the satisfiability relation $M, [a,b] \Vdash \varphi$ is defined recursively as follows: for every $p \in \mathscr{P}$, $M, [a,b] \Vdash p$ iff $p \in V(t)$, for all $t \in [a,b]$; the satisfiability relation definition for a formula of the form π, $\neg\varphi$, $\varphi_1 \vee \varphi_2$, or $\Diamond_r \varphi$, is equivalent to satisfiability relation definition for corresponding formula in RPNL.

From Definition 2.3, for any HRPNL model $M = (\mathbb{I}(T), V)$, we can construct a corresponding RPNL model $\mathbf{M} = (\mathbb{I}(T), \mathscr{V})$ by defining $\mathscr{V}$ in such a way that for any $[a,b] \in \mathbb{I}(T)$: $p \in \mathscr{V}([a,b])$ iff $p \in V(t)$ for all $t \in [a,b]$. Furthermore, Definition 2.4 implies that for any $p \in \mathscr{P}$ and $[a,b] \in \mathbb{I}(T)$, whenever $M, [a,b] \Vdash p$, then for all $[c,d] \subseteq [a,b]$, we have $M, [c,d] \Vdash p$.

3. Allen's Temporal Logic (ATL), ATL$^+$, and ATL^+_B

Allen's temporal logic (ATL) is a formal framework for specifying relative temporal information about two events as in "event 1 overlaps event 2" [1]. ATL formulas are constructed from predicates that express binary relation between events based on their temporal interpretations. The temporal interpretation of any event in ATL is an interval.

3.1. **Intuitive Meaning of** ATL **Formulas.**

There are thirteen basic relations between any two events in ATL: Equals, Before, After, Meets, MetBy, Overlaps, OverlappedBy, Contains, During, Starts, StartedBy, Ends, and EndedBy[2]. The relations After, MetBy, OverlappedBy, During, StartedBy, and EndedBy, are the inverses of Before, Meets, Overlaps,

[2]We adapt [9]'s choice of notation, which can differ from what is found in some other works.

Contains, Starts, and Ends, respectively. Suppose e and f are two events. The intuitive meaning of each relation is illustrated in Figure 1. From the illustration in Figure 1, we

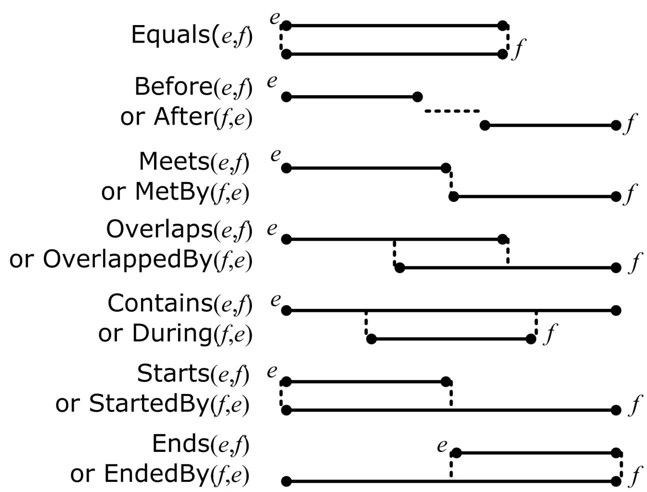

FIGURE 1. Basic relations of two events.

know that, for example, Contains(e, f) (or During(e, f)) holds iff f starts strictly after e starts, and terminates strictly before e ends. Other intuitive descriptions in English can be obtained similarly (see [9] for complete explanation).

In addition to the thirteen binary relations between events, ATL in this paper is also endowed with predicates Holds and Occurs as in [2, 9]. These predicates are useful to state that some propositions hold *all the times* or *sometime* during an event. The intuitive meanings for Holds and Occurs are explained as follows. Suppose β is a propositional formula and e is an event. The formula Holds(β, e) is true iff the formula β is satisfied *all the times* when event e happens. The formula Occurs(β, e) is a shorthand of $\neg$Holds$(\neg\beta, e)$, and thus it is true iff *there is a moment* when formula β is satisfied and event e takes place.

An atomic ATL formula has one of the following forms: one of the thirteen predicates that express binary relations between events, or a formula of the form Holds(β, e) or

$\mathsf{Occurs}(\beta, e)$, where β is a propositional formula and e is an event. A well-formed ATL formula is any boolean combination of the aforementioned atomic ATL formulas. We now illustrate the formalization of a system using ATL.

Example 3.1. *A system has to perform three sequential processes, namely* $\mathtt{P}_1$*,* $\mathtt{P}_2$*, and* $\mathtt{P}_3$*. No two processes are performed simultaneously. For each* $1 \leq i \leq 2$*, the process* $\mathtt{P}_i$ *needs to be completed before executing process* $\mathtt{P}_{i+1}$*.*

In addition to $\mathtt{P}_1$*,* $\mathtt{P}_2$*, and* $\mathtt{P}_3$*, there is a main process, denoted by* $\mathtt{P}_{main}$*, that has to be active whenever* $\mathtt{P}_1$*,* $\mathtt{P}_2$*, and* $\mathtt{P}_3$ *are performed. The process* $\mathtt{P}_{main}$ *starts before or at the same time as* $\mathtt{P}_1$*. Moreover,* $\mathtt{P}_{main}$ *and* $\mathtt{P}_3$ *terminate together.*

There is also a property denoted by `Access`*. For each* $1 \leq i \leq 3$*, when each* $\mathtt{P}_i$ *is performed, there is a moment while property* `Access` *is satisfied.*

We now show how to formalize the system in ATL. The description in the first paragraph is specified as

$$(\mathsf{Meets}(\mathtt{P}_1, \mathtt{P}_2) \vee \mathsf{Before}(\mathtt{P}_1, \mathtt{P}_2)) \wedge (\mathsf{Meets}(\mathtt{P}_2, \mathtt{P}_3) \vee \mathsf{Before}(\mathtt{P}_2, \mathtt{P}_3)).$$

The description in the second paragraph is specified as

$$(\mathsf{Starts}(\mathtt{P}_1, \mathtt{P}_{main}) \vee \mathsf{During}(\mathtt{P}_1, \mathtt{P}_{main})) \wedge \mathsf{Ends}(\mathtt{P}_3, \mathtt{P}_{main}).$$

And finally, the description in the third paragraph is specified as

$$\mathsf{Occurs}(\mathtt{Access}, \mathtt{P}_1) \wedge \mathsf{Occurs}(\mathtt{Access}, \mathtt{P}_2) \wedge \mathsf{Occurs}(\mathtt{Access}, \mathtt{P}_3).$$

3.2. **Syntax and Semantics of** ATL^+ **and** ATL_B^+**.** We now turn to formally presenting the syntax and semantics of ATL_B^+. In order to do this, a variant of ATL called ATL^+ is first defined. It uses closed intervals for the temporal interpretation of events. ATL_B^+ is a version of ATL^+ with models limited to bounded below linearly ordered sets. The language of ATL$^+$ relies on a countable set $\mathscr{P}$ of atomic propositions and a set $\mathscr{E}$ of events. We will sometimes denote an ATL$^+$ over $\mathscr{P}$ and $\mathscr{E}$ by ATL$^+(\mathscr{P},\mathscr{E})$. The formulas of ATL$^+(\mathscr{P},\mathscr{E})$ are defined recursively by the following grammar:

$$\begin{aligned}
\phi &::= \alpha \mid \neg\phi \mid \phi \vee \phi \\
\alpha &::= \mathsf{Equals}(e,f) \mid \mathsf{Before}(e,f) \mid \mathsf{Meets}(e,f) \mid \mathsf{Overlaps}(e,f) \mid \\
&\qquad \mathsf{Contains}(e,f) \mid \mathsf{Starts}(e,f) \mid \mathsf{Ends}(e,f) \mid \mathsf{Holds}(\beta,e) \\
\beta &::= p \mid \neg\beta \mid \beta \vee \beta,
\end{aligned}$$

where $p \in \mathscr{P}$ and $e, f \in \mathscr{E}$. The other propositional connectives, as well as logical constants $\top$ and $\bot$, are defined as usual. The reader should note that the syntaxes of ATL and ATL$^+$ are exactly the same (see [2, 9] for the syntax of ATL which includes Holds and Occurs predicates).

We define a β-*formula* in ATL$^+(\mathscr{P},\mathscr{E})$ as a propositional formula over the set $\mathscr{P}$. A formula is an α-*formula* or an *atomic* ATL$^+(\mathscr{P},\mathscr{E})$ formula iff it consists exactly of one formula which is a binary relation between events, or between a propositional formula and an event. An atomic formula which considers the relation After, MetBy, OverlappedBy, During, StartedBy, or EndedBy is defined using the relation Before, Meets, Overlaps, Contains, Starts, or Ends, respectively. As mentioned previously, an atomic formula of the form $\mathsf{Occurs}(\beta,e)$ is considered as a shorthand for $\neg\mathsf{Holds}(\neg\beta,e)$.

The set of all well-formed ATL$^+(\mathscr{P},\mathscr{E})$ formulas is denoted by $\mathfrak{F}\left(\mathrm{ATL}^+(\mathscr{P},\mathscr{E})\right)$ or simply $\mathfrak{F}\left(\mathrm{ATL}^+\right)$ when $\mathscr{P}$ and $\mathscr{E}$ are understood from the context. The *size* of an ATL$^+$ formula ϕ, denoted by $|\phi|$, is the number of *all boolean operators* in ϕ. For example, we have $|\neg\mathsf{Equals}(e_1,e_2) \vee \neg\mathsf{Holds}(\neg p_1,e_1)| = 4$ and
$|\neg\mathsf{Equals}(e_1,e_2) \vee \neg\mathsf{Holds}(\neg p_1 \vee \neg p_2,e_1)| = 6$. Assuming that all ATL$^+$ formulas only use $\neg$ or $\vee$ as logical operators, by structural induction on ϕ, it is not difficult to prove that the number of α-formulas (counting multiplicity) within ϕ is always one more than

the number of all conjunction operators in ϕ *by ignoring all conjunction operators that present in any β-formula within ϕ.*

In order to formalize the semantics of ATL$^+$, we first define an appropriate notion of model. The following definition is adapted from Definition 3 in [9].

Definition 3.2. An ATL$^+(\mathscr{P},\mathscr{E})$ model is a 4-tuple $\mathscr{M}=(T,<,v,\sigma)$, where $(T,<)$ is a linearly ordered set, $v: T \to 2^{\mathscr{P}}$ is a valuation map that assigns to every $t \in T$ the set of atomic propositions true on it, and $\sigma: \mathscr{E} \to \mathbb{I}(T)$ is a mapping from each event to a closed interval over T.

The *temporal interpretation* of an event $e \in \mathscr{E}$ is the interval $\sigma(e)$ which is closed and bounded. This means every event in ATL$^+$ happens within a limited time. Before we define the semantics of ATL$^+$ formula, we first define the satisfiability relation for β-formulas in ATL$^+(\mathscr{P},\mathscr{E})$.

Definition 3.3. Let $\mathscr{M}=(T,<,v,\sigma)$ be an ATL$^+(\mathscr{P},\mathscr{E})$ model and let $t \in T$. For any propositional formula β over the set $\mathscr{P}$, the satisfiability relation $\mathscr{M},t \models \beta$ is defined recursively as follows: $\mathscr{M},t \models p$ iff $p \in v(t)$, for all $p \in \mathscr{P}$; $\mathscr{M},t \models \neg\beta$ iff $\mathscr{M},t \not\models \beta$; and $\mathscr{M},t \models \beta_1 \vee \beta_2$ iff $\mathscr{M},t \models \beta_1$ *or* $\mathscr{M},t \models \beta_2$.

We are now ready to give the formal semantics of ATL$^+$ formulas. The following definition is adapted from Definition 4 in [9].

Definition 3.4. Let $\mathscr{M}=(T,<,v,\sigma)$ be an ATL$^+(\mathscr{P},\mathscr{E})$ model. The satisfiability relation $\mathscr{M} \models \phi$ for an ATL$^+(\mathscr{P},\mathscr{E})$ formula ϕ is defined recursively on the structure of ϕ as follows:

If ϕ is an α-formula, and suppose $e,f \in \mathscr{E}$ and β is a propositional formula over the set $\mathscr{A}\mathscr{P}$, then: $\mathscr{M} \models \mathsf{Equals}(e,f)$ iff $\sigma(e)=\sigma(f)$; $\mathscr{M} \models \mathsf{Before}(e,f)$ iff $\sigma(e)<\sigma(f)$;

$\mathscr{M} \models \mathsf{Meets}(e,f)$ iff there is $t \in T$ such that $\sigma(e) \cap \sigma(f) = \{t\}$ and $(\sigma(e) \smallsetminus \{t\}) < (\sigma(f) \smallsetminus \{t\})$; $\mathscr{M} \models \mathsf{Overlaps}(e,f)$ iff there is $[a,b] \in \mathbb{I}(T)$ where $a < b$ such that $\sigma(e) \cap \sigma(f) = [a,b]$, there is $t_e \in \sigma(e)$ satisfying $t_e < \sigma(f)$, and there is $t_f \in \sigma(f)$ satisfying $t_f > \sigma(e)$; $\mathscr{M} \models \mathsf{Contains}(e,f)$ iff there are $s_e, t_e \in \sigma(e)$ such that $s_e < \sigma(f) < t_e$; $\mathscr{M} \models \mathsf{Starts}(e,f)$ iff $\sigma(e) \subset \sigma(f)$, there is $t_f \in \sigma(f)$ such that $t_f > \sigma(e)$, but there is no $t_f \in \sigma(f)$ such that $t_f < \sigma(e)$; $\mathscr{M} \models \mathsf{Ends}(e,f)$ iff $\sigma(e) \subset \sigma(f)$, there is $t_f \in \sigma(f)$ such that $t_f < \sigma(e)$, but there is no $t_f \in \sigma(f)$ such that $t_f > \sigma(e)$; and $\mathscr{M} \models \mathsf{Holds}(\beta, e)$ iff $\mathscr{M}, t \models \beta$, for all $t \in \sigma(e)$, where $\mathscr{M}, t \models \beta$ is the satisfiability relation in Definition 3.3;

If ϕ is of the form $\neg\phi_1$, then $\mathscr{M} \models \neg\phi_1$ iff $\mathscr{M} \not\models \phi_1$;

If ϕ is of the form $\phi_1 \vee \phi_2$, then $\mathscr{M} \models \phi_1 \vee \phi_2$ iff $\mathscr{M} \models \phi_1$ *or* $\mathscr{M} \models \phi_2$.

The reader should verify that in Definition 3.4, the satisfiability relations for atomic formulas with the predicates Meets, Before, and Overlaps for ATL$^+$ are slightly different to those for ATL in [9]. However, this difference does not affect the mutual exclusiveness of the thirteen possible binary relations between events. This means that, for any pair (e,f) of event, there is exactly one relation R among $\{\mathsf{Equals}, \mathsf{Before}, \ldots, \mathsf{EndedBy}\}$ such that $\mathscr{M} \models \mathsf{R}(e,f)$.

We define ATL^+_B as a version of ATL$^+$ whose models are limited to bounded below linearly ordered sets. Thus, any 4-tuple $\mathscr{M} = (T, <, v, \sigma)$ in Definition 3.2 is an ATL^+_B model iff $(T, \leq)$ has a minimal element. The set of all well-formed ATL^+_B formulas is denoted by $\mathfrak{F}\left(\mathrm{ATL}^+_B(\mathscr{P}, \mathscr{E})\right)$ or simply $\mathfrak{F}\left(\mathrm{ATL}^+_B\right)$ when $\mathscr{P}$ and $\mathscr{E}$ are understood from the context.

4. Translation from ATL_B^+ to HRPNL

A translation between two temporal logics is a satisfiability-preserving mapping between them. Consider two temporal logics L_1 and L_2 whose sets of all well-formed formulas are denoted by $\mathfrak{F}(\mathrm{L}_1)$ and $\mathfrak{F}(\mathrm{L}_2)$, respectively. In general, a translation from L_1 to L_2 is a mapping $\tau : \mathfrak{F}(\mathrm{L}_1) \to \mathfrak{F}(\mathrm{L}_2)$ satisfying the following condition: a formula $\varphi \in \mathfrak{F}(\mathrm{L}_1)$ is satisfiable in L_1 iff $\tau(\varphi) \in \mathfrak{F}(\mathrm{L}_2)$ is satisfiable in L_2 [7].

To construct a translation from ATL_B^+ to HRPNL, first observe that HRPNL relies solely on a set of atomic propositions, whereas ATL_B^+ relies on a set of propositional atoms and a set of events. Hence, in order to establish a semantic relationship between them, we have to add a syntactic representation of each event in ATL_B^+ to HRPNL. We adapt the techniques from [9] and add an atomic propositions p_e to the syntax of HRPNL for each event e in ATL_B^+. Every event e in ATL_B^+ is associated with precisely one atomic proposition p_e. We denote the set of all atomic propositions p_e by $\mathscr{P}_{\mathscr{E}}$. Given an ATL_B^+ which relies on a set $\mathscr{P}$ of atomic propositions and a set $\mathscr{E}$ of events, the corresponding HRPNL relies exclusively on the set $\mathscr{P} \cup \mathscr{P}_{\mathscr{E}}$ of propositional atoms. From now on, we abbreviate $\mathscr{P} \cup \mathscr{P}_{\mathscr{E}}$ by $\mathscr{P}\mathscr{E}$.

4.1. **Construction of** HRPNL **Models.** Suppose $\mathscr{M} = (T, <, v, \sigma)$ is an $\mathrm{ATL}_B^+(\mathscr{P}, \mathscr{E})$ model over $(T, \leq)$. The corresponding $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ model is a pair $M = (\mathbb{I}(T), V)$, for a particular valuation function V. The function V must be able to represent the truth of each $p \in \mathscr{P} \subset \mathscr{P}\mathscr{E}$ and the temporal interpretation of each $e \in \mathscr{E}$. One way to do this is by defining V such that for any $t \in T$ we have: $p \in V(t)$ iff $p \in v(t)$, for all $p \in \mathscr{P}$; and $p_e \in V(t)$ iff $t \in \sigma(e)$, for all $e \in \mathscr{E}$ and $p_e \in \mathscr{P}_{\mathscr{E}}$.

Lemma 4.1. *Let* $\mathscr{M} = (T, <, v, \sigma)$ *be an* $\mathrm{ATL}_B^+(\mathscr{P}, \mathscr{E})$ *model over* $(T, \leq)$ *and* $M = (\mathbb{I}(T), V)$ *be an* $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ *model with valuation function* V *such that for each* $t \in T$*:* $p \in V(t)$ *iff*

$p \in v(t)$ for all $p \in \mathscr{P} \subset \mathscr{P}\mathscr{E}$; and $p_e \in V(t)$ iff $t \in \sigma(e)$ for all $e \in \mathscr{E}$ and $p_e \in \mathscr{P}_{\mathscr{E}} \subset \mathscr{P}\mathscr{E}$. Then for all $t \in D$ and $[t_1,t_2] \in \mathbb{I}(T)$, we have: $M,[t,t] \Vdash p$ iff $\mathscr{M},t \models p$, for all $p \in \mathscr{P}$; and $M,[t_1,t_2] \Vdash p_e$ iff $[t_1,t_2] \subseteq \sigma(e)$, for all $p_e \in \mathscr{P}_{\mathscr{E}}$ and $e \in \mathscr{E}$.

Proof. The proof follows immediately from Definitions 2.4, 3.2, 3.3, and 3.4. Observe that for any $p \in \mathscr{P} \subset \mathscr{P}\mathscr{E}$ and $t \in T$, and we have: $M,[t,t] \Vdash p$ iff $p \in V(t)$ iff $p \in v(t)$ iff $\mathscr{M},t \models p$. For any $p_e \in \mathscr{P}_{\mathscr{E}} \subset \mathscr{P}\mathscr{E}$ and $[t_1,t_2] \in \mathbb{I}(T)$ we have: $M,[t_1,t_2] \Vdash p_e$ iff $p_e \in V(t)$ for all $t \in [t_1,t_2]$, $p_e \in V(t)$ for all $t \in [t_1,t_2]$ iff $t \in \sigma(e)$ for all $t \in [t_1,t_2]$, and $t \in \sigma(e)$ for all $t \in [t_1,t_2]$ iff $[t_1,t_2] \subseteq \sigma(e)$. Thus, $M.[t_1,t_2] \Vdash p_e$ iff $[t_1,t_2] \subseteq \sigma(e)$. □

In any model $\mathscr{M}$ of $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$, the temporal interpretation of each event $e \in \mathscr{E}$ is a closed interval over $(T,\leq)$. As a consequence, in the corresponding model M of $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$, every atomic proposition p_e has to be satisfied precisely in one interval. We adapt following definition from Definition 5 in [9].

Definition 4.2. For every $p_e \in \mathscr{P}_{\mathscr{E}}$, we define ψ_e as a formula

$$\Diamond_r \Diamond_r p_e \wedge \neg \Diamond_r \Diamond_r (p_e \wedge \Diamond_r (\neg p_e \wedge \Diamond_r p_e)) \wedge \neg \Diamond_r \Box_r p_e$$

and $\Psi_{\mathscr{E}}$ as the set of all formulas of the form ψ_e where $e \in \mathscr{E}$. In addition, for an $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ model M, we write $M,[t_1,t_2] \Vdash \Psi_{\mathscr{E}}$ iff $M,[t_1,t_2] \Vdash \psi_e$ for all $\psi_e \in \Psi_{\mathscr{E}}$.

Intuitively, the first conjunct of ψ_e, i.e. $\Diamond_r \Diamond_r p_e$, expresses that the temporal representation of event e in $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ is non-empty. The second conjunct of ψ_e, i.e. $\neg \Diamond_r \Diamond_r (p_e \wedge \Diamond_r (\neg p_e \wedge \Diamond_r p_e))$, ensures the convexity of the temporal representation of event e in $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$. The third conjunct of ψ_e, i.e. $\neg \Diamond_r \Box_r p_e$, describes that event e happens within a limited time.

Lemma 4.3. *Let $M = (\mathbb{I}(T), V)$ be an $\mathrm{HRPNL}_{\mathscr{P}_\mathscr{E}}$ model over $(T, \leq)$ and let t_0 be the minimal element of T. Every atomic proposition $p_e \in \mathscr{P}_\mathscr{E}$ holds precisely in one interval iff $M, [t_0, t_0] \Vdash \Psi_\mathscr{E}$.*

Proof. By Definition 4.2, $M, [t_0, t_0] \Vdash \Psi_\mathscr{E}$ iff the following satisfiability relations hold for all $p_e \in \mathscr{P}_\mathscr{E}$:

$$M, [t_0, t_0] \Vdash \Diamond_r \Diamond_r p_e \tag{1}$$

$$M, [t_0, t_0] \Vdash \neg \Diamond_r \Diamond_r (p_e \wedge \Diamond_r (\neg p_e \wedge \Diamond_r p_e)) \tag{2}$$

$$M, [t_0, t_0] \Vdash \neg \Diamond_r \Box_r p_e. \tag{3}$$

The satisfiability relation (1) holds iff there exists $[t_1, t_2] \in \mathbb{I}(T)$ such that $M, [t_1, t_2] \Vdash p_e$. Consequently, (1) holds iff p_e is *at least satisfied in one interval*.

The satisfiability relation $M, [t_0, t_0] \Vdash \Diamond_r \Diamond_r (p_e \wedge \Diamond_r (\neg p_e \wedge \Diamond_r p_e))$ holds iff there is $[t_1, t_2] \in \mathbb{I}(T)$ satisfying $M, [t_1, t_2] \Vdash p_e \wedge \Diamond_r (\neg p_e \wedge \Diamond_r p_e)$, which holds iff there are $[t_1, t_2], [t_2, t_3], [t_3, t_4] \in \mathbb{I}(T)$ with $t_1 \leq t_2 \leq t_3 \leq t_4$ such that $M, [t_1, t_2] \Vdash p_e$ and $M, [t_3, t_4] \Vdash p_e$, but $M, [t_2, t_3] \nVdash p_e$. In other words, $M, [t_1, t_2] \Vdash p_e \wedge \Diamond_r (\neg p_e \wedge \Diamond_r p_e)$ iff there are two disjoint intervals $[t_1, t_2]$ and $[t_3, t_4]$ where p_e holds. Thus (2) holds iff p_e is *never satisfied in more than one interval*. By Lemma 4.1, these conditions mean that the set $\sigma(e)$ is convex.

The satisfiability relation $M, [t_0, t_0] \Vdash \Diamond_r \Box_r p_e$ holds iff there is $t_1 \geq t_0$ satisfying $M, [t_0, t_1] \Vdash \Box_r p_e$, which holds iff p_e is satisfied in *all* intervals of the form $[t_1, t_2]$ where $t_2 \geq t_1$. In other words, $M, [t_0, t_1] \Vdash \Box_r p_e$ iff p_e holds in an *unbounded interval*. Since $(T, \leq)$ is bounded below by t_0, any interval where p_e holds is also bounded below. Therefore (3) holds iff p_e is *never satisfied in an unbounded interval*.

Based on explanation given, we conclude that any $p_e \in \mathscr{P}_\mathscr{E}$ holds exactly in one (closed) interval iff $M, [t_0, t_0] \Vdash \Psi_\mathscr{E}$. □

In the following lemma, we show that there is a bijection between $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$ models and $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ models that satisfy $M,[t_0,t_0] \Vdash \Psi_{\mathscr{E}}$, where t_0 is the minimal element of the linearly ordered set.

Lemma 4.4. *Given a bounded below linearly ordered set $(T,\leq)$ with minimal element t_0, a model $M = (\mathbb{I}(T),V)$ of $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ that satisfy $M,[t_0,t_0] \Vdash \Psi_{\mathscr{E}}$ can be constructed uniquely from a model $\mathscr{M} = (T,<,v,\sigma)$ of $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$, and vice versa.*

Proof. Consider a model $\mathscr{M} = (T,<,v,\sigma)$ of ATL_B^+, where $v : T \to 2^{\mathscr{P}}$ and $\sigma : \mathscr{E} \to \mathbb{I}(T)$. We build a model $M = (\mathbb{I}(T),V)$ of $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ with valuation function V as explained in Lemma 4.1. Since each $e \in \mathscr{E}$ holds precisely in one interval, by Lemma 4.3 we have $M,[t_0,t_0] \Vdash \Psi_{\mathscr{E}}$.

Conversely, suppose $M = (\mathbb{I}(T),V)$ is an $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ model that satisfy $M,[t_0,t_0] \Vdash \Psi_{\mathscr{E}}$. We construct a 4-tuple $(T,<,v,\sigma)$ where: $v : T \to 2^{\mathscr{P}}$, with $p \in v(t)$ iff $p \in V(t)$ for all $t \in T$ and $p \in \mathscr{P} \subset \mathscr{P}\mathscr{E}$; and $\sigma : \mathscr{E} \to 2^T$, with $t \in \sigma(e)$ iff $p_e \in V(t)$, for all $t \in v$ and $p_e \in \mathscr{P}_{\mathscr{E}} \subset \mathscr{P}\mathscr{E}$. Since $M,[t_0,t_0] \Vdash \Psi_{\mathscr{E}}$, by Lemma 4.3, each atomic proposition p_e holds precisely in one interval. Therefore, the map σ associates an event to an interval, and thus $\sigma(\mathscr{E}) \subseteq \mathbb{I}(T)$. We conclude that $\mathscr{M} = (T,<,v,\sigma)$ is a well-defined model of $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$. □

Lemma 4.4 also ensures that any ATL_B^+ model has a unique corresponding $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ model.

Definition 4.5. Given a model $\mathscr{M} = (T,<,v,\sigma)$ of $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$, the corresponding $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ model for $\mathscr{M}$, denoted by $M_{\mathscr{M}}$, is a pair $(\mathbb{I}(T),V)$ that satisfies $M,[t_0,t_0] \Vdash \Psi_{\mathscr{E}}$, where V is the valuation function defined as in Lemma 4.1 and $\Psi_{\mathscr{E}}$ is the set of formulas in Definition 4.2.

4.2. **Translation Construction.** We see that the truth of any ATL_B^+ formula is evaluated in a model $\mathscr{M}$, whereas the truth of an HRPNL formula is evaluated at an *interval within a model* M. In order to represent the occurrences of all events over $(T,\leq)$, HRPNL formulas are evaluated at $[t_0,t_0]$, where t_0 is the minimal element of T. The occurrence of each event in any interval over $(T,\leq)$ is captured in HRPNL using the right neighborhood modality $\Diamond_r$. We can now define formally the notion of a translation from ATL_B^+ to HRPNL.

Definition 4.6. A translation from $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$ to $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ is a mapping

$$\tau : \mathfrak{F}\left(\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})\right) \to \mathfrak{F}\left(\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}\right)$$

such that any formula $\phi \in \mathfrak{F}\left(\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})\right)$ satisfies $\mathscr{M} \models \phi$ *iff* $M_{\mathscr{M}},[t_0,t_0] \Vdash \tau(\phi)$, where $M_{\mathscr{M}}$ is the corresponding $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ model for $\mathscr{M}$ in Definition 4.5.

We adapt the translation rule from ALTL for LTL in [9] in our translation from ATL_B^+ to HRPNL. Recall that every ATL_B^+ formula is either an α-formula or a boolean combination of some α-formulas. Therefore, in order to construct a translation from ATL_B^+ to HRPNL, we need to first translate each possible form of an α-formula into an equisatisfiable HRPNL formula.

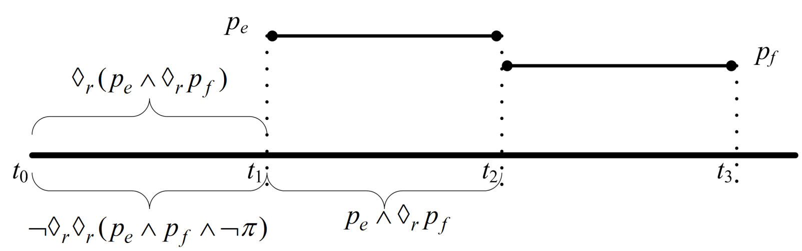

FIGURE 2. Translation of $\mathsf{Meets}(e,f)$.

For example, suppose $\mathscr{M}$ is an ATL_B^+ model over $(T,\leq)$ whose minimal element is t_0. In Figure 2, we give an illustration of how an α-formula of the form $\mathsf{Meets}(e,f)$ is translated into an equisatisfiable HRPNL formula. In the corresponding model $M_{\mathscr{M}}$ of HRPNL, the events e and f are presented using propositional atoms p_e and p_f, respectively. From Definition 3.4 and Lemma 4.1, we infer that there are $[t_1,t_2],[t_2,t_3] \in \mathbb{I}(T)$ such that $M_{\mathscr{M}},[t_1,t_2] \Vdash p_e$ and $M_{\mathscr{M}},[t_2,t_3] \Vdash p_f$. Moreover, since there is no nonpoint subinterval of $[t_1,t_3]$ satisfying $p_e \wedge p_f$, we have $M_{\mathscr{M}},[t_0,t_1] \Vdash \neg\Diamond_r\Diamond_r\left(p_e \wedge p_f \wedge \neg\pi\right)$. Consequently, we obtain $M_{\mathscr{M}},[t_0,t_1] \Vdash \Diamond_r\left(p_e \wedge \Diamond_r p_f\right) \wedge \neg\Diamond_r\Diamond_r\left(p_e \wedge p_f \wedge \neg\pi\right)$. Since (by Definition 4.6) the resulting HRPNL formula has to be evaluated at interval $[t_0,t_0]$, we conclude that $\mathsf{Meets}(e,f)$ is translated into $\Diamond_r\left(\Diamond_r\left(p_e \wedge \Diamond_r p_f\right) \wedge \neg\Diamond_r\Diamond_r\left(p_e \wedge p_f \wedge \neg\pi\right)\right)$.

Definition 4.7. Let $\mathfrak{F}\left(\mathrm{ATL}_B^+\right)$ and $\mathfrak{F}(\mathrm{HRPNL})$ be two sets of all well-formed formulas of $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$ and $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$, respectively. The function tr is a mapping from $\mathfrak{F}\left(\mathrm{ATL}_B^+\right)$ to $\mathfrak{F}(\mathrm{HRPNL})$ defined recursively on the structure of $\phi \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$ as follows:

If ϕ is an α-formula, $e,f \in \mathscr{E}$, and β is a propositional formula over $\mathscr{P}$, then

$$
\begin{aligned}
\mathrm{tr}[\mathsf{Equals}(e,f)] &:= \Box_r\Box_r\left(p_e \leftrightarrow p_f\right), \\
\mathrm{tr}[\mathsf{Before}(e,f)] &:= \Diamond_r\Diamond_r\left(p_e \wedge \Diamond_r\left(\neg p_e \wedge \neg p_f \wedge \Diamond_r p_f\right)\right), \\
\mathrm{tr}[\mathsf{Meets}(e,f)] &:= \Diamond_r\left(\Diamond_r\left(p_e \wedge \Diamond_r p_f\right) \wedge \neg\Diamond_r\Diamond_r\left(p_e \wedge p_f \wedge \neg\pi\right)\right), \\
\mathrm{tr}[\mathsf{Overlaps}(e,f)] &:= \Diamond_r\Diamond_r\left(p_e \wedge \neg p_f \wedge \Diamond_r\left(p_e \wedge p_f \wedge \neg\pi \wedge \Diamond_r\left(\neg p_e \wedge p_f\right)\right)\right), \\
\mathrm{tr}[\mathsf{Contains}(e,f)] &:= \Diamond_r\Diamond_r\left(p_e \wedge \neg p_f \wedge \Diamond_r\left(p_e \wedge p_f \wedge \Diamond_r\left(p_e \wedge \neg p_f\right)\right)\right), \\
\mathrm{tr}[\mathsf{Starts}(e,f)] &:= \Box_r\Box_r\left(p_e \to p_f\right) \wedge \Diamond_r\Diamond_r\left(\neg p_e \wedge p_f\right) \\
&\quad\ \wedge\neg\Diamond_r\Diamond_r\left(\neg p_e \wedge p_f \wedge \Diamond_r p_e\right), \\
\mathrm{tr}[\mathsf{Ends}(e,f)] &:= \Box_r\Box_r\left(p_e \to p_f\right) \wedge \Diamond_r\Diamond_r\left(\neg p_e \wedge p_f\right) \\
&\quad\ \wedge\neg\Diamond_r\Diamond_r\left(p_e \wedge \Diamond_r\left(\neg p_e \wedge p_f\right)\right), \\
\mathrm{tr}[\mathsf{Holds}(\beta,e)] &:= \Box_r\Box_r(p_e \to \beta).
\end{aligned}
$$

If ϕ is of the form $\neg\phi_1$, then $\mathrm{tr}[\neg\phi_1] := \neg\mathrm{tr}[\phi_1]$.

If ϕ is of the form $\phi_1 \vee \phi_2$, then $\mathrm{tr}[\phi_1 \vee \phi_2] := \mathrm{tr}[\phi_1] \vee \mathrm{tr}[\phi_2]$.

We now illustrate how the function tr is applied to a particular ATL_B^+ formula.

Example 4.8. Let ϕ be the following $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$ formula

$$\mathsf{Equals}(e,f) \vee (\mathsf{Meets}(f,g) \wedge \mathsf{Before}(f,g)),$$

for some $e,f,g \in \mathscr{E}$. Suppose p_e, p_f, and p_g are the corresponding propositional atoms for e, f, and g, respectively. The resulting $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ formula $\mathrm{tr}[\phi]$ is computed by first using the following equalities: $\mathrm{tr}[\neg\phi_1] = \neg\mathrm{tr}[\phi_1]$, $\mathrm{tr}[\phi_1 \vee \phi_2] = \mathrm{tr}[\phi_1] \vee \mathrm{tr}[\phi_2]$, and

$$\begin{aligned}\mathrm{tr}[\phi_1 \wedge \phi_2] = \mathrm{tr}[\neg(\neg\phi_1 \vee \neg\phi_2)] &= \neg\mathrm{tr}[\neg\phi_1 \vee \neg\phi_2]\\ &= \neg(\mathrm{tr}[\neg\phi_1] \vee \mathrm{tr}[\neg\phi_2]) = \neg(\neg\mathrm{tr}[\phi_1] \vee \neg\mathrm{tr}[\phi_2])\\ &= \mathrm{tr}[\phi_1] \wedge \mathrm{tr}[\phi_2].\end{aligned}$$

Thus, we have

$$\begin{aligned}&\mathrm{tr}[\mathsf{Equals}(e,f) \wedge (\mathsf{Meets}(f,g) \vee \mathsf{Before}(f,g))] =\\ &\mathrm{tr}[\mathsf{Equals}(e,f)] \wedge (\mathrm{tr}[\mathsf{Meets}(f,g)] \vee \mathrm{tr}[\mathsf{Before}(f,g)]).\end{aligned}$$

Finally, we translate each of the three α-formulas. Therefore, $\mathrm{tr}[\phi]$ is equivalent to

$$\Box_r\Box_r(p_e \leftrightarrow p_f) \wedge \left(\begin{array}{l}\Diamond_r(\Diamond_r(p_e \wedge p_f) \wedge \neg\Diamond_r\Diamond_r(p_e \wedge p_f \wedge \pi))\\ \vee\Diamond_r\Diamond_r(p_e \wedge \Diamond_r(\neg p_e \wedge \neg p_f \wedge \Diamond_r p_f))\end{array}\right).$$

We now prove that the function tr in Definition 4.7 is a translation from $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$ to $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ with respect to Definition 4.6.

Theorem 4.9. *Let $\mathscr{M}$ be an $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$ model over $(T,\leq)$ whose minimal element is t_0. For any $\mathrm{ATL}_B^+(\mathscr{P},\mathscr{E})$ formula ϕ, the function* tr *in Definition 4.7 satisfies $\mathscr{M} \models \phi$ iff $M_{\mathscr{M}},[t_0,t_0] \Vdash \mathrm{tr}[\phi]$, where $M_{\mathscr{M}}$ is the corresponding $\mathrm{HRPNL}_{\mathscr{P}\mathscr{E}}$ model for $\mathscr{M}$.*

Proof. We prove the theorem by induction on the structure of $\phi \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$. Recall that ϕ is expressible in one of the following forms: an α-formula, a formula of the form $\neg\phi_1$, or a formula of the form $\phi_1 \vee \phi_2$, for some $\phi_1, \phi_2 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$.

Induction basis: Suppose ϕ is an α-formula, then there are eight different forms of ϕ. We discuss only one of them, namely $\mathsf{Before}(e,f)$, for some $e,f \in \mathscr{E}$. We will show that $\mathscr{M} \models \mathsf{Before}(e,f)$ iff $M_{\mathscr{M}}, [t_0,t_0] \Vdash \Diamond_r \Diamond_r \left(p_e \wedge \Diamond_r \left(\neg p_e \wedge \neg p_f \wedge \Diamond_r p_f\right)\right)$.

"*Only if*": By Definition 3.4, $\mathscr{M} \models \mathsf{Before}(e,f)$ iff $\sigma(e) < \sigma(f)$. We deduce that there are $[t_1,t_2], [t_3,t_4] \in \mathbb{I}(T)$ where $t_2 < t_3$ such that $[t_1,t_2] \subseteq \sigma(e)$ and $[t_3,t_4] \subseteq \sigma(f)$. Moreover, we also have $[t_2,t_3] \nsubseteq \sigma(e)$ and $[t_2,t_3] \nsubseteq \sigma(f)$. Hence by Lemma 4.1, we obtain

$$M_{\mathscr{M}}, [t_1,t_2] \Vdash p_e \tag{4}$$

$$M_{\mathscr{M}}, [t_2,t_3] \Vdash \neg p_e \wedge \neg p_f \tag{5}$$

$$M_{\mathscr{M}}, [t_3,t_4] \Vdash p_e. \tag{6}$$

Observe that these satisfiability relations imply $t_2 < t_3$, since, if $t_2 = t_3$, then we get $p_e \in V(t_2)$ from (4) and we get $p_e \notin V(t_2)$ from (5), which is a contradiction. Thus we infer that $M_{\mathscr{M}}, [t_1,t_2] \Vdash p_e \wedge \Diamond_r \left(\neg p_e \wedge \neg p_f \wedge \Diamond_r p_f\right)$, and so the result follows.

"*If*": Observe that $M_{\mathscr{M}}, [t_0,t_0] \Vdash \Diamond_r \Diamond_r \left(p_e \wedge \Diamond_r \left(\neg p_e \wedge \neg p_f \wedge \Diamond_r p_f\right)\right)$ iff there is $[t_1,t_2] \in \mathbb{I}(T)$ such that $M_{\mathscr{M}}, [t_1,t_2] \Vdash p_e \wedge \Diamond_r \left(\neg p_e \wedge \neg p_f \wedge \Diamond_r p_f\right)$. This implies that there are $[t_1,t_2], [t_2,t_3], [t_3,t_4] \in \mathbb{I}(T)$ such that $M_{\mathscr{M}}, [t_1,t_2] \Vdash p_e$, $M_{\mathscr{M}}, [t_2,t_3] \Vdash \neg p_e \wedge \neg p_f$, and $M_{\mathscr{M}}, [t_3,t_4] \Vdash p_f$. By using the same argument as before, we infer that $t_2 < t_3$. Moreover, by Lemma 4.1 and 4.3, we deduce that $[t_1,t_2] \subseteq \sigma(e)$, $[t_3,t_4] \subseteq \sigma(f)$, and $[t_2,t_3] \nsubseteq \sigma(e) \cup \sigma(f)$. Since $t_2 < t_3$, we conclude that $\sigma(e) < \sigma(f)$, and thus $\mathscr{M} \models \mathsf{Before}(e,f)$.

The proof for other forms of α-formulas can be established analogously.

Induction step: As an induction hypothesis, we assume that the theorem holds for $\phi_1, \phi_2 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$. Consider two following cases:

Case 1. If ϕ has the form $\neg\phi_1$, for some $\phi_1 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$, then

$$
\begin{aligned}
\mathscr{M} \models \neg\phi_1 \quad &\text{iff} \quad \mathscr{M} \not\models \phi_1 && \text{(by Definition 3.4)}\\
&\text{iff} \quad M_{\mathscr{M}}, [t_0, t_0] \not\Vdash \mathrm{tr}[\phi_1] && \text{(by induction hypothesis)}\\
&\text{iff} \quad M_{\mathscr{M}}, [t_0, t_0] \Vdash \neg\,\mathrm{tr}[\phi_1] && \text{(by Definition 2.4)}
\end{aligned}
$$

Case 2. If ϕ has the form $\phi_1 \vee \phi_2$ for some $\phi_1, \phi_2 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$, then

$$
\begin{aligned}
\mathscr{M} \models \phi_1 \vee \phi_2 \quad &\text{iff} \quad \mathscr{M} \models \phi_1 \textit{ or } \mathscr{M} \models \phi_2 && \text{(by Definition 3.4)}\\
&\text{iff} \quad M_{\mathscr{M}}, [t_0, t_0] \Vdash \mathrm{tr}[\phi_1] \textit{ or} &&\\
&\qquad\;\; M_{\mathscr{M}}, [t_0, t_0] \Vdash \mathrm{tr}[\phi_2] && \text{(by induction hypothesis)}\\
&\text{iff} \quad M_{\mathscr{M}}, [t_0, t_0] \Vdash \mathrm{tr}[\phi_1] \vee \mathrm{tr}[\phi_2] && \text{(by Definition 2.4)}
\end{aligned}
$$

This completes the proof. □

In Example 4.8, we see that $\mathrm{tr}[\phi_1 \wedge \phi_2] = \mathrm{tr}[\phi_1] \wedge \mathrm{tr}[\phi_2]$. Moreover, it is easy to check that $\mathrm{tr}[\phi_1 \to \phi_2] = \mathrm{tr}[\phi_1] \to \mathrm{tr}[\phi_2]$ and $\mathrm{tr}[\phi_1 \leftrightarrow \phi_2] = \mathrm{tr}[\phi_1] \leftrightarrow \mathrm{tr}[\phi_2]$.

4.3. **Translation Algorithm.** The translation algorithm from ATL_B^+ to HRPNL is constructed straightforwardly from the function tr in Definition 4.7. We restrict the inputs of the algorithm to well-formed ATL_B^+ formulas that only use $\neg$ or $\vee$ as logical operators. The logical connectives other that $\neg$ or $\vee$ are considered as abbreviations.

It is clear that Algorithm 4.1 terminates for any input since every well-formed ATL_B^+ formula has finitely many operators. The running time of this algorithm on an input formula ϕ depends on the number of invocations of tr on ϕ, which is related to the size of ϕ. We define $N_\neg(\phi)$ and $N_\vee(\phi)$ as the number of *all* negation and disjunction operators in ϕ, respectively. By assuming that all inputs only use $\neg$ or $\vee$ as logical operators, we have $|\phi| = N_\neg(\phi) + N_\vee(\phi)$.

Algorithm 4.1 function tr$[\phi]$

precondition: ϕ is a well-formed ATL_B^+ formula which is an α-formula, $\neg\phi_1$, or $\phi_1 \vee \phi_2$, for some $\phi_1, \phi_2 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$.

postcondition: $\mathrm{tr}[\phi]$ computes an equisatisfiable HRPNL formula for ϕ.

begin function

  case

  ϕ is an α-formula: **return** $\mathrm{tr}[\phi]$ directly according to Definition 4.7

  ϕ is $\neg\phi_1$: **return** $\neg\mathrm{tr}[\phi_1]$

  ϕ is $\phi_1 \vee \phi_2$: **return** $\mathrm{tr}[\phi_1] \vee \mathrm{tr}[\phi_2]$

  end case

end function

For an ATL_B^+ formula ϕ, we define $n_\neg(\phi)$ as the number of all negation operators in ϕ *by ignoring all negation operators that present in any β-formula within ϕ*. The notation $n_\vee(\phi)$ is defined analogously for disjunction operator. It is obvious that $n_\neg(\phi) \leq N_\neg(\phi)$ and $n_\vee(\phi) \leq N_\vee(\phi)$ with equalities hold iff ϕ contains no α-formula of the forms $\mathsf{Holds}(\beta, e)$ or $\mathsf{Occurs}(\beta, e)$. For example, if ϕ is $\neg(\neg\mathsf{Holds}(\neg p_1, e_1) \vee \neg(\mathsf{Holds}(\neg p_1, e_1) \vee \mathsf{Holds}(\neg p_1 \vee \neg p_2, e_2)$ then $n_\neg(\phi) = 3$, $N_\neg(\phi) = 7$, $n_\vee(\phi) = 2$, and $N_\vee(\phi) = 3$.

The number of calls to tr performed by Algorithm 4.1 is given in the following lemma.

Lemma 4.10. *The total number of invocations of* tr *on any input formula ϕ in Algorithm 4.1 is equal to* $1 + n_\neg(\phi) + 2n_\vee(\phi)$.

Proof. We prove the lemma by structural induction on ϕ. Let $n_{\mathrm{tr}}(\phi)$ be the number of calls to tr (including the recursive ones) performed by Algorithm 4.1 on ϕ. We will show that $n_{\mathrm{tr}}(\phi) = 1 + n_\neg(\phi) + 2n_\vee(\phi)$.

Induction basis: Suppose ϕ is an α-formula. It follows that $n_{\mathrm{tr}}(\phi) = 1$, $n_{\neg}(\phi) = n_{\vee}(\phi) = 0$, and thus the lemma holds.

Induction step: As an induction hypothesis, we assume that the lemma holds for $\phi_1, \phi_2 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$. Consider two following cases:

Case 1. Suppose ϕ has the form $\neg\phi_1$, for some $\phi_1 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$. By inspecting Algorithm 4.1, we get $n_{\mathrm{tr}}(\neg\phi_1) = 1 + n_{\mathrm{tr}}(\phi_1)$. Since $n_{\neg}(\neg\phi_1) = 1 + n_{\neg}(\phi_1)$ and $n_{\vee}(\neg\phi_1) = n_{\vee}(\phi_1)$, we have

$$\begin{aligned} n_{\mathrm{tr}}(\neg\phi_1) &= 1 + n_{\mathrm{tr}}(\phi_1) = 1 + (1 + n_{\neg}(\phi_1) + 2n_{\vee}(\phi_1)) \\ &= 1 + n_{\neg}(\neg\phi_1) + 2n_{\vee}(\neg\phi_1). \end{aligned}$$

Case 2. Suppose ϕ has the form $\phi_1 \vee \phi_2$, for some $\phi_1, \phi_2 \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$. From Algorithm 4.1, we get $n_{\mathrm{tr}}(\phi_1 \vee \phi_2) = 1 + n_{\mathrm{tr}}(\phi_1) + n_{\mathrm{tr}}(\phi_2)$. Since $n_{\neg}(\phi_1 \vee \phi_2) = n_{\neg}(\phi_1) + n_{\neg}(\phi_2)$ and $n_{\vee}(\phi_1 \vee \phi_2) = 1 + n_{\vee}(\phi_1) + n_{\vee}(\phi_2)$, we obtain

$$\begin{aligned} n_{\mathrm{tr}}(\phi_1 \vee \phi_2) &= 1 + n_{\mathrm{tr}}(\phi_1) + n_{\mathrm{tr}}(\phi_2) \\ &= 1 + (1 + n_{\neg}(\phi_1) + 2n_{\vee}(\phi_1)) + (1 + n_{\neg}(\phi_2) + 2n_{\vee}(\phi_2)) \\ &= 1 + (n_{\neg}(\phi_1) + n_{\neg}(\phi_2)) + 2(1 + n_{\vee}(\phi_1) + n_{\vee}(\phi_2)) \\ &= 1 + n_{\neg}(\phi_1 \vee \phi_2) + 2n_{\vee}(\phi_1 \vee \phi_2). \end{aligned}$$

This completes the proof. $\square$

We can now measure the complexity of our translation algorithm with respect to the size of the input formula.

Theorem 4.11. *The complexity of Algorithm 4.1 is linear in terms of the size of an input formula.*

Proof. Suppose each call to tr in Algorithm 4.1 takes time k, where $k > 0$. By Lemma 4.10, we deduce that the running time of Algorithm 4.1 satisfies $k(1+n_{\neg}(\phi)+2n_{\vee}(\phi)) \leq k(1+N_{\neg}(\phi)+2N_{\vee}(\phi)) \leq 2k(N_{\neg}(\phi)+N_{\vee}(\phi)) = 2k|\phi|$, which is $\mathscr{O}(|\phi|)$. $\square$

4.4. **Size Propagation of Translated Formula.** From Example 4.8, we see that the translation makes the size of a resulting HRPNL formula is larger (i.e. contains more operators) than its original size. Furthermore, by a straightforward induction on the structure of ϕ, we obtain $|\mathrm{tr}[\phi]| > |\phi|$ for any $\phi \in \mathfrak{F}\left(\mathrm{ATL}_B^+\right)$. The size propagation of an ATL_B^+ formula is defined as the size of $\mathrm{tr}[\phi]$. We will express $|\mathrm{tr}[\phi]|$ in terms of $|\phi|$.

To simplify our analysis, we write α-formulas of the form $\mathsf{Equals}(e,f), \ldots, \mathsf{Ends}(e,f)$, and $\mathsf{Holds}(\beta,e)$ with $\mathsf{R}_1(e,f), \ldots, \mathsf{R}_7(e,f)$, and $\mathsf{R}_8(\beta,e)$, respectively. It is clear that the size propagation of any α-formula of the same form is always equal. For each $1 \leq i \leq 8$, we denote the *size propagation* for α-formulas of the form $\mathsf{R}_i(\ldots,\ldots)$ by m_i. Given an ATL_B^+ formula ϕ, we define $n_i(\phi)$ as the number of α-formulas of the form $\mathsf{R}_i(\ldots,\ldots)$ within ϕ, where $1 \leq i \leq 8$. For instance, if ϕ is $(\mathsf{Meets}(e,f) \wedge \mathsf{Holds}(p,e)) \vee (\mathsf{Meets}(f,g) \wedge \mathsf{Holds}(q,f))$, then $n_3(\phi) = 2$, $n_8(\phi) = 2$, and $n_i(\phi) = 0$ for i other than 2 or 8.

Theorem 4.12. *Let ϕ be an ATL_B^+ formula. The size of $\mathrm{tr}[\phi]$ increases linearly in terms of $|\phi|$.*

Proof. By structural induction on ϕ, we obtain $|\mathrm{tr}[\phi]| = |\phi| + \sum_{1\leq i\leq 8} n_i(\phi)\cdot m_i$. Let $m_{\max} = \max_{1\leq i\leq 8} m_i$, then $|\mathrm{tr}[\phi]| \leq |\phi| + m_{\max}\sum_{1\leq i\leq 8} n_i(\phi)$. Since the number of α-formulas within ϕ is not more than twice the size of ϕ, then $n_i(\phi) \leq 2|\phi|$ for $1 \leq i \leq 8$, and thus $|\mathrm{tr}[\phi]| \leq (1+16m_{\max})|\phi|$. We conclude that $|\mathrm{tr}[\phi]|$ is $\mathscr{O}(|\phi|)$ as required. $\square$

Results from [5, 6] show that the satisfiability problems for HRPNL whose models are interpreted over discrete or dense linearly ordered sets are decidable in NEXPTIME. This means that the satisfiability of an HRPNL formula φ can be checked by a nondeterministic algorithm in $\mathscr{O}\left(2^{p(|\varphi|)}\right)$ time, where $p(|\varphi|)$ is a polynomial in $|\varphi|$. From this result and Theorem 4.12, we can argue that the satisfiability of an ATL_B^+ formula ϕ can be checked in $\mathscr{O}\left(2^{p(|\phi|)}\right)$ time.

5. Concluding Remarks

In this paper we show that every ATL_B^+ formula can be translated in linear time into an equisatisfiable RPNL formula using Algorithm 4.1. The size of the resulting RPNL formula increases linearly in terms of the size of the translated formula. These results imply that RPNL verification technique can be used to verify a system whose specifications are formalized using ATL_B^+.

Conflict of Interests

The author declares that there is no conflict of interests.

References

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

[2] J. F. Allen, Towards a general theory of actions and time, Artificial Intelligence, 23 (1984) 123-154.

[3] D. Bresolin, V. Goranko, A. Montanari, G. Sciavicco, On decidability and expressiveness of propositional neighborhood logics, Proceeding of the International Symposium on Logical Foundations of Computer Science, Lecture Notes in Computer Science, Springer, (2007) 84-99.

[4] D. Bresolin, V. Goranko, A. Montanari, G. Sciavicco, Propositional Interval Neighborhood Logics: Expressiveness, Decidability, and Undecidable Extensions, Annals of Pure and Applied Logics 161 (2009) 289-304.

[5] D. Bresolin, A. Montanari, P. Sala, G. Sciavicco, Optimal tableaux for right propositional neighborhood logic over linear orders, Proceeding of the 11th European Conference on Logics in AI, eds. by S. Holldöbler, C. Lutz, and H. Wansing, Lecture Notes in AI, Springer, 5293 (2008) 62-75.

[6] D. Bresolin, A. Montanari, P. Sala, G. Sciavicco, Optimal tableau systems for propositional neighborhood logic over all, dense, and discrete linear orders, Proceeding of the 20th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX), Lecture Notes in AI, Springer, 6793 (2011) 73-87.

[7] M. E. Coniglio, Towards a stronger notion of translation between logics, Manuscrito - Revista Internacional de Filosofia 28 (2005) 231-262.

[8] V. Goranko, A. Montanari, G. Sciavicco, Propositional interval nneighborhood temporal logics, Journal of Universal Computer Science, 9 (2003) 1137-1167.

[9] G. Roşu, S. Bensalem, Allen linear (interval) temporal logic - Translation to LTL and Monitor Synthesis, International Conference on Computer Aided Verification, Lecture Notes in Computer Science, Springer, 4144 (2006) 263-277.