A Normal Form for Linear Temporal Equilibrium Logic*

Pedro Cabalar

Dept. Computación, University of Corunna (Spain)
cabalar@udc.es

Abstract. In previous work, the so-called Temporal Equilibrium Logic (TEL) was introduced. This formalism provides an extension of the Answer Set semantics for logic programs to arbirary theories in the syntax of Linear Temporal Logic. It has been already shown that, in the non-temporal case, arbitrary propositional theories can always be reduced to logic program rules (with disjunction and negation in the head) independently on the context. That is, logic programs constitute a normal form for the non-temporal case. In this paper we show that TEL can be similarly reduced to a normal form consisting of a set of implications (embraced by a necessity operator) quite close to logic program rules. This normal form may be useful both for a practical implementation of TEL and a simpler analysis of theoretical problems.

1 Introduction

Logic programs under the answer set (or stable model) semantics [1] have become a succesful paradigm for practical knowledge representation. The success of *Answer Set Programming* (ASP) partly comes from a combination of solid theoretical foundations with the availability of efficient solvers [2] that allowed its use for real world applications. Among these typical applications of ASP we frequently find dealing with transition systems and action theories. In this setting, the nonmonotonic reasoning capabilities of ASP play a crucial role for a suitable treatment of problems like prediction, explanation, planning or diagnostics, allowing a natural representation of default rules like the well-known *inertia default* for solving the frame problem [3]. However, the use of ASP solvers for action domains has an important limitation: it requires fixing a finite length for the sequence of transitions *a priori*, so that the program can be properly grounded. In this way, it is impossible to deal with problems like the non-existence of solution (of any length) for a given planning problem or the study of properties like the equivalence of two representations, to put a pair of examples.

A natural choice for dealing with this kind of problems is extending ASP with modal operators, as those used in Propositional Linear Temporal Logic [4] (LTL). Defining such an extension becomes quite straightforward if we start

* This research was partially supported by Spanish MEC project TIN2009-14562-C05-04 and Xunta de Galicia project INCITE08-PXIB105159PR.

from a purely logical characterisation of ASP, like the one provided by Equilibrium Logic [5,6]. Equilibrium Logic has proved to be a powerful tool for the theoretical analysis of ASP, motivating the study of strong equivalence[1] between logic programs [7], covering most syntactic extensions considered up to date, or being closely related to the conception of new definitions of stable models for arbitrary propositional [8] and first order theories [9]. Another important advantage is that its formal definition is extremely simple: it amounts to a selection criterion among models of the (monotonic) intermediate logic of *Here-and-There* (HT) [10].

An extension of Equilibrium Logic for dealing with LTL operators was first introduced in [11] under the name of *Temporal Equilibrium Logic* (TEL). This modal extension has been already used for encoding action languages [11] or for checking strong equivalence of temporal logic programs by a reduction to LTL [12]. However, the interest of TEL has mostly remained theoretical, as there does not exist any automated method for computing the temporal equilibrium models of an arbitrary modal theory yet. An important step in this direction has to do with reducing the arbitrary syntax of temporal theories into a normal form closer to logic programming rules. For instance, in the non-temporal case, it has been already proved [13] that any arbitrary propositional theory is strongly equivalent to a logic program (allowing disjunction and negation in the head), so that logic programs constitute a normal form for Equilibrium Logic. Similarly, in the case of (monotonic) LTL, an implicational clause-like normal form introduced in [14] was used for designing a temporal resolution method.

In this paper we show that TEL can be similarly reduced (under strong equivalence) to a normal form consisting of a set of implications (embraced by a necessity operator) quite close to logic program rules. The reduction into normal form starts from the structure-preserving polynomial transformation presented in [15] for the non-temporal case. This transformation has as a main feature the introduction of an auxiliary atom per each subformula in the original theory. We then combine this technique with the inductive definitions of temporal operators used for LTL in [14]. The obtained normal form considerably reduces the possible uses of modal operators and may be useful both for a future practical implementation of TEL and a simpler analysis of theoretical problems.

The rest of the paper is organised as follows. In Section 2, we introduce the (monotonic) temporal extension of HT. In the next section, we then define the models selection criterion that gives raise to TEL, providing some concepts and definitions and introducing the normal form. Section 4 details the translation and contains the proof of its correctness. Finally, Section 5 discusses related work and Section 6 concludes the paper.

[1] Two programs are strongly equivalent when they yield the same answer sets even when they are included in a common larger program or context.

2 Linear Temporal Here-and-There (THT)

The logic of *Linear Temporal Here-and-There* (THT) is defined as follows. We start from a finite set of atoms V called the *propositional signature*. A (temporal) *formula* is defined as any combination of the classical connectives $\wedge, \vee, \rightarrow, \bot$ with the the (binary) temporal operators $\mathcal{U}$ (read "until") and $\mathcal{R}$ (read "release") and the atoms in V. A formula is said to be *non-modal* if it does not contain temporal operators. Negation is defined as $\neg\varphi \stackrel{\text{def}}{=} \varphi \rightarrow \bot$ whereas $\top \stackrel{\text{def}}{=} \neg\bot$. As usual, $\varphi \leftrightarrow \psi$ stands for $(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$. Other usual temporal operators can be defined in terms of $\mathcal{U}$ and $\mathcal{R}$ as follows:

$$\Box\varphi \stackrel{\text{def}}{=} \bot \,\mathcal{R}\, \varphi \qquad \Diamond\varphi \stackrel{\text{def}}{=} \top \,\mathcal{U}\, \varphi \qquad \varphi \,\mathcal{W}\, \psi \stackrel{\text{def}}{=} (\varphi \,\mathcal{U}\, \psi) \vee \Box\varphi$$

Given a formula Γ, by $size(\Gamma)$ we understand the number of occurrences of atoms and connectives $\wedge, \vee, \rightarrow, \bot, \bigcirc, \mathcal{U}, \mathcal{R}$ in Γ. When Γ is a finite theory, we assume we deal with the conjunction of all its formulas. For any theory Γ, $subf(\Gamma)$ will denote the set of all subformulas of Γ.

A (temporal) *interpretation* $\mathbf{M}$ is an infinite sequence of pairs $m_i = \langle H_i, T_i\rangle$ with $i = 0, 1, 2, \ldots$ where $H_i \subseteq T_i$ are sets of atoms standing for *here* and *there* respectively. For simplicity, given a temporal interpretation, we write $\mathbf{H}$ (resp. $\mathbf{T}$) to denote the sequence of pair components $H_0, H_1, \ldots$ (resp. $T_0, T_1, \ldots$). Using this notation, we will sometimes abbreviate the interpretation as $\mathbf{M} = \langle \mathbf{H}, \mathbf{T}\rangle$. An interpretation $\mathbf{M} = \langle \mathbf{H}, \mathbf{T}\rangle$ is said to be *total* when $\mathbf{H} = \mathbf{T}$.

Let $\mathbf{M}$ be an interpretation for a signature U and let $V \subset U$. The expression $\mathbf{M} \cap V$ denotes the interpretation $\mathbf{M}$ restricted to signature V, that is $\mathbf{M} \cap V$ is a sequence of pairs $\langle H_i \cap V, T_i \cap V\rangle$ for any $\langle H_i, T_i\rangle$ with $i \geq 0$ in $\mathbf{M}$.

Given an interpretation $\mathbf{M}$ and an integer number $k > 0$, by $\mathbf{M}_k$ we denote a new interpretation that results from "shifting" $\mathbf{M}$ in k positions, that is, the sequence of pairs $\langle H_k, T_k\rangle, \langle H_{k+1}, T_{k+1}\rangle, \langle H_{k+2}, T_{k+2}\rangle, \ldots$ Note that $\mathbf{M}_0 = \mathbf{M}$.

Definition 1 (satisfaction). *An interpretation* $\mathbf{M} = \langle \mathbf{H}, \mathbf{T}\rangle$ *satisfies a formula* φ*, written* $\mathbf{M} \models \varphi$*, when:*

1. $\mathbf{M} \models p$ *if* $p \in H_0$*, for any atom* p*.*
2. $\mathbf{M} \models \varphi \wedge \psi$ *if* $\mathbf{M} \models \varphi$ *and* $\mathbf{M} \models \psi$*.*
3. $\mathbf{M} \models \varphi \vee \psi$ *if* $\mathbf{M} \models \varphi$ *or* $\mathbf{M} \models \psi$*.*
4. $\langle \mathbf{H}, \mathbf{T}\rangle \models \varphi \rightarrow \psi$ *if* $\langle x, \mathbf{T}\rangle \not\models \varphi$ *or* $\langle x, \mathbf{T}\rangle \models \psi$ *for all* $x \in \{\mathbf{H}, \mathbf{T}\}$*.*
5. $\mathbf{M} \models \bigcirc\varphi$ *if* $\mathbf{M}_1 \models \varphi$*.*
6. $\mathbf{M} \models \varphi \,\mathcal{U}\, \psi$ *if* $\exists j \geq 0$*,* $\mathbf{M}_j \models \psi$ *and* $\forall k$ *s.t.* $0 \leq k < j$*,* $\mathbf{M}_k \models \varphi$
7. $\mathbf{M} \models \varphi \,\mathcal{R}\, \psi$ *if* $\forall j \geq 0$*,* $\mathbf{M}_j \models \psi$ *or* $\exists k$ *s.t.* $0 \leq k < j$*,* $\mathbf{M}_k \models \varphi$

A formula φ is *valid* if $\mathbf{M} \models \varphi$ for any $\mathbf{M}$. A *theory* is any set of formulas. An interpretation $\mathbf{M}$ is a *model* of a theory Γ, written $\mathbf{M} \models \Gamma$, if $\mathbf{M} \models \alpha$, for all formula $\alpha \in \Gamma$.

We assume that a finite sequence $\mathbf{M} = m_1, m_2, \ldots, m_n$ is an abbreviation of an infinite sequence where the remaining elements coincide with m_n, that is, that for $i > n$, $m_i = m_n$. The logic of THT is an orthogonal combination of

the logic of HT and the (standard) linear temporal logic (LTL) [4]. When we restrict temporal interpretations to finite sequences of length 1, that is $\langle H_0, T_0\rangle$ and disregard temporal operators, we obtain the logic of HT. On the other hand, if we restrict the semantics to total interpretations, $\langle \mathbf{T}, \mathbf{T}\rangle \models \varphi$ corresponds to satisfaction of formulas $\mathbf{T} \models \varphi$ in LTL. In this sense, item 4 of Definition 1 can be rephrased as:

4'. $\langle \mathbf{H}, \mathbf{T}\rangle \models \varphi \rightarrow \psi$ if both (1) $\langle \mathbf{H}, \mathbf{T}\rangle \models \varphi$ implies $\langle \mathbf{H}, \mathbf{T}\rangle \models \psi$; and (2) $\mathbf{T} \models \varphi \rightarrow \psi$ in LTL.

Similarly $\langle \mathbf{H}, \mathbf{T}\rangle \models \varphi \leftrightarrow \psi$ if both (1) $\langle \mathbf{H}, \mathbf{T}\rangle \models \varphi$ iff $\langle \mathbf{H}, \mathbf{T}\rangle \models \psi$; and (2) $\mathbf{T} \models \varphi \leftrightarrow \psi$ in LTL. The following proposition can also be easily checked.

Proposition 1. *For any Γ and any $\mathbf{M} = \langle \mathbf{H}, \mathbf{T}\rangle$, if $\mathbf{M} \models \Gamma$ then $\mathbf{T} \models \Gamma$.* □

The next result shows that, for formulas not containing implications, equivalence in LTL and THT coincides.

Proposition 2. *Let φ and ψ be two formulas not containing implication*[2]*. Then $\varphi \leftrightarrow \psi$ is a THT tautology iff it is an LTL tautology.*

Proof. As LTL models correspond to THT total models, it is obvious that any THT tautology is an LTL tautology too. For the other direction, assume $\varphi \leftrightarrow \psi$ is LTL valid but for some intepretation $\mathbf{M} = \langle \mathbf{H}, \mathbf{T}\rangle$, $\mathbf{M} \not\models \varphi \leftrightarrow \psi$. This means that, either (i) $\mathbf{T} \models \varphi$ is not equivalent to $\mathbf{T} \models \psi$ or (ii) $\mathbf{M} \models \varphi$ is not equivalent to $\mathbf{M} \models \psi$. The former immediately contradicts that $\varphi \leftrightarrow \psi$ is an LTL tautology. So, suppose (ii) and, without loss of generality, that $\mathbf{M} \models \varphi$ but $\mathbf{M} \not\models \psi$. Looking at the definition of THT satisfaction, it is easy to observe that the only way to refer to the $\mathbf{T}$ component in $\langle \mathbf{H}, \mathbf{T}\rangle$ is via implication. Since φ and ψ do not contain implications, the $\mathbf{T}$ component is irrelevant and we conclude that for any interpretation $\mathbf{M}' = \langle \mathbf{H}, \mathbf{T}'\rangle$, $\mathbf{M}' \models \varphi$ and $\mathbf{M}' \not\models \psi$, including the case $\mathbf{M}' = \langle \mathbf{H}, \mathbf{H}\rangle$. But this means there exists a LTL intepretation $\mathbf{H}$ for which $\mathbf{H} \models \varphi$ and $\mathbf{H} \not\models \psi$ contradicting that $\varphi \leftrightarrow \psi$ is an LTL tautology. □

In particular, the following LTL valid formulas are also THT valid:

$$\varphi \,\mathcal{U}\, \psi \leftrightarrow \psi \vee (\varphi \wedge \bigcirc(\varphi \,\mathcal{U}\, \psi)) \tag{1}$$

$$\varphi \,\mathcal{R}\, \psi \leftrightarrow \psi \wedge (\varphi \vee \bigcirc(\varphi \,\mathcal{R}\, \psi)) \tag{2}$$

We can alternatively represent any interpretation $\mathbf{M} = \langle \mathbf{H}, \mathbf{T}\rangle$ by seeing each $m_i = \langle H_i, T_i\rangle$ as a three-valued mapping $m_i : V \rightarrow \{0, 1, 2\}$ so that, for any atom p, $m_i(p) = 0$ when $p \notin T_i$ (the atom is false), $m_i(p) = 2$ when $p \in H_i$ (the atom is true), and $m_i(p) = 1$ when $p \in T_i \setminus H_i$ (the atom is undefined). We can then define a valuation for any formula φ, written[3] $\mathbf{M}(\varphi)$, by similarly considering

[2] Remember that negation is a form of implication.

[3] We use the same name $\mathbf{M}$ for a temporal interpretation and for its induced three-valued valuation function – ambiguity is removed by the way in which it is applied (a structure or a function on formulas).

which formulas are satisfied by $\langle \mathbf{H}, \mathbf{T} \rangle$ (which will be assigned 2), not satisfied by $\langle \mathbf{T}, \mathbf{T} \rangle$ (which will be assigned 0) or none of the two (which will take value 1). By $\mathbf{M}_i(\varphi)$ we mean the 3-valuation of φ induced by the temporal interpretation $\mathbf{M}_i$, that is, $\mathbf{M}$ shifted i positions. From the definitions in the previous section, we can easily derive the following conditions:

1. $\mathbf{M}(p) \stackrel{\text{def}}{=} m_0(p)$
2. $\mathbf{M}(\varphi \wedge \psi) \stackrel{\text{def}}{=} min(\mathbf{M}(\varphi), \mathbf{M}(\psi)); \quad \mathbf{M}(\varphi \vee \psi) \stackrel{\text{def}}{=} max(\mathbf{M}(\varphi), \mathbf{M}(\psi))$
3. $\mathbf{M}(\varphi \rightarrow \psi) \stackrel{\text{def}}{=} \begin{cases} 2 & \text{if } \mathbf{M}(\varphi) \leq \mathbf{M}(\psi) \\ \mathbf{M}(\psi) & \text{otherwise} \end{cases}$
4. $\mathbf{M}(\bigcirc \varphi) \stackrel{\text{def}}{=} \mathbf{M}_1(\varphi)$
5. $\mathbf{M}(\varphi \; \mathcal{U} \; \psi) \stackrel{\text{def}}{=} \begin{cases} 2 \text{ if } \exists j \geq 0 : \mathbf{M}_j(\varphi) = 2 \text{ and } \forall k, 0 \leq k < j \Rightarrow \mathbf{M}_k(\psi) = 2 \\ 0 \text{ if } \forall j \geq 0 : \mathbf{M}_j(\varphi) = 0 \text{ or } \exists k, 0 \leq k < j, \mathbf{M}_k(\psi) = 0 \\ 1 \text{ otherwise} \end{cases}$
6. $\mathbf{M}(\varphi \; \mathcal{R} \; \psi) \stackrel{\text{def}}{=} \begin{cases} 2 \text{ if } \forall j \geq 0 : \mathbf{M}_j(\varphi) = 2 \text{ or } \exists k, 0 \leq k < j, \mathbf{M}_k(\psi) = 2 \\ 0 \text{ if } \exists j \geq 0 : \mathbf{M}_j(\varphi) = 0 \text{ and } \forall k, 0 \leq k < j \Rightarrow \mathbf{M}_k(\psi) = 0 \\ 1 \text{ otherwise} \end{cases}$

From their definition, the interpretation of the temporal derived operators becomes $\mathbf{M}(\Box\varphi) = min\ \{\mathbf{M}_i(\varphi) \mid i \geq 0\}$ and $\mathbf{M}(\Diamond\varphi) = max\ \{\mathbf{M}_i(\varphi) \mid i \geq 0\}$.

Under this alternative three-valued definition, an interpretation $\mathbf{M}$ *satisfies* a formula φ when $\mathbf{M}(\varphi) = 2$. It is easy to see that, for any interpretation $\mathbf{M}$, $\mathbf{M} \models \varphi \leftrightarrow \psi$ iff $\mathbf{M}(\varphi) = \mathbf{M}(\psi)$ whereas, $\mathbf{M} \models \Box(\varphi \leftrightarrow \psi)$ iff for all $i \geq 0$, $\mathbf{M}_i(\varphi) = \mathbf{M}_i(\psi)$. When $\mathbf{M} = \langle \mathbf{T}, \mathbf{T} \rangle$, its induced valuation will be just written as $\mathbf{T}(\varphi)$ and obviously becomes a two-valued function, that is $\mathbf{T}(\varphi) \in \{0, 2\}$.

Observation 1 *Given* $\mathbf{M} = \{\mathbf{H}, \mathbf{T}\}$ *and a pair of formulas* φ, ψ*, if* $\mathbf{M}(\varphi) = \mathbf{M}(\psi)$ *then also* $\mathbf{T}(\varphi) = \mathbf{T}(\psi)$. □

3 Linear Temporal Equilibrium Logic (TEL)

We can now proceed to describe the models selection criterion that defines temporal equilibrium models. Given two interpretations $\mathbf{M} = \langle \mathbf{H}, \mathbf{T} \rangle$ and $\mathbf{M}' = \langle \mathbf{H}', \mathbf{T}' \rangle$ we say that $\mathbf{M}$ is *lower than* $\mathbf{M}'$, written $\mathbf{M} \leq \mathbf{M}'$, when $\mathbf{T} = \mathbf{T}'$ and for all $i \geq 0$, $H_i \subseteq H'_i$. As usual, $\mathbf{M} < \mathbf{M}'$ stands for $\mathbf{M} \leq \mathbf{M}'$ but $\mathbf{M} \neq \mathbf{M}'$.

Definition 2 (Temporal Equilibrium Model). *An interpretation* $\mathbf{M}$ *is a* temporal equilibrium model *of a theory* Γ *if* $\mathbf{M}$ *is a total model of* Γ *and there is no other* $\mathbf{M}' < \mathbf{M}$, $\mathbf{M}' \models \Gamma$. □

Note that any temporal equilibrium model is total, that is, it has the form $\langle \mathbf{T}, \mathbf{T} \rangle$ and so can be actually seen as an interpretation $\mathbf{T}$ in the standard LTL.

By $Eq(V, \Gamma)$ we denote the set of temporal equilibrium models under signature V of a theory $\Gamma \subseteq \mathcal{L}_V$. Note that the consequence relation induced by temporal equilibrium models is nonmonotonic, as happens in the non-modal case. Thus, when dealing with equivalence of two theories, Γ_1, Γ_2, the mere coincidence of equilibrium models $Eq(V, \Gamma_1) = Eq(V, \Gamma_2)$ will not suffice for safely replacing one by each other, since they may behave in a different way in the presence of additional information. Two theories Γ_1, Γ_2 are said to be *strongly equivalent* when $Eq(V, \Gamma_1 \cup \Gamma) = Eq(V, \Gamma_1 \cup \Gamma)$ for any arbitrary theory Γ.

We will further refine this idea of strong equivalence for transformations that deal with an extended signature possibly containing auxiliary atoms.

Definition 3 (Strong faithfulness). *We say that a translation $\sigma(\Gamma) \subseteq \mathcal{L}_U$ of some theory $\Gamma \subseteq \mathcal{L}_V$ with $V \subseteq U$ is* strongly faithful *if, for any theory $\Gamma' \subseteq \mathcal{L}_V$:*

$$Eq(V, \Gamma \cup \Gamma') = \{\mathbf{M} \cap V \mid \mathbf{M} \in Eq(U, \sigma(\Gamma) \cup \Gamma')\}$$

Finally, we describe the normal form we are interested in. Given a signature V, we define a *temporal literal* as any expression in the set[4] $\{p, \bigcirc p, \neg p \mid p \in V\}$.

Definition 4 (Temporal rule). *A* temporal rule *is either:*

1. *an atom $p \in V$;*
2. *an implication like $\Box(B_1 \wedge \cdots \wedge B_n \rightarrow C_1 \vee \cdots \vee C_m)$ where the B_i and C_j are temporal literals, $n \geq 0$ and $m \geq 0$;*
3. *or an implication like $\Box(\Box p \rightarrow q)$ or like $\Box(p \rightarrow \Diamond q)$ with p, q atoms.* □

A *temporal logic program* (TLP for short) is a finite set of temporal rules.

4 Reduction to Temporal Logic Programs

The translation uses an extended signature $V_{\mathbf{L}}$ that contains an atom (a label) for each non-constant formula in the original language $\mathcal{L}_V$, that is $V_{\mathbf{L}} = \{\mathbf{L}_\varphi \mid \varphi \in \mathcal{L}_V \setminus \{\bot, \top\}\}$. For convenience, we use $\mathbf{L}_\varphi \stackrel{\text{def}}{=} \varphi$ when φ is $\top$, $\bot$ or an atom $p \in V$. This allows us to consider $V_{\mathbf{L}}$ as a superset of V. For any non-atomic formula γ, we call its *definition*, $df(\gamma)$ to:

$$df(\gamma) \stackrel{\text{def}}{=} \begin{cases} \Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\varphi \bullet \mathbf{L}_\psi) & \text{if } \gamma = (\varphi \bullet \psi) \text{ with } \bullet \in \{\wedge, \vee, \rightarrow\}; \\ \Box(\mathbf{L}_\gamma \leftrightarrow \bigcirc \mathbf{L}_\varphi) & \text{if } \gamma = \bigcirc\varphi; \\ \Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\psi \vee (\mathbf{L}_\varphi \wedge \bigcirc \mathbf{L}_\gamma)\ ) & \\ \quad \wedge \Box(\mathbf{L}_\gamma \rightarrow \Diamond \mathbf{L}_\psi) & \text{if } \gamma = (\varphi\ \mathcal{U}\ \psi); \\ \Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\psi \wedge (\mathbf{L}_\varphi \vee \bigcirc \mathbf{L}_\gamma)\ ) & \\ \quad \wedge \Box(\Box \mathbf{L}_\psi \rightarrow \mathbf{L}_\gamma) & \text{if } \gamma = (\varphi\ \mathcal{R}\ \psi). \end{cases}$$

Definition 5. *For any theory Γ in $\mathcal{L}_V$, we define the translation $\sigma(\Gamma)$ as:*

$$\sigma(\Gamma) \stackrel{\text{def}}{=} \{\mathbf{L}_\varphi \mid \varphi \in \Gamma\} \cup \bigcup_{\gamma \in subf(\Gamma)} df(\gamma)$$

[4] We do not need literals like $\neg \bigcirc p$.

That is, $\sigma(\Gamma)$ collects the labels for all the formulas in Γ plus the definitions for all the subformulas in Γ. When the main connective in γ is a derived operator $\neg, \Diamond, \Box$, after simplifying truth constants, we obtain the following $df(\gamma)$:

$$df(\gamma) = \begin{cases} \Box(\mathbf{L}_\gamma \leftrightarrow \neg \mathbf{L}_\varphi\ ) & \text{if } \gamma = \neg\varphi; \\ \Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\varphi \vee \bigcirc \mathbf{L}_\gamma\ ) \wedge \Box(\mathbf{L}_\gamma \rightarrow \Diamond \mathbf{L}_\varphi) & \text{if } \gamma = \Diamond\varphi; \\ \Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\varphi \wedge \bigcirc \mathbf{L}_\gamma\ ) \wedge \Box(\Box \mathbf{L}_\varphi \rightarrow \mathbf{L}_\gamma) & \text{if } \gamma = \Box\varphi. \end{cases}$$

Lemma 1. *Let* $\mathbf{M}$ *be a model of a theory* Γ *in* $\mathcal{L}_V$*. Then, there exists some* $\mathbf{M}'$ *such that* $\mathbf{M}' = \mathbf{M} \cap V$ *and* $\mathbf{M}' \models \sigma(\Gamma)$*.*

Proof. Take $\mathbf{M}'$ as the sequence of 3-valued mappings $\mathbf{M}' = m'_1, m'_2, \ldots$ for signature $V_{\mathbf{L}}$ so that:

$$m'_i(\mathbf{L}_\varphi) \stackrel{\text{def}}{=} \mathbf{M}_i(\varphi) \tag{3}$$

for any formula $\varphi \in \mathcal{L}_V$. When φ is an atom p, $m'_i(p) = m'_i(\mathbf{L}_p) = \mathbf{M}_i(p) = m_i(p)$ for all $i \geq 0$, thus, the valuations for atoms in $\mathbf{M}$ and $\mathbf{M}'$ coincide. This means that $\mathbf{M}' \cap V = \mathbf{M}$.

Furthermore, as $\mathbf{M} \models \Gamma$, for any $\varphi \in \Gamma$ we get $2 = \mathbf{M}(\varphi) = \mathbf{M}_0(\varphi) \stackrel{(3)}{=} m'_0(\mathbf{L}_\varphi) = \mathbf{M}'(\mathbf{L}_\varphi)$. In other words $\mathbf{M}' \models \{\mathbf{L}_\varphi \mid \varphi \in \Gamma\}$. To prove that $\mathbf{M}' \models \sigma(\Gamma)$ we remain to show that $\mathbf{M}' \models df(\gamma)$ for any $\gamma \in subf(\Gamma)$. We will show it by cases, depending on each type of subformula γ.

1. For $\gamma = (\varphi \bullet \psi)$ with $\bullet \in \{\wedge, \vee, \rightarrow\}$ we have to prove $\mathbf{M}'_i(\mathbf{L}_\gamma) = \mathbf{M}'_i(\mathbf{L}_\varphi \bullet \mathbf{L}_\psi)$ for all $i \geq 0$:

$$\begin{aligned} \mathbf{M}'_i(\mathbf{L}_\gamma) = m'_i(\mathbf{L}_{\varphi\bullet\psi}) &\stackrel{(3)}{=} \mathbf{M}_i(\varphi \bullet \psi) = f^\bullet(\mathbf{M}_i(\varphi), \mathbf{M}_i(\psi)) \\ &\stackrel{(3)}{=} f^\bullet(m'_i(\mathbf{L}_\varphi), m'_i(\mathbf{L}_\psi)) = \mathbf{M}'_i(\mathbf{L}_\varphi \bullet \mathbf{L}_\psi) \end{aligned}$$

   where $f^\bullet$ denotes, for each $\bullet \in \{\wedge, \vee, \rightarrow\}$ their corresponding three-valued mappings.
2. For $\gamma = \bigcirc\varphi$ we have:

$$\begin{aligned} \mathbf{M}'_i(\mathbf{L}_\gamma) = m'_i(\mathbf{L}_{\bigcirc\varphi}) &\stackrel{(3)}{=} \mathbf{M}_i(\bigcirc\varphi) = \mathbf{M}_{i+1}(\varphi) \\ &\stackrel{(3)}{=} m'_{i+1}(\mathbf{L}_\varphi) = \mathbf{M}'_{i+1}(\mathbf{L}_\varphi) = \mathbf{M}'_i(\bigcirc \mathbf{L}_\varphi) \end{aligned}$$

3. For $\gamma = (\varphi\ \mathcal{U}\ \psi)$ we prove first $\mathbf{M}'_i(\mathbf{L}_\gamma) = \mathbf{M}'_i(\mathbf{L}_\psi \vee (\mathbf{L}_\varphi \wedge \bigcirc \mathbf{L}_\gamma))$ for any $i \geq 0$.

$$\begin{aligned} \mathbf{M}'_i(\mathbf{L}_\gamma) &= m'_i(\mathbf{L}_{\varphi\mathcal{U}\psi}) \\ &\stackrel{(3)}{=} \mathbf{M}_i(\varphi\ \mathcal{U}\ \psi) \stackrel{(1)}{=} \mathbf{M}_i(\psi \vee \varphi \wedge \bigcirc(\varphi \mathcal{U} \psi)) \\ &= max(\ \mathbf{M}_i(\psi),\ min\ \{\mathbf{M}_i(\varphi), \mathbf{M}_{i+1}(\varphi\mathcal{U}\psi)\}\ ) \\ &\stackrel{(3)}{=} max(\ m'_i(\mathbf{L}_\psi),\ min\ \{m'_i(\mathbf{L}_\varphi), m'_{i+1}(\mathbf{L}_{\varphi\mathcal{U}\psi})\}\ ) \\ &= \mathbf{M}'_i(\mathbf{L}_\psi \vee \mathbf{L}_\varphi \wedge \bigcirc \mathbf{L}_{\varphi\mathcal{U}\psi}\ ) \end{aligned}$$

We have to prove now that $\mathbf{M}' \models \Box(\mathbf{L}_\gamma \rightarrow \Diamond\mathbf{L}_\psi)$, that is, the implication holds at any $i \geq 0$. Assume at some $i \geq 0$, $\mathbf{M}'_i \models \mathbf{L}_\gamma$. By construction of $\mathbf{M}'$ this means $\mathbf{M}_i \models \varphi \,\mathcal{U}\, \psi$. This implies $\mathbf{M}_i \models \Diamond\psi$, that is, for some $j \geq i$, $\mathbf{M}_j \models \psi$. But then, by construction of $\mathbf{M}'$ again, $\mathbf{M}'_j \models \mathbf{L}_\psi$ for some $j \geq i$, and this implies $\mathbf{M}'_i \models \Diamond\mathbf{L}_\psi$. The same reasoning can be repeated replacing $\mathbf{M}'$ by $\mathbf{T}'$, and thus $\mathbf{M}'_i \models \mathbf{L}_\gamma \rightarrow \Diamond\mathbf{L}_\psi$ for any $i \geq 0$.

4. For $\gamma = (\varphi \,\mathcal{R}\, \psi)$ the proof is completely analogous to 3 replacing the use of equivalence (1) by (2) and exchanging the roles of conjunction/min and disjunction/max. □

Lemma 2. *Let Γ be a THT theory in $\mathcal{L}_V$ and $\mathbf{M}$ a model for $\sigma(\Gamma)$. Then for any $\gamma \in subf(\Gamma)$ and any $i \geq 0$, $\mathbf{M}_i(\mathbf{L}_\gamma) = \mathbf{M}_i(\gamma)$.*

Proof. We use structural induction on γ.

1. When the subformula γ has the shape $\top$, $\bot$ or an atom p this is trivial, since $\mathbf{L}_\gamma = \gamma$ by definition.
2. When $\gamma = \varphi \bullet \psi$ for any connective $\bullet \in \{\wedge, \vee, \rightarrow\}$ then:

$$\begin{aligned}
\mathbf{M}_i(\mathbf{L}_{\varphi\bullet\psi}) &= \mathbf{M}_i(\mathbf{L}_\varphi \bullet \mathbf{L}_\psi) && \text{because } \mathbf{M} \models df(\varphi \bullet \psi)\\
&= f^\bullet(\ \mathbf{M}_i(\mathbf{L}_\varphi), \mathbf{M}_i(\mathbf{L}_\psi)\ )\\
&= f^\bullet(\mathbf{M}_i(\varphi), \mathbf{M}_i(\psi)) && \text{applying induction on } \mathbf{L}_\varphi, \mathbf{L}_\psi\\
&= \mathbf{M}_i(\varphi \bullet \psi)
\end{aligned}$$

3. When $\gamma = \bigcirc\varphi$:

$$\begin{aligned}
\mathbf{M}_i(\mathbf{L}_{\bigcirc\varphi}) &= \mathbf{M}_i(\bigcirc\mathbf{L}_\varphi) \text{ because } \mathbf{M} \models df(\bigcirc\varphi)\\
&= \mathbf{M}_{i+1}(\mathbf{L}_\varphi)\\
&= \mathbf{M}_{i+1}(\varphi) \quad \text{applying induction on } \mathbf{L}_\varphi\\
&= \mathbf{M}_i(\bigcirc\varphi)
\end{aligned}$$

4. When $\gamma = (\varphi \,\mathcal{U}\, \psi)$, if we apply structural induction using $df(\gamma)$ as we did in the previous cases, we can only prove that, for any $i \geq 0$:

$$\begin{aligned}
&\mathbf{M}_i(\mathbf{L}_\gamma)\\
&= \mathbf{M}_i(\mathbf{L}_\psi \vee \mathbf{L}_\varphi \wedge \bigcirc\mathbf{L}_\gamma) \quad \text{because } \mathbf{M} \models df(\varphi \,\mathcal{U}\, \psi)\\
&= \mathbf{M}_i(\psi \vee \varphi \wedge \bigcirc\mathbf{L}_\gamma) \qquad \text{by induction on } \mathbf{L}_\varphi, \mathbf{L}_\psi
\end{aligned} \tag{4}$$

but we cannot get rid of $\mathbf{L}_\gamma$, since γ itself is the formula to be proved in the induction step.

To prove that $\mathbf{M}_i(\mathbf{L}_\gamma) = \mathbf{M}_i(\gamma)$, we will equivalently show that $\mathbf{M}_i \models \mathbf{L}_\gamma \leftrightarrow \gamma$, showing both $\mathbf{M}_i \models \mathbf{L}_\gamma$ iff $\mathbf{M}_i \models \gamma$ and $\mathbf{T}_i \models \mathbf{L}_\gamma$ iff $\mathbf{T}_i \models \gamma$.

(a) We prove first the two directions of $\mathbf{M}_i \models \mathbf{L}_\gamma$ iff $\mathbf{M}_i \models \gamma$.
From left to right, given $\mathbf{M}_i \models \mathbf{L}_\gamma$ we get $\mathbf{M}_i \models \psi \vee \varphi \wedge \bigcirc\mathbf{L}_\gamma$ due to (4). From $\mathbf{M} \models df(\gamma)$ we also conclude $\mathbf{M}_i \models \mathbf{L}_\gamma \rightarrow \Diamond\mathbf{L}_\psi$ and thus $\mathbf{M}_i \models \Diamond\mathbf{L}_\psi$. Applying structural induction on $\mathbf{L}_\psi$, we get $\mathbf{M}_i \models \Diamond\psi$. But then, there exists $j \geq i$ such that $\mathbf{M}_j \models \psi$. Take the smallest j satisfying $\mathbf{M}_j \models \psi$, so that we further have $\mathbf{M}_k \not\models \psi$ for any k, $i \leq k < j$. We

will inductively prove that $\mathbf{M}_k \models \varphi \wedge \bigcirc \mathbf{L}_\gamma$ for all $k = i, i+1, \ldots, j-1$ which, together with $\mathbf{M}_j \models \psi$ implies $\mathbf{M}_i \models (\varphi \; \mathcal{U} \; \psi) = \gamma$. For $k = i$ we know $\mathbf{M}_i \not\models \psi$ and so $\mathbf{M}_i \models \varphi \wedge \bigcirc \mathbf{L}_\gamma$. Assume proved for k with $i \leq k < j-1$ and we want to prove it for $k+1$. By induction, $\mathbf{M}_k \models \bigcirc \mathbf{L}_\gamma$ which is equivalent to $(\mathbf{M}, k+1) \models \mathbf{L}_\gamma$. This corresponds in its turn to $(\mathbf{M}, k+1) \models \psi \vee \varphi \wedge \bigcirc \mathbf{L}_\gamma$ but as $k+1 < j$ we also have $(\mathbf{M}, k+1) \not\models \psi$ so that $(\mathbf{M}, k+1) \models \varphi \wedge \bigcirc \mathbf{L}_\gamma$.

From right to left, suppose $\mathbf{M}_i \models \gamma$, that is, $\mathbf{M}_i \models \varphi \; \mathcal{U} \; \psi$. This means there exists some $j \geq i$ such that $\mathbf{M}_j \models \psi$ and $\mathbf{M}_k \models \varphi$ for all k, $i \leq k < j$. We will inductivley show that for any $k = j, j-1, \ldots, i$, $\mathbf{M}_k \models \mathbf{L}_\gamma$ which includes the case $k = i$ we really want to prove. For $k = j$, we saw that $\mathbf{M}_j \models \psi$ and, from (4), this implies $\mathbf{M}_j \models \mathbf{L}_\gamma$. Assume proved for $k+1$ with $i \leq k < j$ and we want to prove it for k. As $i \leq k < j$, we had that $\mathbf{M}_k \models \varphi$. On the other hand, by induction $(\mathbf{M}, k+1) \models \mathbf{L}_\gamma$ and so $\mathbf{M}_k \models \bigcirc \mathbf{L}_\gamma$. Altogether, we get $\mathbf{M}_j \models \varphi \wedge \bigcirc \mathbf{L}_\gamma$ which again from (4) implies $\mathbf{M}_k \models \mathbf{L}_\gamma$.

(b) Now, we must prove $\mathbf{T}_i \models \mathbf{L}_\gamma$ iff $\mathbf{T}_i \models \gamma$. Note that, due to Observation 1, the induction hypothesis $\mathbf{M}_i(\varphi) = \mathbf{M}_i(\mathbf{L}_\varphi)$ also holds for $\mathbf{T}_i(\varphi) = \mathbf{T}_i(\mathbf{L}_\varphi)$, and the same happens with subformula ψ. Following the same reasoning as in (4) we also have $\mathbf{T}_i(\mathbf{L}_\gamma) = \mathbf{T}_i(\psi \vee \varphi \wedge \bigcirc \mathbf{L}_\gamma)$. From $\mathbf{M}' \models \mathit{df}(\gamma)$ we also conclude $\mathbf{T}_i \models \mathbf{L}_\gamma \rightarrow \Diamond \mathbf{L}_\psi$. Using these premises, it is easy to check that the previous proof of item 4a still applies when replacing $\mathbf{M}$ by $\mathbf{T}$. □

5. When $\gamma = (\varphi \; \mathcal{R} \; \psi)$, analogously to case 4 we can use $\mathit{df}(\gamma)$ to obtain, for any $i \geq 0$: $\mathbf{M}_i(\mathbf{L}_\gamma) = \mathbf{M}_i(\psi \wedge (\varphi \vee \bigcirc \mathbf{L}_\gamma))$. The rest of the proof is dual to case 4 switching the roles of $\wedge$ and $\vee$, and of '$\models$' with '$\not\models$.' □

Theorem 1. *For any theory Γ in $\mathcal{L}_V$:* $\{\mathbf{M} \mid \mathbf{M} \models \Gamma\} = \{\mathbf{M}' \cap V \mid \mathbf{M}' \models \sigma(\Gamma)\}$.

Proof. The '$\subseteq$' direction immediately follows from Lemma 1. For proving the '$\supseteq$' direction, suppose we have some $\mathbf{M}'$ model of $\sigma(\Gamma)$. This implies $\mathbf{M}' \models \{\mathbf{L}_\varphi \mid \varphi \in \Gamma\}$, i.e. $\mathbf{M}'(\mathbf{L}_\varphi) = 2$ for all $\varphi \in \Gamma$. As $\Gamma \subseteq \mathit{subf}(\Gamma)$, we can apply Lemma 2 to conclude $\mathbf{M}'_i(\mathbf{L}_\varphi) = \mathbf{M}'_i(\varphi)$ for any $i \geq 0$. But then $\mathbf{M}'_0(\varphi) = 2$ for any $\varphi \in \Gamma$, that is, $\mathbf{M}' \models \Gamma$. Finally, it follows that $\mathbf{M}' \cap V \models \Gamma$ since Γ is a theory in language $\mathcal{L}_V$. □

Clearly, including an arbitrary theory $\Gamma' \subseteq \mathcal{L}_V$ in Theorem 1 as follows:

$$\{\mathbf{M} \mid \mathbf{M} \models \Gamma \cup \Gamma'\} = \{\mathbf{M}' \cap V \mid \mathbf{M}' \models \sigma(\Gamma) \cup \Gamma'\}$$

and then taking the minimal models on both sides trivially preserves the equality. As a result:

Corollary 1. *Translation $\sigma(\Gamma)$ is strongly faithful.*

Transformation $\sigma(\Gamma)$ is obviously modular, and its polynomial complexity can be easily deduced, but is not a temporal logic program yet, as it contains nested implications. However, we can apply some simple transformations on implication,

γ	$df(\gamma)$	$df^*(\gamma)$
$\varphi \wedge \psi$	$\Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\varphi \wedge \mathbf{L}_\psi)$	$\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\varphi)$ $\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\psi)$ $\Box(\mathbf{L}_\varphi \wedge \mathbf{L}_\psi \rightarrow \mathbf{L}_\gamma)$
$\varphi \vee \psi$	$\Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\varphi \vee \mathbf{L}_\psi)$	$\Box(\mathbf{L}_\varphi \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\psi \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\varphi \vee \mathbf{L}_\psi)$
$\varphi \rightarrow \psi$	$\Box(\mathbf{L}_\gamma \leftrightarrow (\mathbf{L}_\varphi \rightarrow \mathbf{L}_\psi)$	$\Box(\mathbf{L}_\gamma \wedge \mathbf{L}_\varphi \rightarrow \mathbf{L}_\psi)$ $\Box(\neg\mathbf{L}_\varphi \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\psi \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\varphi \vee \neg\mathbf{L}_\psi \vee \mathbf{L}_\gamma)$
$\varphi \mathcal{U} \psi$	$\Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\psi \vee \mathbf{L}_\varphi \wedge \bigcirc\mathbf{L}_\gamma)$ $\wedge \Box(\mathbf{L}_\gamma \rightarrow \Diamond\mathbf{L}_\psi)$	$\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\psi \vee \mathbf{L}_\varphi)$ $\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\psi \vee \bigcirc\mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\psi \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\varphi \wedge \bigcirc\mathbf{L}_\gamma \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\gamma \rightarrow \Diamond\mathbf{L}_\psi)$
$\varphi \mathcal{R} \psi$	$\Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\psi \wedge (\mathbf{L}_\varphi \vee \bigcirc\mathbf{L}_\gamma))$ $\wedge \Box(\Box\mathbf{L}_\psi \rightarrow \mathbf{L}_\gamma)$	$\Box(\mathbf{L}_\psi \wedge \mathbf{L}_\varphi \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\psi \wedge \bigcirc\mathbf{L}_\gamma \rightarrow \mathbf{L}_\gamma)$ $\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\psi)$ $\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\varphi \vee \bigcirc\mathbf{L}_\gamma)$ $\Box(\Box\mathbf{L}_\psi \rightarrow \mathbf{L}_\gamma)$

Fig. 1. Transformation $\sigma^*(\gamma)$ generating a temporal logic program.

conjunction and disjunction that have been shown to be strongly equivalent at the (non-temporal) propositional level [15], and obtain a TLP without changing the signature $V_{\mathbf{L}}$. For each definition $df(\gamma)$, we define the strongly equivalent set (understood as the conjunction) of temporal logic program rules $df^*(\gamma)$ as shown in Figure 1. The temporal logic program $\sigma^*(\Gamma)$ is obtained by replacing in $\sigma(\Gamma)$ each subformula definition $df(\varphi)$ by the corresponding set of rules $df^*(\varphi)$. Note that, as $\sigma^*(\Gamma)$ is strongly equivalent to $\sigma(\Gamma)$ (under the same vocabulary) it preserves strong faithfulness with respect to Γ.

Figure 2 shows the translation that results for derived operators after applying their definitions. To illustrate the effect of σ^* consider the example theory Γ_1 just consisting of $\Box(\neg p \rightarrow q \ \mathcal{U} \ p)$. The translation $\sigma^*(\Gamma_1)$ consists of the conjunction of $\mathbf{L}_4$ plus the rules in the $df^*(\gamma)$ columns of tables in Figure 3.

Although $\sigma^*(\Gamma)$ is systematically applied on any subformula, for a practical implementation, we can frequently avoid the introduction of new labels, when the obtained expressions are already a TLP. For instance, in the example above, it would actually suffice with considering $df^*(p \ \mathcal{U} \ q)$ that introduces label $\mathbf{L}_1$ and then replacing Γ_1 with $\Box(\neg p \rightarrow \mathbf{L}_1)$. The next results shows that $\sigma^*(\Gamma)$ keeps polynomial (in fact, linear) complexity on the size of Γ.

γ	$\pi(\gamma)$	
$\neg\varphi$	$\Box(\mathbf{L}_\gamma \wedge \mathbf{L}_\varphi \rightarrow \bot)$	$\Box(\neg\mathbf{L}_\varphi \rightarrow \mathbf{L}_\gamma)$
$\Diamond\varphi$	$\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\varphi \vee \bigcirc\mathbf{L}_\gamma)$	$\Box(\mathbf{L}_\varphi \rightarrow \mathbf{L}_\gamma)$
	$\Box(\mathbf{L}_\gamma \rightarrow \Diamond\mathbf{L}_\varphi)$	$\Box(\bigcirc\mathbf{L}_\gamma \rightarrow \mathbf{L}_\gamma)$
$\Box\varphi$	$\Box(\mathbf{L}_\varphi \wedge \bigcirc\mathbf{L}_\gamma \rightarrow \mathbf{L}_\gamma)$	$\Box(\mathbf{L}_\gamma \rightarrow \mathbf{L}_\varphi)$
	$\Box(\Box\mathbf{L}_\varphi \rightarrow \mathbf{L}_\gamma)$	$\Box(\mathbf{L}_\gamma \rightarrow \bigcirc\mathbf{L}_\gamma)$

Fig. 2. Transformation $\sigma^*(\gamma)$ that results for derived operators.

γ	$df^*(\gamma)$
$p\ \mathcal{U}\ q$	$\Box(\mathbf{L}_1 \rightarrow p \vee q)$
	$\Box(\mathbf{L}_1 \rightarrow p \vee \bigcirc\mathbf{L}_1)$
	$\Box(p \rightarrow \mathbf{L}_1)$
	$\Box(q \wedge \bigcirc\mathbf{L}_1 \rightarrow \mathbf{L}_1)$
	$\Box(\mathbf{L}_1 \rightarrow \Diamond p)$
$\neg p$	$\Box(\mathbf{L}_2 \wedge p \rightarrow \bot)$
	$\Box(\neg p \rightarrow \mathbf{L}_2)$

γ	$df^*(\gamma)$
$\neg p \rightarrow p\ \mathcal{U}\ q$	$\Box(\mathbf{L}_3 \wedge \mathbf{L}_2 \rightarrow \mathbf{L}_1)$
	$\Box(\neg\mathbf{L}_2 \rightarrow \mathbf{L}_3)$
	$\Box(\mathbf{L}_1 \rightarrow \mathbf{L}_3)$
	$\Box(\mathbf{L}_2 \vee \neg\mathbf{L}_1 \vee \mathbf{L}_3)$
$\Box(\neg p \rightarrow q\ \mathcal{U}\ p)$	$\Box(\mathbf{L}_3 \wedge \bigcirc\mathbf{L}_4 \rightarrow \mathbf{L}_4)$
	$\Box(\mathbf{L}_4 \rightarrow \mathbf{L}_3)$
	$\Box(\mathbf{L}_4 \rightarrow \bigcirc\mathbf{L}_4)$
	$\Box(\Box\mathbf{L}_3 \rightarrow \mathbf{L}_4)$

Fig. 3. Transformation $\sigma^*(\Gamma_1)$ for example theory $\Gamma_1 = \{\Box(\neg p \rightarrow q\ \mathcal{U}\ p)\}$.

Theorem 2. *Translation $\pi(\Gamma)$ is linear and its size can be bounded as follows: $size(\sigma^*(\Gamma)) \leq 2\ |\Gamma| + 34\ size(\Gamma)$.*

Proof. Theory $\sigma^*(\Gamma)$ can be written as $\left(\bigwedge_{\gamma\in\Gamma} \mathbf{L}_\gamma\right) \wedge df^*(\Gamma)$. For the size of the first conjunct, we have an atom plus a conjunction connective per each formula in Γ (this also includes the last $\wedge$ connecting to $df^*(\Gamma)$), so we get $2\ |\Gamma|$. The second conjunct, $df^*(\Gamma)$, corresponds to the conjunction of all $df^*(\gamma)$ per each subformula γ in Γ. In the worst case, operators $\mathcal{U}$ and $\mathcal{R}$, we have 5 temporal rules using a total of 13 atom occurrences and 16 connectives. These 5 rules will be joined by 4 implicit conjunctions, and we can use an additional one to join them to the rest of subformulas. Thus, we obtain $size(df^*(\Gamma)) \leq 34\ |subf(\Gamma)|$. Finally, observe that the number of subformulas can be bounded by $size(\Gamma)$ (it will be strictly lower only if repeated subformulas occur). □

5 Discussion and related work

It is perhaps interesting to compare the obtained TLP form to the so-called *Separated Normal Form* (SNF) previously introduced in [14] for the case of LTL.

An LTL formula is in SNF if it has one of the following forms[5]:

1.	$C_1 \vee \cdots \vee C_n$	an *initial* $\Box$-rule
2.	$\Box(B_1 \vee \cdots \vee B_m \rightarrow \bigcirc C_1 \vee \cdots \vee \bigcirc C_n)$	a *global* $\Box$-rule
3.	$\Diamond C$	an *initial* $\Diamond$-rule
4.	$\Box(B_1 \vee \cdots \vee B_m \rightarrow \Diamond C)$	a *global* $\Diamond$-rule

where B_i, C_j, B and C are (non-modal) literals (that is, an atom or its negation). Apart from the minor difference in initial rules, which can be easily removed in favour of auxiliary atoms, we can observe that the main difference is that the bodies of rules do not contain modal operators, whereas the heads always refer to a modal operator, either a disjunction of $\bigcirc C_i$'s or a single $\Diamond C$. In LTL, an obvious way for obtaining SNF from our TLP form would be moving any $\bigcirc B$ in the body to $\bigcirc \neg B$ in the head, and vice versa, moving any non-modal literal C in the head to $\neg C$ in the body, removing double negations afterwards. For instance, a TLP rule like $\Box(\neg p \wedge \bigcirc q \rightarrow r \vee \bigcirc s)$ becomes the LTL-equivalent SNF global $\Box$-rule $\Box(\neg p \wedge \neg r \rightarrow \bigcirc s \vee \bigcirc \neg q)$. For the case of rules like $\Box(\Box p \rightarrow q)$ we could similarly transform them into the SNF global $\Diamond$-rule: $\Box(\neg q \rightarrow \Diamond \neg p)$. Unfortunately, in the logic of HT (even in the non-modal case) exchanging literals between the body and the head in this way is not generally possible. This is because, in this logic, $\neg\neg C \leftrightarrow C$ is not valid. As a result, we cannot replace, for instance $\Box C$ with $\neg\Diamond\neg C$ and we must maintain rules $\Box(p \rightarrow \Diamond q)$ and their dual $\Box(\Box p \rightarrow q)$. By this same reason, in THT it is not possible to define operator $\mathcal{R}$ in terms of $\mathcal{U}$ or vice versa, by just applying De Morgan laws.

Another interesting question is why the $\mathcal{U}$ operator cannot be simply encoded with the formula $\Box(\mathbf{L}_\gamma \leftrightarrow \mathbf{L}_\psi \vee \mathbf{L}_\varphi \wedge \bigcirc \mathbf{L}_\gamma)$ that results from applying the inductive definition (1). The reason is that, in this way, we could infinitely make the auxiliary atom $\mathbf{L}_\gamma$ true without guaranteeing that at some finite future $\mathbf{L}_\psi$ is made true. The latter is accomplished by the formula $\Box(\mathbf{L}_\gamma \rightarrow \Diamond \mathbf{L}_\psi)$. Thus, we cannot get rid of $\Diamond$ operator in the rule heads (as happens in SNF too). The explanation for the $\mathcal{R}$ operator and the use of $\Box$ in the body is completely dual.

6 Conclusions

We have introduced a normal form for Temporal Equilibrium Logic, a formalism that provides an answer set semantics for arbitrary theories in the syntax of propositional linear temporal logic. This normal form, called Temporal Logic Programs, is close to logic programming rules (with disjunction and negation in the head), embraced with necessity operators. As a result, we can disregard the arbitrary nesting of temporal operators, or even the whole use of operators like "until" and "release." Besides, the close similarity of the obtained form to standard logic programming may help in the future to apply well-known techniques like the use of *loop formulas* [16] or the technique of *splitting* [17] to (some families of) temporal programs.

[5] For comparison purposes, we have adapted the original formulation that dealt with both future and past operators, to the case in which only future operators are used.

References

1. Gelfond, M., Lifschitz, V.: The stable model semantics for logic programming. In Kowalski, R.A., Bowen, K.A., eds.: Logic Programming: Proc. of the Fifth International Conference and Symposium (Volume 2). MIT Press, Cambridge, MA (1988) 1070–1080
2. Denecker, M., Vennekens, J., Bond, S., Gebser, M., Truszczyński, M.: The second Answer Set Programming competition. In: Proc. of the 10th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'09). Lecture Notes in Artificial Intelligence (5753), Springer-Verlag (2009) 637–654
3. McCarthy, J., Hayes, P.: Some philosophical problems from the standpoint of artificial intelligence. Machine Intelligence Journal **4** (1969) 463–512
4. Manna, Z., Pnueli, A.: The Temporal Logic of Reactive and Concurrent Systems: Specification. Springer-Verlag (1991)
5. Pearce, D.: A new logical characterisation of stable models and answer sets. In: Non monotonic extensions of logic programming. Proc. NMELP'96. (LNAI 1216). Springer-Verlag (1996)
6. Pearce, D.: Equilibrium logic. Annals of Mathematics and Artificial Intelligence **47**(1-2) (2006) 3–41
7. Lifschitz, V., Pearce, D., Valverde, A.: Strongly equivalent logic programs. Computational Logic **2**(4) (2001) 526–541
8. Ferraris, P.: Answer sets for propositional theories. In: Proc. of the 8th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'05). Lecture Notes in Computer Science (vol. 3662). (2005) 119–131
9. Ferraris, P., Lee, J., Lifschitz, V.: A new perspective on stable models. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07). (2007) 372–379
10. Heyting, A.: Die formalen Regeln der intuitionistischen Logik. Sitzungsberichte der Preussischen Akademie der Wissenschaften, Physikalisch-mathematische Klasse (1930) 42–56
11. Cabalar, P., Vega, G.P.: Temporal equilibrium logic: a first approach. In: Proc. of the 11th International Conference on Computer Aided Systems Theory, (EUROCAST'07). Lecture Notes in Computer Science (4739). (2007) 241–248
12. Aguado, F., Cabalar, P., Pérez, G., Vidal, C.: Strongly equivalent temporal logic programs. In: Proc. of the 11th European Conference on Logics in Artificial Intelligence (JELIA'08). Lecture Notes in Computer Science (vol. 5293). (2008) 8–20
13. Cabalar, P., Ferraris, P.: Propositional theories are strongly equivalent to logic programs. Theory and Practice of Logic Programming **7**(6) (2007) 745–759
14. Fisher, M.: A resolution method for temporal logic. In: Proceedings of the 12th international joint conference on Artificial Intelligence (IJCAI'91), Morgan Kaufmann Publishers Inc. (1991) 99–104
15. Cabalar, P., Valverde, A., Pearce, D.: Reducing propositional theories in equilibrium logic to logic programs. In: Proc. of the 12th Portuguese Conference on Artificial Intelligence (EPIA'05). Lecture Notes in Computer Science (3803), Springer-Verlag (2005) 4–17
16. Lin, F., Zhao, Y.: ASSAT: Computing answer sets of a logic program by sat solvers. In: Artificial Intelligence. (2002) 112–117
17. Lifschitz, V., Turner, H.: Splitting a logic program. In: Proceedings of the 11th International Conference on Logic programming (ICLP'94). (1994) 23–37