An Asymptotically Correct Finite Path Semantics for LTL

Andreas Morgenstern, Manuel Gesell, and Klaus Schneider

Embedded Systems Group, Department of Computer Science,
University of Kaiserslautern
P.O. Box 3049
67653 Kaiserslautern, Germany
{morgenstern,gesell,schneider}@cs.uni-kl.de
http://es.cs.uni-kl.de

Abstract. Runtime verification of temporal logic properties requires a definition of the truth value of these properties on the finite paths that are observed at runtime. However, while the semantics of temporal logic on infinite paths has been precisely defined, there is not yet an agreement on the definition of the semantics on finite paths. Recently, it has been observed that the accuracy of runtime verification can be improved by a 4-valued semantics of temporal logic on finite paths. However, as we argue in this paper, even a 4-valued semantics is not sufficient to achieve a semantics on finite paths that converges to the semantics on infinite paths. To overcome this deficiency, we consider in this paper Manna and Pnueli's temporal logic hierarchy consisting of safety, liveness (guarantee), co-Büchi (persistence), and Büchi (recurrence) properties. We propose the use of specialized semantics for each of these subclasses to improve the accuracy of runtime verification. In particular, we prove that our new semantics converges to the infinite path semantics which is an important property that has not been achieved by previous approaches.

1 Introduction

Runtime verification aims at detecting faults of a system by monitoring its input/output behavior during runtime. For the specification of the desired behavior, temporal logics in general, and linear temporal logic (LTL) in particular, proved to be convenient formalisms to precisely and conveniently determine complex temporal properties. During the last two decades, many model-checking procedures for temporal logics have been developed that improved the efficiency to become interesting for practical use. As a consequence, the PSL logic (extending LTL) became now an industry standard that is used by many tools and programming languages. Since temporal logics are therefore well-established, it is natural to use them also for runtime verification.

However, while the tools used to solve the model-checking problem refer to the original LTL semantics that is given for infinite behaviors, runtime verification can only reason about the finite behavior that has been observed up to a considered point of time. Whether a fault occurred at runtime can therefore not be decided by the existing LTL semantics, and instead one has to consider the meaning of LTL formulas on finite paths.

N. Bjørner and A. Voronkov (Eds.): LPAR-18, LNCS 7180, pp. 304–319, 2012.

While the semantics of LTL on infinite paths has been precisely defined in the literature (without producing alternatives), there is not yet such a consensus on the meaning of LTL properties on finite paths. Several two-valued semantics for LTL on finite paths have been proposed [6] that are well-suited for safety properties. In [1], special reset and abort operators have been added to LTL to cope with finite path semantics, but these do not solve the problem for the other operators. Recently, it has been observed that at least a three-valued semantics is necessary to give informative results [14,2,13]. Using three-valued semantics, a property evaluates to true or false whenever the truth value defined by the LTL semantics on infinite paths is already determined by its finite prefix. If the finite prefix does not determine the truth value, an inconclusive result is obtained by a third truth value. In the first case, the considered prefix is called a *good* prefix, otherwise it is called a *bad* prefix. This scheme is well-suited for pure *safety* properties like $\mathsf{G}p$ and simple liveness (guarantee) properties like $\mathsf{F}p$. Indeed, it has been observed in [8] that the only properties for which a three-valued semantics gives satisfactory results consists of boolean combinations of safety and guarantee properties which form the obligation properties in the temporal logic hierarchy of Manna and Pnueli [5]. For this reason, the formulas in this class have been already called prefix properties in [15,16].

However, there are many properties which can not be dealt with such a three-valued semantics: consider e. g. the request/acknowledge property $\mathsf{G}(r \rightarrow \mathsf{F}a)$ taken from [2] that states that every request is finally acknowledged. Finite paths cannot decide the truth of this property since it belongs to the Büchi (recurrence) class, but not to the prefix (obligation) class. Hence a three-valued logic will always evaluate to an inconclusive result. In [3], previously proposed semantics for LTL are compared with each other and a new four-valued semantics for LTL on finite paths was proposed that is argued to overcome these problems. For the request/acknowledge property, the proposed RV-LTL semantics yields value $\top_P$ (meaning *'presumable good'*) whenever the so-far read finite input path ends at a point of time where a holds. The value $\bot_P$ (meaning *'presumably bad'*) is used whenever the so-far read finite input path ends at a point of time where r holds, indicating that is is likely that the specification remains unsatisfied.

While the proposed solution gives a reasonable result for the above mentioned request/acknowledge property, we argue that for other interesting properties, the proposed RV-LTL semantics gives misleading results: For example, consider the property $\mathsf{FG}p_1 \vee \mathsf{FG}p_2$. This property states that from a certain point on, either always p_1 or always p_2 holds (note its equivalence to $\mathsf{F}(\mathsf{G}p_1 \vee \mathsf{G}p_2)$). For the behavior $p_1, p_2, p_1, p_2, \ldots$, RV-LTL determines the value $\top_P$ (presumably good) for every finite prefix, indicating the misleading result that the property is 'presumably true' while the property is not satisfied on the infinite behavior.

In this paper, we therefore define a new semantics of LTL on finite paths to improve the previously proposed semantics so that the definition on finite paths converges to the definition of infinite paths. To this end, we consider the temporal logic hierarchy of Manna and Pnueli [5,16]. Instead of distinguishing between *presumably good* and *presumably bad* in case no definitive answer is possible, we use truth values that are more specialized to the unknown infinite suffix. For example, a persistence (co-Büchi) property like $\mathsf{FG}p_1 \vee \mathsf{FG}p_2$ is evaluated

over a four-valued semantics with the truth values $\{\mathsf{true}, \mathsf{false}, \top_{\mathsf{FG}}, \bot_{\mathsf{FG}}\}$ with the intuition that whenever an infinite path satisfies the property from a certain point of time on, we assign $\top_{\mathsf{FG}}$ for the corresponding prefixes of that path from that point on. On the other hand, output $\bot_{\mathsf{FG}}$ is used whenever the system has not yet stabilized; and outputs true and false are used whenever a definite answer is possible.

While this modification of the many-valued semantics seems to be only a notational change, it already improves the evaluation of the above example: $\mathsf{FG}p_1 \vee \mathsf{FG}p_2$ is evaluated on every finite prefix of even length of p_1, p_2, p_1, p_2 to $\bot_{\mathsf{FG}}$ so that a verification engineer considering the results during simulation or runtime verification will see that either the system has not yet stabilized or that something is wrong in the system. Indeed, we prove that our new semantics is asymptotically correct for persistence properties in the sense that only finitely many prefixes of a satisfying infinite path of a persistence property yield the wrong result $\bot_{\mathsf{FG}}$.

Recurrence properties like the request/response property are evaluated over a *different four-valued set of truth values* $\{\mathsf{true}, \mathsf{false}, \top_{\mathsf{GF}}, \bot_{\mathsf{GF}}\}$. For the prefixes of a satisfying infinite path of a recurrence property, infinitely often the right result $\top_{\mathsf{GF}}$ is obtained so that we again obtain an asymptotically correct semantics for recurrence properties. For the simpler classes of safety, guarantee and prefix (boolean combination of safety and guarantee) properties that can be already evaluated on a three-valued semantics, we obtain the same semantics as in RV-LTL (and LTL_3). Our improvements are based on a new definition of the disjunction operator which also considers the prefix of a path, and a context-dependent interpretation of the next operator.

While we ultimately fail to give a finite path semantics for full LTL, we are able to provide a solution for all classes of the temporal logic hierarchy. In practice, this is no restriction: nearly all formulas belong (syntactically) to the most powerful class of the hierarchy and for others, it is typically not difficult to find an equivalent formula in that class [11]. This is due to the fact that this class contains an (semantically) equivalent formula for every LTL formula [16].

The outline of this paper is as follows: In Section 2, the syntax and semantics of LTL over infinite words and Manna and Pnueli's temporal logic hierarchy are reviewed. We reconsider the definition of two previously published definitions of LTL on finite words in Section 3, namely LTL_3 [4] which is essentially the same as [13] and the four-valued semantics of RV-LTL [3] which is essentially the logic-based variant of [8]. Since both logics produce misleading results on certain properties, we present a new semantics of LTL on finite paths in Section 4. We prove that our new semantics is asymptotically correct in Section 4.3 and add concluding remarks in Section 5.

2 Syntax and Semantics of LTL

Linear Temporal Logic (LTL) [12,7] is a popular formalism for the specification of temporal properties. For a given set of boolean variables (propositions) $\mathcal{V}$, we define the set of LTL formulas by the following grammar: $\varphi := \mathcal{V} \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{X}\varphi \mid [\varphi \,\underline{\mathsf{U}}\, \varphi]$. Additionally, we define $\varphi \wedge \psi$, $\mathsf{F}\varphi$, $\mathsf{G}\varphi$, and $[\varphi \,\mathsf{U}\, \psi]$ as abbreviations

for $\neg(\neg\varphi \vee \neg\psi)$, $[1 \underline{\mathsf{U}} \varphi]$, $\neg\mathsf{F}\neg\varphi$, and $[\varphi \underline{\mathsf{U}} \psi] \vee \mathsf{G}\varphi$, respectively. The semantics of LTL is usually given with respect to an infinite path through a transition system. These infinite paths are nothing else than infinite sequences of boolean assignments to the variables $\mathcal{V}$:

Definition 1 (Infinite Words). *Given a set of atomic propositions $\mathcal{V}$, an* infinite word *is a function $\mathfrak{v} : \mathbb{N} \to \wp(\mathcal{V})$. For reasons of simplicity, $\mathfrak{v}(i)$ is often denoted by $\mathfrak{v}^{(i)}$ for $i \in \mathbb{N}$. Using this notation, words are often given in the form $\mathfrak{v}^{(0)}\mathfrak{v}^{(1)}\ldots$. The suffix starting at t is written as : $\mathfrak{v}^{(t\ldots)} := \mathfrak{v}^{(t)}\mathfrak{v}^{(t+1)}\ldots$. For $a \in \mathcal{V}$, we define $\mathfrak{v} = a^\omega$ as $\mathfrak{v} = a^{(0)}a^{(1)}a^{(2)}\ldots$. Given an infinite word $\mathfrak{v} = a^{(0)}a^{(1)}\ldots$, we define $\mathfrak{v}^{(s\ldots t)}$ as the finite word $\mathfrak{u} = \mathfrak{v}^{(s)}\mathfrak{v}^{(s+1)}\ldots\mathfrak{v}^{(t)}$.*

The semantics of LTL is typically defined as follows [7,16]:

Definition 2 (Semantics of LTL). *Given an infinite word $\mathfrak{v}$, the following rules define the semantics of LTL:*

- $[\mathfrak{v} \models_\omega p]$ *iff* $p \in \mathfrak{v}^{(0)}$ *for* $p \in \mathcal{V}$
- $[\mathfrak{v} \models_\omega \neg\varphi]$ *iff* $[\mathfrak{v} \not\models_\omega \varphi]$
- $[\mathfrak{v} \models_\omega \varphi \wedge \psi]$ *iff* $[\mathfrak{v} \models_\omega \varphi]$ *and* $[\mathfrak{v} \models_\omega \psi]$
- $[\mathfrak{v} \models_\omega \varphi \vee \psi]$ *iff* $[\mathfrak{v} \models_\omega \varphi]$ *or* $[\mathfrak{v} \models_\omega \psi]$
- $[\mathfrak{v} \models_\omega \mathsf{X}\varphi]$ *iff* $[\mathfrak{v}^{(1\ldots)} \models_\omega \varphi]$
- $[\mathfrak{v} \models_\omega [\varphi \underline{\mathsf{U}} \psi]]$ *iff there is a* δ *such that* $[\mathfrak{v}^{(\delta\ldots)} \models_\omega \psi]$ *and for all* t *with* $t < \delta$, *we have* $[\mathfrak{v}^{(t\ldots)} \models_\omega \varphi]$

In [5,15,16], a temporal logic hierarchy has been defined in analogy to the hierarchy of ω-automata. Following [15], we define the hierarchy of temporal formulas by the grammar rules of Figure 1:

$P_\mathsf{G} ::= \mathcal{V} \mid \neg P_\mathsf{F} \mid P_\mathsf{G} \wedge P_\mathsf{G} \mid P_\mathsf{G} \vee P_\mathsf{G}$ $\mid \mathsf{X}P_\mathsf{G} \mid [P_\mathsf{G} \,\mathsf{U}\, P_\mathsf{G}]$	$P_\mathsf{F} ::= \mathcal{V} \mid \neg P_\mathsf{G} \mid P_\mathsf{F} \wedge P_\mathsf{F} \mid P_\mathsf{F} \vee P_\mathsf{F}$ $\mid \mathsf{X}P_\mathsf{F} \mid [P_\mathsf{F} \,\underline{\mathsf{U}}\, P_\mathsf{F}]$
$P_\mathsf{Prefix} ::= P_\mathsf{G} \mid P_\mathsf{F} \mid \neg P_\mathsf{Prefix} \mid P_\mathsf{Prefix} \wedge P_\mathsf{Prefix} \mid P_\mathsf{Prefix} \vee P_\mathsf{Prefix}$	
$P_\mathsf{GF} ::= P_\mathsf{Prefix}$ $\mid \neg P_\mathsf{FG} \mid P_\mathsf{GF} \wedge P_\mathsf{GF} \mid P_\mathsf{GF} \vee P_\mathsf{GF}$ $\mid \mathsf{X}P_\mathsf{GF} \mid [P_\mathsf{GF} \,\mathsf{U}\, P_\mathsf{GF}] \mid [P_\mathsf{GF} \,\underline{\mathsf{U}}\, P_\mathsf{F}]$	$P_\mathsf{FG} ::= P_\mathsf{Prefix}$ $\mid \neg P_\mathsf{GF} \mid P_\mathsf{FG} \wedge P_\mathsf{FG} \mid P_\mathsf{FG} \vee P_\mathsf{FG}$ $\mid \mathsf{X}P_\mathsf{FG} \mid [P_\mathsf{FG} \,\underline{\mathsf{U}}\, P_\mathsf{FG}] \mid [P_\mathsf{G} \,\mathsf{U}\, P_\mathsf{FG}]$
$P_\mathsf{Streett} ::= P_\mathsf{GF} \mid P_\mathsf{FG} \mid \neg P_\mathsf{Streett} \mid P_\mathsf{Streett} \wedge P_\mathsf{Streett} \mid P_\mathsf{Streett} \vee P_\mathsf{Streett}$	

Fig. 1. Classes of the Temporal Logic Hierarchy

Definition 3 (Temporal Logic Classes). *We define the logics TL_κ for $\kappa \in \{\mathsf{G}, \mathsf{F}, \mathsf{Prefix}, \mathsf{FG}, \mathsf{GF}, \mathsf{Streett}\}$ by the grammar rules given in Figure 1, where TL_κ is the set of formulas that can be derived from the non-terminal P_κ ($\mathcal{V}$ represents any variable $v \in \mathcal{V}$).*

TL_G is the set of formulas where each occurrence of a weak/strong temporal operator is positive/negative, and similarly, each occurrence of a weak/strong temporal operator in TL_F is negative/positive. Hence, both logics are dual to each other, which means that one contains the negations of the other one. $\mathsf{TL}_\mathsf{Prefix}$

is the boolean closure of TL_G and TL_F. The logics TL_GF and TL_FG are constructed in the same way as TL_G and TL_F; however, there are two differences: (1) these logics allow occurrences of $\mathsf{TL}_\mathsf{Prefix}$ where otherwise variables would have been required in TL_G and TL_F, and (2) there are additional 'asymmetric' grammar rules. It can be easily proved that TL_GF and TL_FG are also dual to each other, and their intersection strictly contains $\mathsf{TL}_\mathsf{Prefix}$. Finally, $\mathsf{TL}_\mathsf{Streett}$ is the boolean closure of TL_GF and TL_FG. While there are syntactic restrictions on $\mathsf{TL}_\mathsf{Streett}$, i. e. not every LTL formula is a $\mathsf{TL}_\mathsf{Streett}$ formula, $\mathsf{TL}_\mathsf{Streett}$ contains for each LTL formula an equivalent formula, and nearly all formulas used in practice belong to $\mathsf{TL}_\mathsf{Streett}$ [11]. Moreover, for those formulas not in $\mathsf{TL}_\mathsf{Streett}$, it is typically not difficult to find an equivalent one in $\mathsf{TL}_\mathsf{Streett}$.

3 Previous Definitions of LTL on Finite Paths

In the following, we consider the recently proposed semantics for LTL on finite paths as given in [3]. We also show that this definition has certain deficiencies.

3.1 LTL_3

In [2], LTL_3 was introduced as an extension of LTL to finite paths which follows the idea that a finite path is a prefix of a so-far unknown infinite path. LTL_3 uses three-valued truth values $\mathbb{B}_3 = \{1, 0, ?\}$. While the syntax of LTL_3 coincides with that of LTL, its semantics is defined on finite words:

Definition 4 (Semantics of LTL_3). *Let $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\ldots\mathfrak{u}^{(n)} \in \Sigma^*$ denote a finite path of length $n+1$. The truth value of a LTL_3 formula φ w.r.t. $\mathfrak{u}$, denoted by $[\mathfrak{u} \models_3 \varphi]$ is defined as follows:*

$$[\mathfrak{u} \models_3 \varphi] = \begin{cases} 1 & \text{if } \forall \mathfrak{w} \in \Sigma^\omega : \mathfrak{u}\mathfrak{w} \models_\omega \varphi \\ 0 & \text{if } \forall \mathfrak{w} \in \Sigma^\omega : \mathfrak{u}\mathfrak{w} \not\models_\omega \varphi \\ ? & \text{else} \end{cases}$$

The intuition behind LTL_3 is clear: whenever all infinite words obtained by concatenating the finite word with an infinite suffix agree on the truth value of φ, this truth value is used also for the prefix. Otherwise, the value ? is used. As argued in [3], LTL_3 can never give a result other than ? for request-response properties like $\mathsf{G}(r \rightarrow \mathsf{F}a)$ since every prefix of an infinite accepted word can be both a good or a bad prefix. Hence the authors propose to combine LTL_3 with another logic called FLTL.

3.2 FLTL

In [9,3] it is argued that there is a need to distinguish between a strong ($\underline{\mathsf{X}}$) and a weak ($\overline{\mathsf{X}}$) next operator when interpreting LTL over finite paths. While a weak next operator should be satisfied whenever no next position exists, a strong next operator should be evaluated to false in that case. This leads to the following definition of FLTL:

Definition 5 (FLTL). *Let* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\dots\mathfrak{u}^{(n)} \in \Sigma^*$ *denote a finite path of length* $n+1$ *with* $\mathfrak{u} \neq \varepsilon$*. The truth value of a FLTL formula* φ *wrt.* $\mathfrak{u}$*, denoted as* $[\mathfrak{u} \models_{FLTL} \varphi]$*, is an element of* $\mathbb{B}_2 = \{\bot, \top\}$ *and is inductively defined as follows: While atomic propositions and boolean operators are defined as for LTL, the temporal operators are defined as follows:*

$$[\mathfrak{u} \models_{FLTL} \overline{\mathsf{X}}\varphi] = \begin{cases} [\mathfrak{u}^1 \models_{FLTL} \varphi] & \text{if } \mathfrak{u}^1 \neq \varepsilon \\ \top & \text{else} \end{cases}$$

$$[\mathfrak{u} \models_{FLTL} \underline{\mathsf{X}}\varphi] = \begin{cases} [\mathfrak{u}^1 \models_{FLTL} \varphi] & \text{if } \mathfrak{u}^1 \neq \varepsilon \\ \bot & \text{else} \end{cases}$$

$$[\mathfrak{u} \models_{FLTL} [\varphi\ \underline{\mathsf{U}}\ \psi]] = \begin{cases} \top & \exists k \in \{1,\dots n\} : [\mathfrak{u}^k \models_{FLTL} \psi] = \top \wedge \\ & \forall 1 \leq l \leq k : [\mathfrak{u}^l \models_{FLTL} \phi] = \top \\ \bot & \text{else} \end{cases}$$

$$[\mathfrak{u} \models_{FLTL} [\varphi\ \mathsf{U}\ \psi]] = \begin{cases} \top & \forall 1 \leq l \leq n : [\mathfrak{u}^l \models_{FLTL} \varphi] = \top \vee \\ & \exists k \in \{1,\dots n\} : [\mathfrak{u}^k \models_{FLTL} \psi] = \top \wedge \\ & \forall 1 \leq l \leq k : [\mathfrak{u}^l \models_{FLTL} \phi] = \top \\ \bot & \text{else} \end{cases}$$

In [3], the two definitions of LTL_3 and FLTL are combined in a logic called RV-LTL. This logic is evaluated over a four-valued de Morgan lattice $0 \sqsubset \bot_P \sqsubset \top_P \sqsubset 1$ to express false, presumably false, presumably true and true. To obtain a de Morgan lattice and thus a truth domain, the operators $\sqcap$ and $\sqcup$ are defined as expected and $1/0$ and $\top_P/\bot_P$, respectively, are defined to be complementary to each other. Note that the thereby obtained truth domain $\mathbb{B}_4$ is not a boolean lattice.

RV-LTL is now defined such that the truth value of LTL_3 is used whenever it is conclusive, i.e. gives 1 or 0. If LTL_3 provides the inconclusive result (?), the definition of FLTL is used instead:

Definition 6 (RV-LTL). *Let* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\dots\mathfrak{u}^{(n)} \in \Sigma^*$ *denote a finite path of length* $n+1$ *with* $\mathfrak{u} \neq \varepsilon$*. The truth value of an RV-LTL formula* φ *wrt.* $\mathfrak{u}$*, denoted as* $[\mathfrak{u} \models_{FLTL} \varphi]$*, is an element of* $\mathbb{B}_4$ *and is defined as follows:*

$$[\mathfrak{u} \models_3 \varphi] = \begin{cases} 1 & \text{if } [\mathfrak{u}\mathfrak{w} \models_3 \varphi] = 1 \\ 0 & \text{if } [\mathfrak{u}\mathfrak{w} \models_3 \varphi] = 0 \\ \top_P & \text{if } [\mathfrak{u}\mathfrak{w} \models_3 \varphi] = ? \wedge [\mathfrak{u}\mathfrak{w} \models_{FLTL} \varphi] = \top \\ \bot_P & \text{if } [\mathfrak{u}\mathfrak{w} \models_3 \varphi] = ? \wedge [\mathfrak{u}\mathfrak{w} \models_{FLTL} \varphi] = \bot \end{cases}$$

3.3 Problems with RV-LTL

In the following, we consider some examples to show unsatisfactory results of the RV-LTL semantics.

Request/Acknowledge Properties: In [3], it has been shown that $\mathsf{F}\varphi \equiv_{RV} \varphi \vee \underline{\mathsf{X}}\mathsf{F}\varphi$ holds, satisfying the intuitive meaning that $\mathsf{F}\varphi$ holds, iff either φ holds immediately or there must be a future state satisfying φ. If no such future state exists, the formula evaluates to $\bot_P$, unless the formula evaluates to one of $\{1, 0\}$, in which case the future is not important. Similarly, we have $\mathsf{G}\varphi \equiv_{RV} \varphi \wedge \overline{\mathsf{X}}\mathsf{G}\varphi$ which shows that φ must be satisfied in the current state and in all observable future states. Hence, if there is no future state, the formula evaluates to $\top_P$, unless the formula evaluates to one of $\{1, 0\}$. Hence, the request/acknowledge property is evaluated as follows:

$$\begin{aligned}\mathsf{G}\,(r \rightarrow \mathsf{F}a) &\equiv_{RV} (r \rightarrow \mathsf{F}a) \wedge \overline{\mathsf{X}}\mathsf{G}\,(r \rightarrow \mathsf{F}a)\\ &\equiv_{RV} (\neg r \vee a \vee \underline{\mathsf{X}}\mathsf{F}a) \wedge \overline{\mathsf{X}}\mathsf{G}\,(r \rightarrow \mathsf{F}a)\end{aligned}$$

This formula evaluates to $\bot_P$ under RV-LTL if the path contains an r but ends before a occurs and evaluates to $\top_P$ in all other cases. Thus, its semantics seems to be reasonable. However, consider the following generalized request/acknowledge property:

$$\begin{aligned}&\mathsf{G}\,(r_1 \rightarrow \mathsf{F}a_1) \wedge \mathsf{G}\,(r_2 \rightarrow \mathsf{F}a_2) \equiv_{RV}\\ &\quad(\neg r_1 \vee a_1 \vee \underline{\mathsf{X}}\mathsf{F}a_1) \wedge \overline{\mathsf{X}}\mathsf{G}\,(r_1 \rightarrow \mathsf{F}a_1) \wedge (\neg r_2 \vee a_2 \vee \underline{\mathsf{X}}\mathsf{F}a_2) \wedge \overline{\mathsf{X}}\mathsf{G}\,(r_2 \rightarrow \mathsf{F}a_2)\end{aligned}$$

According to the previous discussion, a finite word that satisfies $r_1 \wedge a_2$ on odd positions and $r_2 \wedge a_1$ on even positions (the others being false) will be evaluated to $\bot_P$ in all states. This is unfortunate because the infinite word that is obtained by an infinite concatenation of those odd/even positions clearly satisfies the specification under the infinite semantics

Stabilization Properties: While having a semantics that evaluates to a 'bad' value even if we read a 'good' word may be acceptable, the following example demonstrates that RV-LTL even has the undesirable property that for a non-accepted word of a LTL property, each finite prefix may be evaluated to $\top_P$. To this end, consider the following RV-LTL equivalence:

$$\mathsf{FG}a \vee \mathsf{FG}\neg a \equiv_{RV} \mathsf{F}\left(a \wedge \overline{\mathsf{X}}\mathsf{G}a\right) \vee \mathsf{F}\left(\neg a \wedge \overline{\mathsf{X}}\mathsf{G}\neg a\right)$$

Since $\overline{\mathsf{X}}\mathsf{G}a$ is evaluated weakly and $\mathsf{FG}a$ may start the evaluation of $\overline{\mathsf{X}}\mathsf{G}a$ at an arbitrary position (for example, the last position of the finite word read so far), every finite word that ends with a evaluates to true. However, with the same argument, every finite word that ends with $\neg a$ is evaluated to true, too. Hence, the word with a on even positions and $\neg a$ on odd positions will be evaluated to $\top_P$ on each position. Nevertheless, the thereby constructed infinite word is not accepted by the infinite semantics of LTL. The problem is that the evaluation of $\varphi \vee \psi$ in RV-LTL does not consider which property has been responsible for the satisfaction in previous steps and hence such an infinite shift between good and bad prefixes for φ and ψ is possible. We will later see that this problem can be fixed by an improved semantics for the disjunction.

The Problem with Two Next-Operators: While having weak/strong next operators may seem plausible at first sight, we argue that it leads to problems when one is interested in an asymptotically correct semantics for LTL. One might expect that both $\underline{\mathsf{X}}$ and $\overline{\mathsf{X}}$ will behave asymptotically like the original X operator. However, this may not hold: Consider e.g. the property $\mathsf{G}\underline{\mathsf{X}}a$. This property is evaluated to $\bot$ on every input at every step. However, since the formula $\mathsf{GX}a$ holds on the word a^ω with the infinite semantics, one might expect that at least at some point, $\mathsf{G}\underline{\mathsf{X}}a$ yields $\top$, which is however not the case. Moreover, the intuitive meaning of G should be that it is evaluated to $\top$ as long as we have not detected that the property is violated. This intuitive interpretation does no longer hold if we allow a $\underline{\mathsf{X}}$ inside a G. A similar problem occurs with $\mathsf{F}\overline{\mathsf{X}}a$. One might expect that in the limit, this formula behaves like $\mathsf{FX}a$. However, since $\overline{\mathsf{X}}a$ is evaluated weak, it is not hard to see that this formula evaluates to $\top$, no matter which input is read.

To circumvent those problems, we refrain therefore from two different next operators and evaluate the next operator depending on the context of a formula. The intuitive idea behind our construction is that if $\mathsf{X}\varphi$ is in the scope of a weak temporal operator, it is evaluated weakly, otherwise it is evaluated strongly. Hence, for $\mathsf{TL_G}$ formulas, we evaluate the formula always weakly in accordance to the intuitive meaning that a safety formula should be evaluated to $\top$ as long as nothing bad happened. Analogously, we evaluate a X operator in the scope of a strong until operator strongly, as e.g. in $\mathsf{F}(a \wedge \mathsf{X}b)$. This supports the intuitive meaning that a guarantee property should be evaluated to $\bot$ as long as it has not definitely been satisfied.

4 Asymptotic Finite Linear Temporal Logic (RV^∞–LTL)

In this section, we define for each $\kappa \in \{\mathsf{G}, \mathsf{F}, \mathsf{Prefix}, \mathsf{FG}, \mathsf{GF}, \mathsf{Streett}\}$ specialized semantics that are intended to replace the FLTL semantics in the definition of RV-LTL. We call the resulting logics RV^∞–TL_κ. For better readability of the following definitions, we assume that the case conditions are evaluated in a top-down manner, i.e, if the first satisfied case is used (ignoring all remaining ones, including also possibly satisfied cases).

4.1 The Temporal Logic Classes RV^∞–$\mathsf{TL_G}$ and RV^∞–$\mathsf{TL_F}$

We start by defining the base class RV^∞–$\mathsf{TL_G}$:

Definition 7 (Semantics of Linear Temporal Logic RV^∞–$\mathsf{TL_G}$). *Let* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\dots\mathfrak{u}^{(n)} \in \Sigma^*$ *denote a finite path of length* $n+1$*. The truth value of an* $\mathsf{TL_G}$ *formula* φ *wrt.* $\mathfrak{u}$*, denoted with* $[\mathfrak{u} \models_\mathsf{G} \varphi]$*, is an element of* $\mathbb{B}_3$ *and is inductively defined as follows:*

- $[\varepsilon \models_\mathsf{G} \varphi] = \top_\mathsf{G}$
- $[\mathfrak{u} \models_\mathsf{G} a] = \begin{cases} 1 & \text{if } a \in \mathfrak{u}^{(0)} \\ 0 & \text{else} \end{cases}$, *for every* $a \in \mathcal{V}$

$$- \ [\mathfrak{u} \models_{\mathsf{G}} \varphi \wedge \psi] = \begin{cases} 1 & \text{if } \forall \mathfrak{w} \in \Sigma^\omega : \mathfrak{u}\mathfrak{w} \models_\omega \varphi \wedge \psi \\ \top_{\mathsf{G}}, & \text{if } [\mathfrak{u} \models_{\mathsf{G}} \varphi] = \top_{\mathsf{G}} \text{ and } [\mathfrak{u} \models_{\mathsf{G}} \psi] = \top_{\mathsf{G}} \\ 0, & \text{otherwise} \end{cases}$$

$$- \ [\mathfrak{u} \models_{\mathsf{G}} \varphi \vee \psi] = \begin{cases} 1 & \text{if } \forall \mathfrak{w} \in \Sigma^\omega : \mathfrak{u}\mathfrak{w} \models_\omega \varphi \vee \psi \\ \top_{\mathsf{G}}, & \text{if } [\mathfrak{u} \models_{\mathsf{G}} \varphi] = \top_{\mathsf{G}} \text{ or } [\mathfrak{u} \models_{\mathsf{G}} \psi] = \top_{\mathsf{G}} \\ 0, & \text{otherwise} \end{cases}$$

$$- \ [\mathfrak{u} \models_{\mathsf{G}} \mathsf{X}\varphi] = [\mathfrak{u}^{(1\ldots n)} \models_{\mathsf{G}} \varphi]$$

$$- \ [\mathfrak{u} \models_{\mathsf{G}} [\varphi \ \mathsf{U} \ \psi]] = [\mathfrak{u} \models_{\mathsf{G}} (\psi \vee (\varphi \wedge \mathsf{X}[\varphi \ \mathsf{U} \ \psi]))]$$

Taking into account that the X operator is evaluated weakly in a $\mathsf{TL_G}$ formula, the definition of $[\varphi \ \mathsf{U} \ \psi]$ is exactly the fixpoint evaluation of $[\varphi \ \mathsf{U} \ \psi]$. Hence, it is not hard to see that FLTL and RV^∞–LTL are evaluated in the same manner:

Proposition 1. *Let φ be a $\mathsf{TL_G}$ formula and $\mathfrak{u} \neq \varepsilon$ be a finite word. Let φ' be obtained from φ by replacing each X operator by a $\overline{\mathsf{X}}$ operator. Then, the following holds: $[\mathfrak{u} \models_{\mathsf{G}} \varphi] = \top_{\mathsf{G}}$ iff $[\mathfrak{u} \models_{FLTL} \varphi'] = \top$.*

Since the negations of safety properties are guarantee properties, we define:

Definition 8 (Semantics of Linear Temporal Logic RV^∞–$\mathsf{TL_F}$). *Given a finite prefix $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\ldots\mathfrak{u}^{(n)}$ of an infinite word $\mathfrak{u}_\infty$, the the semantics of RV^∞–$\mathsf{TL_F}$ is defined by $[\mathfrak{u} \models_{\mathsf{F}} \varphi] = \begin{cases} 1, & \text{if } [\mathfrak{u} \models_{\mathsf{G}} \neg\varphi] = 0 \\ \bot_{\mathsf{F}}, & \text{if } [\mathfrak{u} \models_{\mathsf{G}} \neg\varphi] = \top_{\mathsf{G}} \\ 0, & \text{otherwise} \end{cases}$*

Hence, the following is also obvious:

Proposition 2. *Let φ be a $\mathsf{TL_F}$ formula and $\mathfrak{u} \neq \varepsilon$ be a finite word. Let φ' be obtained from φ by replacing each X operator by a $\underline{\mathsf{X}}$ operator. Then, the following holds: $[\mathfrak{u} \models_{\mathsf{F}} \varphi] = 1$ iff $[\mathfrak{u} \models_{FLTL} \varphi'] = \top$.*

4.2 The Temporal Logic RV^∞–$\mathsf{TL_{FG}}$

In the following, we will use $\mathfrak{u} \models_{\mathsf{FG}} \varphi$ as shorthand for $[\mathfrak{u} \models_{\mathsf{FG}} \varphi] \in \{1, \top_{\mathsf{FG}}\}$ and $\mathfrak{u} \not\models_{\mathsf{FG}} \varphi$ as a shorthand for $[\mathfrak{u} \models_{\mathsf{FG}} \varphi] \in \{0, \bot_{\mathsf{FG}}\}$

Definition 9 (Semantics of Linear Temporal Logic RV^∞–$\mathsf{TL_{FG}}$). *Let $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\ldots\mathfrak{u}^{(n)} \in \Sigma^*$ denote a finite path of length $n+1$. The truth value of a $\mathsf{TL_{FG}}$ formula φ wrt. $\mathfrak{u}$, denoted with $[\mathfrak{u} \models_{\mathsf{FG}} \varphi]$, is an element of $\mathbb{B}_4$ and is recursively defined as follows:*

$$[\mathfrak{u} \models_{\mathsf{FG}} \varphi] = \begin{cases} 1 & \text{if } \forall \mathfrak{w} \in \Sigma^\omega : \mathfrak{u}\mathfrak{w} \models_\omega \varphi \\ 0 & \text{if } \forall \mathfrak{w} \in \Sigma^\omega : \mathfrak{u}\mathfrak{w} \not\models_\omega \varphi \\ \top_{\mathsf{FG}}{}^1 & \text{if } \varphi \in \mathsf{TL_G}{}^2 \text{ and } \mathfrak{u} \models_{\mathsf{G}} \varphi \\ [\mathfrak{u} \models_{\mathsf{FG}'} \varphi] & \text{otherwise} \end{cases}$$

where we define $[\mathfrak{u} \models_{\mathsf{FG}'} \varphi]$ *by:* [3]

- $[\varepsilon \models_{\mathsf{FG}'} \varphi] = \bot_{\mathsf{FG}}$
- $[\mathfrak{u} \models_{\mathsf{FG}'} \varphi \wedge \psi] = \begin{cases} \top_{\mathsf{FG}}, & \text{if } \mathfrak{u} \models_{\mathsf{FG}} \varphi \text{ and } \mathfrak{u} \models_{\mathsf{FG}} \psi \\ \bot_{\mathsf{FG}}, & \text{otherwise} \end{cases}$
- $[\mathfrak{u} \models_{\mathsf{FG}'} \varphi \vee \psi] = \begin{cases} \top_{\mathsf{FG}}, & \text{if } \exists t \left(\mathfrak{u}^{(0\ldots t)} \not\models_{\mathsf{FG}} \varphi \vee \psi\right) \text{ and} \\ & \quad \left(\left(\forall_{k=t+1}^{n} \mathfrak{u}^{(0\ldots k)} \models_{\mathsf{FG}} \varphi\right) \text{ or } \left(\forall_{k=t+1}^{n} \mathfrak{u}^{(0\ldots k)} \models_{\mathsf{FG}} \psi\right)\right) \\ \bot_{\mathsf{FG}}, & \text{otherwise} \end{cases}$
- $[\mathfrak{u} \models_{\mathsf{FG}'} [\varphi \underline{\mathsf{U}} \psi]] = \begin{cases} \top_{\mathsf{FG}}, & \text{if } \exists t \left(\mathfrak{u}^{(0\ldots t)} \not\models_{\mathsf{FG}} [\varphi \underline{\mathsf{U}} \psi]\right) \text{ and} \\ & \quad \exists j \leq t . \mathfrak{u}^{(j\ldots n)} \models_{\mathsf{FG}} \psi \wedge \forall k < j . \mathfrak{u}^{(k\ldots n)} \models_{\mathsf{FG}} \varphi \\ \bot_{\mathsf{FG}}, & \text{otherwise} \end{cases}$
- $[\mathfrak{u} \models_{\mathsf{FG}'} [\varphi \mathsf{U} \psi]] = \begin{cases} \top_{\mathsf{FG}}, & \text{if } \left(\forall k \leq n . \mathfrak{u}^{(k\ldots n)} \models_{\mathsf{FG}} \varphi\right) \qquad (*) \\ & \text{or} \\ & \quad \exists t \left(\mathfrak{u}^{(0\ldots t)} \not\models_{\mathsf{FG}} [\varphi \mathsf{U} \psi]\right) \text{ and} \\ & \quad \exists j \leq t . \mathfrak{u}^{(j\ldots n)} \models_{\mathsf{FG}} \psi \wedge \forall k < j . \mathfrak{u}^{(k\ldots n)} \models_{\mathsf{FG}} \varphi \\ \bot_{\mathsf{FG}}, & \text{otherwise} \end{cases}$

Before presenting the proof of asymptotic correctness, we would like to emphasize the strength of our definition which is the consideration of *breakpoints*[4] in the definition of the $\vee$ and the two until operators. This *breakpoint* is a point of time where the currently evaluated formula has evaluated to $\bot_{\mathsf{FG}}$ for the last time. In case of a disjunction, the evaluation of a finite word $\mathfrak{u}$ of length $n+1$ evaluates to $\top_{\mathsf{FG}}$ if and only if after a breakpoint (which can be also at position -1 where we evaluate the empty word) one of the two formulas invariantly evaluates to $\top_{\mathsf{FG}}$. This ensures that we can not jump freely from evaluating once φ and once ψ, but must instead stick to one particular subformula.

A similar trick is used in the definition of the strong until operator. Here, we demand that the starting point j from where on ψ holds does not cross the last breakpoint. This ensures that we can not freely jump to an arbitrary position and restart the evaluation of ψ in each step in an RV-context. Consider for example the formula $[a \underline{\mathsf{U}} (\mathsf{FG}b \vee \mathsf{FG}c)]$ and the following path for runtime verification: a holds in every step while in an even step b holds and in an odd step c holds. If we remove the t-breakpoint from the definition, we would have the unpleasant behavior that this formula evaluates to $\top_{\mathsf{FG}}$ in every step which is however not true. Having the breakpoint ensures that this can not happen.

[1] The value $\top_{\mathsf{G}}$ is also reasonable here.

[2] Notice that the case $\varphi \in \mathsf{TL}_{\mathsf{F}}$ is already contained in the first case, because $\mathfrak{u} \models_{\mathsf{F}} \varphi$ is defined as $[\mathfrak{u} \models_{\mathsf{F}} \varphi] = 1$, which means that once we found that a TL_{F} is satisfied, it is satisfied for all suffixes.

[3] Notice that the case of propositional variables is handled by the $\mathsf{RV}^{\infty}\text{–}\mathsf{TL}_{\mathsf{G}}$ evaluation.

[4] Readers familiar with Miyano and Hayashi's breakpoint construction [10] for the non-determinization of alternating Büchi automata or the closely related determinization procedure for co-Büchi automata [16] might notice the similarity: in their construction a set is filled with a new set of states whenever it is discovered that the co-Büchi condition is falsified.

4.3 Asymptotic Correctness

We will now turn to the proof of asymptotic correctness. To this end, we show that an (infinite) word $\mathfrak{u}$ is accepted by a $\mathsf{TL}_{\mathsf{FG}}$ formula if and only if there is a definitive last breakpoint, called the *rv-threshold*, so that after this point, the RV^{∞}–$\mathsf{TL}_{\mathsf{FG}}$-definition invariantly evaluates to $\top_{\mathsf{FG}}$.

Lemma 1. *Let $\mathfrak{u}$ be an infinite word, and Φ be an $\mathsf{TL}_{\mathsf{FG}}$ formula. Then, the following holds: If $\mathfrak{u} \models_{\omega} \Phi$, there exists a rv-threshold $t \in \mathbb{N}$ such that for every $k > t$ we have $\mathfrak{u}^{(0\dots k)} \models_{\mathsf{FG}} \Phi$.*

Proof. We neglect the case that at some point the whole formula evaluates to 1 since in that case the claim trivially holds. We prove this lemma by induction on the formula length. Clearly, if the length is 1, we have a constant value and our rv-threshold is 1 so that the proof is obtained. Assume now that the claim holds for every formula of length l. We show that it also holds for formula of length $l + 1$. To this end, we split the proof into different cases, depending on the top-level operator Φ:

$\varphi \vee \psi$: According to the definition of LTL, we must have that $\mathfrak{u} \models_{\omega} \varphi$ or $\mathfrak{u} \models_{\omega} \psi$ holds. W.l.o.g. assume that $\mathfrak{u} \models_{\omega} \varphi$ holds. Thus, we must have a rv-threshold t for φ according to our induction hypothesis. Now assume that we have infinitely often that $\mathfrak{u}^{(0\dots k)} \not\models_{\mathsf{FG}} \Phi$ holds. Thus, we must have a position $t' > t$ such that $\mathfrak{u}^{(0\dots k)} \not\models_{\mathsf{FG}} \varphi \vee \psi$ holds. However, according to the rv-threshold, we have that $\mathfrak{u}^{(0\dots k)} \models_{\mathsf{FG}} \varphi$ holds for every $k > t$. It is not hard to see that this ensures that $\varphi \vee \psi$ is evaluated to $\top_{\mathsf{FG}}$ from that point on.

$[\varphi \,\underline{\mathsf{U}}\, \psi]$: According to the definition of LTL, there must exist a position j such that $\mathfrak{u}^{(j\cdots)} \models_{\omega} \psi$ and for all $k < j$ we have $\mathfrak{u}^{(k\cdots)} \models_{\omega} \varphi$. According to the induction hypothesis, there must exists a rv-threshold t_{ψ} and for each $k < j$ a rv-threshold t_k such that $\mathfrak{u}^{(0\dots t')} \models_{\mathsf{FG}} \psi$ for every $t' > t_{\psi}$ and $\mathfrak{u}^{(k\dots t')} \models_{\mathsf{FG}} \varphi$ for every $t' > t_k$. Thus, the maximum of $t_{\psi}, t_0 \dots t_{j-1}$ is our desired rv-threshold.

$[\varphi \,\mathsf{U}\, \psi]$: We can distinguish two cases: if $\mathfrak{u}$ also satisfies the strong until operator, we can use the same proof as above. For the second case, notice that φ is a TL_{G} formula (see Figure 1). Since φ is satisfied by $\mathfrak{u}$, the evaluation function for RV^{∞}–TL_{G} will always be evaluated to $\top_{\mathsf{G}}$. Thus, the claim holds.

$\mathsf{X}\psi$: : According to the definition of LTL, we have $\mathfrak{u}^{(1\cdots)} \models_{\omega} \psi$ and we can apply the induction hypothesis on $\mathfrak{u}^{(1\cdots)}$ to proof the claim.

$\varphi \wedge \psi$: According to the definition of LTL, $\mathfrak{u} \models_{\omega} \varphi$ and $\mathfrak{u} \models_{\omega} \psi$ holds. Thus, according to the induction hypothesis, there must exist t_{φ} and t_{ψ} as rv-thresholds. The maximum of them is the rv-threshold for $\varphi \wedge \psi$.

■

The opposite direction is shown in a similar manner:

Lemma 2. *Let $\mathfrak{u}$ be an infinite word, and Φ be an $\mathsf{TL}_{\mathsf{FG}}$ formula. Then, the following holds: If there exists a rv-threshold $t \in \mathbb{N}$ such that for every $k > t$ we have $\mathfrak{u}^{(0\dots k)} \models_{\mathsf{FG}} \Phi$. Then, $\mathfrak{u} \models_{\omega} \Phi$.*

Proof. Again, we neglect the case that at some point $\mathfrak{u}^{(0\ldots k)}$ evaluates to 1 in a rv-context. We prove this lemma by induction on the formula length. Clearly, if the length is 1, we have a constant value and our rv-threshold is 1 and the proof is obtained. Assume now that the claim holds for every formula of length l. We show that it also holds for formula of length $l+1$. To this end, we split the proof into different cases, depending on the top-level operator:

$\varphi \vee \psi$**:** According to our assumption, we have a minimal rv-threshold t such that for every $t' \geq t$ $\mathfrak{u}^{(0\ldots t')} \models_{\mathsf{FG}} \varphi \vee \psi$ holds. Since t is minimal, we have $\mathfrak{u}^{(0\ldots t-1)} \not\models_{\mathsf{FG}} \varphi \vee \psi$. According to the definition of $[\mathfrak{u} \models_{\mathsf{FG}'} \varphi \vee \psi]$, this means that either $\left(\forall_{k=t}^{t'} \mathfrak{u}^{(0\ldots k)} \models_{\mathsf{FG}} \varphi\right)$ or $\left(\forall_{k=t}^{t'} \mathfrak{u}^{(0\ldots k)} \models_{\mathsf{FG}} \psi\right)$ holds. In other words, we can not freely switch between evaluating either φ or ψ, but one of the two formulas must be evaluated to $\top_{\mathsf{FG}}$ in all places after t. This means that we can apply the induction hypothesis and can conclude that either $\mathfrak{u} \models_\omega \varphi$ or $\mathfrak{u} \models_\omega \psi$ holds. Hence, $\mathfrak{u} \models_\omega \varphi \vee \psi$ holds trivially.

$[\varphi \underline{\mathsf{U}} \psi]$**:** According to our assumption, a rv-threshold t exists such that for every $t' \geq t$ we have $\mathfrak{u}^{(0\ldots t')} \models_{\mathsf{FG}} [\varphi \underline{\mathsf{U}} \psi]$. This means that for every t' there must exist a $j_{t'} \leq t$ such that $\mathfrak{u}^{(j_{t'}\ldots t')} \models_{\mathsf{FG}} \psi$ and $\forall k < j_{t'}.\mathfrak{u}^{(k\ldots t')} \models_{\mathsf{FG}} \varphi$ holds. Now, notice that although we might have different $j_{t'}$ for each t', there can be only finitely many of them (namely those less or equal t). Hence, we must have a minimal j such that for every $t' > t$ the following holds: $\mathfrak{u}^{(j\ldots t')} \models_{\mathsf{FG}} \psi$ and $\forall k < j.\mathfrak{u}^{(k\ldots t')} \models_{\mathsf{FG}} \varphi$. Hence, according to our induction hypothesis, we must have $\mathfrak{u}^{(j\ldots)} \models_\omega \psi$ and $\forall k < j.\ \mathfrak{u}^{(k\ldots)} \models_\omega \varphi$.

$[\varphi \mathsf{U} \psi]$**:** The first case is that for every $n \in \mathbb{N}$ and every $n' > n.\mathfrak{u}^{(n\ldots n')} \models_{\mathsf{FG}} \varphi$. This means that for every $n \in \mathbb{N}$ the rv-threshold for $\mathfrak{u}^{(n\ldots)}$ is one. But this implies that we can use our induction hypothesis to show that for every $n \in \mathbb{N}$, we have $\mathfrak{u}^{(n\ldots)} \models_\omega \varphi$. Thus $\mathfrak{u} \models_\omega [\varphi \mathsf{U} \psi]$ holds. Assume now that this property does not hold, i. e. for some $n \in \mathbb{N}$ and some $n < n' \in \mathbb{N}$, we have that $\mathfrak{u}^{(n\ldots n')} \not\models_{\mathsf{FG}} \varphi$. According to the grammar of $\mathsf{TL}_{\mathsf{FG}}$, φ is a TL_{G} formula, thus $\mathfrak{u}^{(n\ldots n')} \not\models_{\mathsf{G}} \varphi$ holds also. However, the safety formula of TL_{G} are evaluated in a way such that if they are evaluated to $\mathsf{0}$ for a finite prefix $\mathfrak{w}$, they are evaluated to $\mathsf{0}$ for every suffix of $\mathfrak{w}$. Hence, after position n', the first case (*) in the $\mathsf{RV}^\infty\text{–}\mathsf{TL}_{\mathsf{FG}}$ definition of $[\varphi \mathsf{U} \psi]$ is never again satisfied. This means that the second condition must be satisfied from that point on which is exactly the same as the condition used for defining $[\varphi \underline{\mathsf{U}} \psi]$. Hence we can use the same proof as for $[\varphi \underline{\mathsf{U}} \psi]$.

$\mathsf{X}\varphi$**,** $\varphi \wedge \psi$**:** are trivial and omitted here. ∎

Remark 1. The proof for the weak until operator $[\varphi \mathsf{U} \psi]$ shows why we restricted our attention to $\mathsf{TL}_{\mathsf{FG}}$ formula: we can guarantee that φ is evaluated to $\bot_{\mathsf{FG}}$ whenever a prefix is evaluated to $\bot_{\mathsf{FG}}$ only due the special syntactic requirement that φ is a safety formula, something that is missing in arbitrary LTL formulas.

Remark 2. An alternative definition for the $[\varphi \underline{\mathsf{U}} \psi]$ operator would be based on the fixpoint iteration scheme known from translating LTL to Büchi automata:

$[\mathfrak{u} \models_{\mathsf{FG}} [\varphi \underline{\mathsf{U}}\, \psi]] := [\mathfrak{u} \models_{\mathsf{FG}} (\psi \vee (\varphi \wedge \mathsf{X}[\varphi \underline{\mathsf{U}}\, \psi]))]$. Here, the $\vee$-operator is evaluated according to our breakpoint-definition. The two definitions are indeed equivalent as one can check by an induction on n. Nevertheless we preferred the one given above since it simplifies the correctness proof.

The following theorem is therefore our main result:

Theorem 1. *Given a finite prefix* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\dots\mathfrak{u}^{(n)}$ *of an infinite word* $\mathfrak{u}_\infty$*, we have* $[\mathfrak{u}_\infty \models_\omega \varphi]$ *iff* $\nexists^\infty k.\mathfrak{u}^{(0\dots k)} \not\models_{\mathsf{FG}} \varphi$ *for every* RV^∞–$\mathsf{TL}_{\mathsf{FG}}$ *formula* φ*.*

Hence, $[\mathfrak{u}_\infty \models_\omega \varphi]$ iff $\lim\limits_{n\to\infty} [\mathfrak{u}^{(0\dots n)} \models_{\mathsf{FG}} \varphi] = \top_{\mathsf{FG}}$.

4.4 The Temporal Logic RV^∞–$\mathsf{TL}_{\mathsf{GF}}$

Since $\mathsf{TL}_{\mathsf{GF}}$ is the dual class of $\mathsf{TL}_{\mathsf{FG}}$, the following definition together with the corresponding theorem is rather straightforward:

Definition 10 (Semantics of RV^∞–$\mathsf{TL}_{\mathsf{GF}}$).
Given a finite prefix $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\dots\mathfrak{u}^{(n)}$ *of an infinite word* $\mathfrak{u}_\infty$*, the semantics of* RV^∞–$\mathsf{TL}_{\mathsf{GF}}$ *is defined by*

$$[\mathfrak{u} \models_{\mathsf{GF}} \varphi] = \begin{cases} 1, & \text{if } [\mathfrak{u} \models_{\mathsf{FG}} \neg\varphi] = 0 \\ \top_{\mathsf{GF}} & \text{if } [\mathfrak{u} \models_{\mathsf{FG}} \neg\varphi] = \bot_{\mathsf{FG}} \\ \bot_{\mathsf{GF}}, & \text{if } [\mathfrak{u} \models_{\mathsf{FG}} \neg\varphi] = \top_{\mathsf{FG}} \\ 0, & \text{if } [\mathfrak{u} \models_{\mathsf{FG}} \neg\varphi] = 1 \end{cases}$$

Theorem 2. *Given a finite prefix* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\dots\mathfrak{u}^{(n)}$ *of an infinite word* $\mathfrak{u}_\infty$*, we have* $[\mathfrak{u}_\infty \models_\omega \varphi]$ *iff* $\exists^\infty k.\mathfrak{u}^{(0\dots k)} \models_{\mathsf{GF}} \varphi$ *for every* RV^∞–$\mathsf{TL}_{\mathsf{GF}}$ *formula* φ*.*

Hence, $[\mathfrak{u}_\infty \models_\omega \varphi]$ iff $\lim\limits_{n\to\infty} [\mathfrak{u}^{(1\dots n)} \models_{\mathsf{GF}} \varphi] \notin \{\bot_{\mathsf{GF}}, 0\}$. This means, that either (1) no limit exists or (2) the limit exists and is neither $\bot_{\mathsf{GF}}$ nor 0. In case (1) holds, the result of the evaluation must oscillate between the two possible truth values, hence $\top_{\mathsf{GF}}$ holds infinitely often (note that 1 is a limit of the evaluation). If (2) holds, the limit exists and is neither $\bot_{\mathsf{GF}}$ nor 0, hence either $\top_{\mathsf{GF}}$ must hold infinitely often or 1 holds from a certain point on.

4.5 The Temporal Logic RV^∞–$\mathsf{TL}_{\mathsf{Streett}}$

We now consider the most expressive logic RV^∞–$\mathsf{TL}_{\mathsf{Streett}}$ that is obtained from $\mathsf{TL}_{\mathsf{Streett}}$. Looking at the grammar of $\mathsf{TL}_{\mathsf{Streett}}$, one sees that this logic is a positive boolean combination of $\mathsf{TL}_{\mathsf{FG}}$ and $\mathsf{TL}_{\mathsf{GF}}$ formulas. Hence, in the following we assume that our formula is given in conjunctive normal form, meaning that we have a formula of the following form:

$$\bigwedge_{i=0}^{k} \left(\bigvee_{j=0}^{m} \varphi_{i,j} \vee \bigvee_{j=0}^{n} \psi_{i,j} \right)$$

where every $\varphi_j \in \mathsf{TL_{FG}}$ and every $\psi_j \in \mathsf{TL_{GF}}$. This means that for every i we have $\left(\bigvee_{j=0}^{m} \varphi_{i,j}\right) \in \mathsf{TL_{FG}}$ and $\left(\bigvee_{j=0}^{n} \psi_{i,j}\right) \in \mathsf{TL_{GF}}$. Thus, we may even assume that our formula has the form: $\bigwedge_{i=0}^{k} \varphi_i \vee \psi_i$ where $\varphi_i \in \mathsf{TL_{FG}}$ and $\psi_i \in \mathsf{TL_{GF}}$. Hence, we can restrict ourself to formulae of that type, since every formula from $\mathsf{TL_{Streett}}$ can be brought into the desired form. To formally define a semantics for these formulae, we introduce first the Streett-k class:

Definition 11. *A* $\mathsf{TL_{Streett\text{-}k}}$ *formula is a formula of the form* $\bigwedge_{i=0}^{k} \varphi_i \vee \psi_i$*, where each* $\varphi_i \in \mathsf{TL_{FG}}$ *and each* $\psi_i \in \mathsf{TL_{GF}}$*.*

Restricting our attention first to $\mathsf{TL_{Streett\text{-}1}}$-formulas, a straightforward definition for their runtime semantics is given as follows:

Definition 12 (Semantics of $\mathsf{RV^{\infty}}$–$\mathsf{TL_{Streett\text{-}1}}$). *Let* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\ldots\mathfrak{u}^{(n)} \in \Sigma^*$ *denote a finite path of length* $n+1$*. The truth value of a* $\mathsf{TL_{Streett\text{-}1}}$ *formula* $\varphi \vee \psi$ *wrt.* $\mathfrak{u}$*, denoted with* $[\mathfrak{u} \models_{\mathsf{Street\text{-}1}} \varphi]$*, is defined as follows:*

$$[\mathfrak{u} \models_{\mathsf{Street\text{-}1}} \varphi \vee \psi] = \begin{cases} 1 & \text{if } \forall \mathfrak{w} \in \Sigma^{\omega} : \mathfrak{u}\mathfrak{w} \models_{\omega} \varphi \vee \psi \\ 0 & \text{if } \forall \mathfrak{w} \in \Sigma^{\omega} : \mathfrak{u}\mathfrak{w} \not\models_{\omega} \varphi \vee \psi \\ \top_{\mathsf{FG}} & \text{if } [\mathfrak{u} \models_{\mathsf{FG}} \varphi] \\ \top_{\mathsf{GF}} & \text{if } [\mathfrak{u} \not\models_{\mathsf{FG}} \varphi] \text{ and} \\ & \quad \exists t \leq n. \begin{pmatrix} \mathfrak{u}^{(0\ldots t)} \models_{\mathsf{GF}} \psi \text{ and} \\ \forall t \leq t' < n. [\mathfrak{u} \models_{\mathsf{Street\text{-}1}} \varphi \vee \psi] \neq \top_{\mathsf{GF}} \end{pmatrix} \\ \bot & \text{else} \end{cases}$$

Hence, if from a certain point on the so-far read prefix invariantly evaluates to $\top_{\mathsf{FG}}$, we can be sure that the corresponding φ-formula from $\mathsf{TL_{FG}}$ is invariantly satisfied. If, on the other hand, this does not hold, and we have detected that at some point $t \leq n$ the following holds: $\mathfrak{u}^{(0\ldots t)} \models_{\mathsf{GF}} \psi$ and this 'good' event has not been registered, i. e. for all values between t and n we have $[\mathfrak{u} \models_{\mathsf{Street\text{-}1}} \varphi \vee \psi] \neq \top_{\mathsf{GF}}$, then this 'good' event must be reported in the current step. Accordingly, if $\top_{\mathsf{GF}}$ holds infinitely often, φ need not hold, but we know from Theorem 2 that in that case ψ holds in the limit. Hence, the following theorem immediately follows:

Theorem 3. *Given a finite prefix* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\ldots\mathfrak{u}^{(n)}$ *of an infinite word* $\mathfrak{u}_{\infty}$*, the following holds for the semantics of* $\mathsf{RV^{\infty}}$–$\mathsf{TL_{Streett\text{-}1}}$*:*

$$[\mathfrak{u}_{\infty} \models_{\omega} \varphi] \text{ iff } \begin{pmatrix} \exists^{\infty} k. [\mathfrak{u}^{(0\ldots k)} \models_{\mathsf{Street\text{-}1}} \varphi] = \top_{\mathsf{GF}} \text{ or} \\ \nexists^{\infty} k. [\mathfrak{u}^{(0\ldots k)} \models_{\mathsf{Street\text{-}1}} \varphi] \notin \{\top_{\mathsf{FG}}, 1\} \end{pmatrix}$$

Hence, $[\mathfrak{u}_{\infty} \models_{\omega} \varphi]$ holds iff $\lim_{n\to\infty} [\mathfrak{u}^{(0\ldots n)} \models_{\mathsf{Street\text{-}1}} \varphi] \notin \{\bot, 0\}$ holds which means that either no limit exists (i.e., $\top_{\mathsf{GF}}$ holds infinitely often), or the limit is in $\{1, \top_{\mathsf{FG}}, \top_{\mathsf{GF}}\}$.

Finally, we can easily generalize this result to $\mathsf{RV^{\infty}}$–$\mathsf{TL_{Streett\text{-}}k}$:

Definition 13 (Semantics of RV^{∞}–$\mathsf{TL}_{\mathsf{Streett\text{-}k}}$). *Let* $\mathfrak{u} = \mathfrak{u}^{(0)}\mathfrak{u}^{(1)}\ldots\mathfrak{u}^{(n)} \in \Sigma^*$ *denote a finite path of length* $n+1$*. The truth value of a* $\mathsf{TL}_{\mathsf{Streett\text{-}k}}$ *formula* $\bigwedge_{i=0}^{k-1} \varphi_i \vee \psi_i$ *wrt.* $\mathfrak{u}$*, denoted with* $[\mathfrak{u} \models_{\mathsf{Streett\text{-}k}} \varphi]$*, is a truth value from the domain* $(\mathbb{B}_5)^k$ *given by:*

$$[\mathfrak{u} \models_{\mathsf{Streett\text{-}k}} \varphi \vee \psi] = [\mathfrak{u} \models_{\mathsf{Streett\text{-}1}} \varphi_0 \vee \psi_0] \times \cdots \times [\mathfrak{u} \models_{\mathsf{Streett\text{-}1}} \varphi_{k-1} \vee \psi_{k-1}]$$

5 Conclusion

In this paper, we show that the semantics for LTL on finite paths used in runtime verification so-far have certain deficiencies, in particular, they do not always converge to the truth values of infinite paths. Therefore, we defined a new semantics for LTL on finite paths that is asymptotically correct in this sense. To this end, we considered the temporal logic hierarchy of Manna and Pnueli [5,16] and developed specialized semantics for each temporal logic of this hierarchy. All classes are evaluated over a different set of truth values which leads to the surprising result that for the most expressive logic $\mathsf{TL}_{\mathsf{Streett}}$ of the hierarchy, we need a n-tuple of five-valued truth values where n is the number of clauses in the conjunctive normal form of the formula. It would be interesting to investigate whether this is unavoidable. More precisely: are there formulas in $\mathsf{TL}_{\mathsf{Streett}}$ such that an asymptotically correct semantics will need at least 5^n different truth values? We speculate that this is the case and that this question is related to the Rabin/Streett index of the formula.

References

1. Armoni, R., Bustan, D., Kupferman, O., Vardi, M.: Resets vs. Aborts in Linear Temporal Logic. In: Garavel, H., Hatcliff, J. (eds.) TACAS 2003. LNCS, vol. 2619, pp. 65–80. Springer, Heidelberg (2003)
2. Bauer, A., Leucker, M., Schallhart, C.: The Good, the Bad, and the Ugly, But How Ugly Is Ugly? In: Sokolsky, O., Taşıran, S. (eds.) RV 2007. LNCS, vol. 4839, pp. 126–138. Springer, Heidelberg (2007)
3. Bauer, A., Leucker, M., Schallhart, C.: Comparing LTL semantics for runtime verification. Journal of Logic and Computation 20(3), 651–674 (2010)
4. Bauer, A., Leucker, M., Schallhart, C.: Runtime verification for LTL and TLTL. ACM Transactions on Software Engineering and Methodology (2011)
5. Chang, E., Manna, Z., Pnueli, A.: Characterization of Temporal Property Classes. In: Kuich, W. (ed.) ICALP 1992. LNCS, vol. 623, pp. 474–486. Springer, Heidelberg (1992)
6. Eisner, C., Fisman, D., Havlicek, J., Lustig, Y., McIsaac, A., van Campenhout, D.: Reasoning with Temporal Logic on Truncated Paths. In: Hunt Jr., W.A., Somenzi, F. (eds.) CAV 2003. LNCS, vol. 2725, pp. 27–39. Springer, Heidelberg (2003)
7. Emerson, E.: Temporal and modal logic. In: van Leeuwen, J. (ed.) Handbook of Theoretical Computer Science: Formal Models and Semantics, vol. B, ch.16, pp. 995–1072. Elsevier (1990)
8. Falcone, Y., Fernandez, J.-C., Mounier, L.: What can you verify and enforce at runtime? Research Report TR-2010-5, Verimag (January 2010)

9. Maler, O., Pnueli, A.: Timing Analysis of Asynchronous Circuits Using Timed Automata. In: Camurati, P.E., Eveking, H. (eds.) CHARME 1995. LNCS, vol. 987, pp. 189–205. Springer, Heidelberg (1995)
10. Miyano, S., Hayashi, T.: Alternating automata on ω-words. Theoretical Computer Science (TCS) 32, 321–330 (1984)
11. Morgenstern, A., Schneider, K., Lamberti, S.: Generating deterministic ω-automata for most LTL formulas by the breakpoint construction. In: Scholl, C., Disch, S. (eds.) Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen (MBMV), Freiburg, Germany, pp. 119–128. Shaker (2008)
12. Pnueli, A.: The temporal logic of programs. In: Foundations of Computer Science (FOCS), pp. 46–57. IEEE Computer Society, Providence (1977)
13. Pnueli, A., Zaks, A.: PSL Model Checking and Run-Time Verification Via Testers. In: Misra, J., Nipkow, T., Karakostas, G. (eds.) FM 2006. LNCS, vol. 4085, pp. 573–586. Springer, Heidelberg (2006)
14. Ruf, J., Hoffmann, D., Kropf, T., Rosenstiel, W.: Simulation-guided property checking based on a multi-valued AR-automata. In: Design, Automation and Test in Europe (DATE), Munich, Germany, pp. 742–748. ACM (2001)
15. Schneider, K.: Improving Automata Generation for Linear Temporal Logic by Considering the Automaton Hierarchy. In: Nieuwenhuis, R., Voronkov, A. (eds.) LPAR 2001. LNCS (LNAI), vol. 2250, pp. 39–54. Springer, Heidelberg (2001)
16. Schneider, K.: Verification of Reactive Systems – Formal Methods and Algorithms. Texts in Theoretical Computer Science (EATCS Series). Springer, Heidelberg (2003)