# Introducing Temporal Stable Models for Linear Dynamic Logic

**Anne-Gwenn Bosser**
ENIB, Brest,
France

**Pedro Cabalar**
University of Corunna,
Spain

**Martín Diéguez**
ENIB, Brest,
France

**Torsten Schaub**
University of Potsdam,
Germany

## Abstract

We propose a new temporal extension of the logic of Here-and-There (HT) and its equilibria obtained by combining it with dynamic logic over (linear) traces. Unlike previous temporal extensions of HT based on linear temporal logic, the dynamic logic features allow us to reason about the composition of actions. For instance, this can be used to exercise fine grained control when planning in robotics, as exemplified by GOLOG. In this paper, we lay the foundations of our approach, and refer to it as *Linear Dynamic Equilibrium Logic*, or simply DEL. We start by developing the formal framework of DEL and provide relevant characteristic results. Among them, we elaborate upon the relationships to traditional linear dynamic logic and previous temporal extensions of HT.

## Introduction

Representing and reasoning about dynamic domains is a central problem in AI, in particular, when it comes to capturing actions and change. Two traditional approaches addressing this are *Dynamic Logic* (DL; Pratt 1976) and *Linear Temporal Logic* (LTL; Pnueli 1977). Their core difference lies in the explicit representation of actions. While LTL accounts for actions implicitly by identifying them with logical atoms, they are first-class objects in DL. For example, the unloading ($l$ to $\neg l$) of a gun when shooting ($s$) could be expressed in DL or LTL as

$$l \to [s]\neg l \quad \text{and} \quad l \wedge s \to \neg l, \text{ respectively.}$$

While action $s$ is simply an atom in LTL, it becomes part of a modality in DL (similar to logical necessity $\Box$), saying that $l$ is false at the end of any state transition associated with $s$ (if $l$ is true at its start). Moreover, in DL, complex programs can be built from primitive actions via a few operators. For instance, "**while** $b$ **do** $a$" is expressed in DL as '$(b?; a)^* + \neg b?$'. Such programs can then be part of modal operators just as action $s$ above. Hence, DL provides us with a logical framework featuring procedural entities such as serial composition as well as conditional and loop constructions. For instance, in planning, this has already demonstrated its usefulness for execution monitoring and expressing control strategies, and has inspired corresponding languages like GOLOG (Levesque et al. 1997).



In fact, in planning and similar settings, the full generality of DL is not even needed, and we may confine ourselves to linear models resembling plans, or traces — just like in LTL. This observation also motivated De Giacomo and Vardi to propose a linear version of DL (2013), called LDL, for capturing actions and change.[1] Interestingly, LDL is more expressive than LTL while sharing the same complexity. In particular, LDL allows for encoding all temporal connectives of LTL but, unlike the latter, has the same expressiveness as Monadic Second Order Logic plus a linear order relation.

We draw upon LDL in what follows for defining an extension of the logic of Here-and-There (HT; Heyting 1930). We refer to the resulting logic as *(Linear) Dynamic logic of Here-and-There* (DHT for short). As usual, the equilibrium models of DHT are used to define *temporal stable models* and induce the non-monotonic counterpart of DHT, referred to as *(Linear) Dynamic Equilibrium Logic* (DEL[2]). Given that HT is the host logic of Answer Set Programming (ASP; Lifschitz 1999), our work thus lays the logical foundations of future extensions of ASP with dynamic logic features. In doing so, we actually parallel earlier work extending HT with linear temporal logic, called THT, that was originally proposed by Cabalar and Vega in (2007) and just recently led to the temporal ASP system *telingo* (Cabalar et al. 2018). In fact, we show that THT (and its equilibrium counterpart TEL) can be embedded into our new logic DHT (and DEL, respectively) — just as LTL can be put in LDL. Moreover, we prove that the satisfiability problem in DEL is EXPSPACE-complete; it thus coincides with that of TEL but goes beyond that of LDL and LTL, both being PSPACE-complete. In fact, the membership part of this result is obtained by means of an automata-based method for computing DEL models. Finally, we show that the monotonic base logic of DEL, namely DHT, allows us to decide strong equivalence in DEL; this reinforces the adequacy of the relation between both logics.

The rest of the paper is organized as follows. We start with

[1] In fact, De Giacomo and Vardi consider LDL over finite traces in (2013); we will elaborate upon this setting in future work.

[2] We hope the reader does not get confused with *Dynamic Epistemic Logic*, which shares the same acronym. We prefer to keep **D**EL as a derivation of **T**EL, similar to L**D**L from L**T**L and **D**HT from **T**HT.

the definition of the syntax of LDL under the semantics of DHT, a weaker logic than LDL that acts as monotonic basis for DEL — in fact, the latter is defined by a selection criterion on DHT models. We explain some basic properties of these formalisms, including the result showing that DHT equivalence is a necessary and sufficient condition for strong equivalence (i.e., equivalence of DEL theories under any arbitrary context). In the next section, we present a method for computing DEL models based on automata construction. Then, we illustrate the behavior of DEL on some variations of a well-known example from action theories. The next section discusses related work and, finally, we conclude the paper and include some open topics for future work.

## Linear Dynamic Equilibrium Logic

Given a set $\mathcal{P}$ of propositional variables (called *signature*), *formulas* $\varphi$ are built in LDL as follows:

$$\varphi \quad ::= \quad p \mid \bot \mid \top \mid \; [\rho]\,\varphi_1 \;\mid\; \langle\rho\rangle\,\varphi_1.$$

where $p \in \mathcal{P}$ is an atom; $\varphi_1$ and $\varphi_2$ are LDL formulas in their turn; and $[\rho]$, $\langle\rho\rangle$ are modal operators built on *path expressions* $\rho$ such that

$$\rho \quad ::= \quad \top \mid \varphi? \mid \rho_1 + \rho_2 \mid \rho_1;\rho_2 \mid \rho_1^*$$

where $\rho_1, \rho_2$ are path expressions in their turn. As we can see, each $\rho$ is a regular expression formed with the truth constant $\top$ plus the test construct $\varphi?$ typical for Propositional Dynamic Logic (Harel, Tiuryn, and Kozen 2000).

As usual, a theory is a set of formulas.

For the semantics, we start by defining a *trace* $H$ over signature $\mathcal{P}$ as a mapping $H : \mathbb{N} \rightarrow 2^{\mathcal{P}}$ that assigns a set of atoms to each natural number. Given traces $H$ and $H'$ we write $H \leq H'$ if $H(i) \subseteq H'(i)$ for each $i \geq 0$; accordingly, $H < H$ iff both $H \leq H'$ and $H \neq H'$. A *Here-and-There trace* (or HT *trace*) is a pair of traces $\langle H, T\rangle$ where $H \leq T$. An HT trace can be seen as a kind of three-valued mapping where, for each time point $i$, atoms in $H(i)$ are "*certainly true*," atoms not in $T(i)$ are "*false*" and atoms in $T(i)$ are "*potentially true*." This explains the condition $H(i) \subseteq T(i)$, meaning that anything certainly true is also potentially true. An HT trace $\langle H, T\rangle$ is said to be *total* if $H = T$, that is, the mapping becomes two-valued.

We proceed next to introduce the HT extension of LDL, we call DHT. The definition of satisfaction of formulas in DHT relies on a double induction. Given any HT trace $M$, we define the DHT satisfaction relation $\models$ in terms of an accessibility relation for path expressions $\|\,\rho\,\|^M$ whose extent depends back on $\models$.

**Definition 1** (DHT satisfaction)**.** *An* HT *trace* $\langle H, T\rangle$ *satisfies an* LDL *formula* $\varphi$ *at time point* $k \in \mathbb{N}$*, written* $\langle H, T\rangle, k \models \varphi$*, if the following conditions hold:*

1. $\langle H, T\rangle, k \models \top$ *and* $\langle H, T\rangle, k \not\models \bot$
2. $\langle H, T\rangle, k \models p$ *if* $p \in H(k)$ *for any atom* $p \in \mathcal{P}$
3. $\langle H, T\rangle, k \models \langle\rho\rangle\,\varphi$ *if* $\langle H, T\rangle, i \models \varphi$ *for some* $i$ *with* $(k, i) \in\|\,\rho\,\|^{\langle H,T\rangle}$
4. $\langle H, T\rangle, k \models [\rho]\,\varphi$ *if* $\langle H', T\rangle, i \models \varphi$ *for all* $i$ *with* $(k, i) \in\|\,\rho\,\|^{\langle H',T\rangle}$ *and all* $H' \in \{H, T\}$

*where, for any* HT *trace* $M$*,* $\|\,\rho\,\|^M \subseteq \mathbb{N}^2$ *is a relation on pairs of time points inductively defined as follows.*

5. $\|\top\|^M \stackrel{def}{=} \{\ (i, i+1)\ \mid i \in \mathbb{N}\ \}$
6. $\|\varphi?\|^M \stackrel{def}{=} \{\ (i, i) \mid M, i \models \varphi\ \}$
7. $\|\rho_1{+}\rho_2\|^M \stackrel{def}{=} \|\rho_2\|^M \cup \|\rho_2\|^M$
8. $\|\rho_1\,;\rho_2\|^M \stackrel{def}{=} \{\ (i, j) \mid (i, k) \in \|\rho_1\|^M$ *and* $(k, j) \in \|\rho_2\|^M$ *for some* $k\ \}$
9. $\|\rho^*\|^M \stackrel{def}{=} \bigcup_{n\geq 0} \|\rho^*\|^M_n$ *where*

$$\|\rho^*\|^M_0 \stackrel{def}{=} \{\ (i, i) \ \mid i \in \mathbb{N}\ \}$$
$$\|\rho^*\|^M_{n+1} \stackrel{def}{=} \|\rho^*\|^M_n \cup \{\ (i, j) \ \mid (i, k) \in \|\rho\|^M \text{ and } (k, j) \in \|\rho^*\|^M_n \text{ for some } k\} \boxtimes$$

A formula $\varphi$ is a *tautology* (or is *valid*), written $\models \varphi$, iff $M, k \models \varphi$ for any HT trace and any $k \in \mathbb{N}$. We call the logic induced by the set of all tautologies *(Linear) Dynamic logic of Here-and-There* (DHT for short). An HT trace $M$ is a *model* of an LDL theory $\Gamma$ if $M, 0 \models \varphi$ for all $\varphi \in \Gamma$. Two LDL formulas $\varphi, \psi$ are said to be *equivalent*, written $\varphi \equiv \psi$ whenever $M, k \models \varphi$ iff $M, k \models \psi$ for any HT trace $M$ and any $k \geq 0$. This is the same as requiring that $\varphi \leftrightarrow \psi$ is a tautology. Similarly, we say that two path expressions $\rho_1, \rho_2$ are *equivalent*, also written $\rho_1 \equiv \rho_2$, when they satisfy $\|\rho_1\|^M{=}\|\rho_2\|^M$ for any HT trace $M$. For instance, it is not difficult to see that $\rho^* \equiv (\top?) + (\rho; \rho^*)$.

For simplicity, we will not introduce here the semantics of LDL although, as stated by the following proposition, it just corresponds to DHT for total traces $\langle T, T\rangle$.

**Proposition 1.** *Let* $\langle H, T\rangle$ *be an* HT *trace,* $k \in \mathbb{N}$ *a time point,* $\varphi$ *be an* LDL *formula, and* $\rho$ *a path expression.*

1. $\langle T, T\rangle, k \models \varphi$ *in* DHT *iff* $T, k \models \varphi$ *as in* LDL
2. $\langle H, T\rangle, k \models \varphi$ *implies* $\langle T, T\rangle, k \models \varphi$ *(or just* $T, k \models \varphi$*)*
3. $\|\rho\|^{\langle H,T\rangle} \subseteq \|\rho\|^{\langle T,T\rangle} = \|\rho\|^T$ *where the latter represents the* LDL *accessibility relation for* $\rho$ *and* $T$. $\boxtimes$

Item 1 means that any total HT trace, i.e., of the form $\langle T, T\rangle$ can be seen as the LDL trace $T$. Moreover, it also implies that any DHT tautology is also an LDL tautology, so the former constitutes a weaker logic. In fact, it is strictly weaker, as happens with HT versus classical logic: for instance, $p \vee \neg p$ is LDL valid but not DHT valid. Items 2 and 3 represent the so-called *persistence* property from intuitionistic logic. Intuitively, this means that accessible worlds satisfy the same or more formulas than the current world, where $T$ is "accessible" from $H$ in HT. The intuitionistic reading also explains the semantics of $[\rho]\,\varphi$ which, being a kind of implication (see definition of $\rightarrow$ in next paragraph), it must be satisfied in all accessible worlds $H' \in \{H, T\}$. One more observation is that Items 1 and 2 together imply that $\varphi$ is DHT satisfiable iff it is LDL satisfiable. Since the latter is a PSPACE-complete problem (De Giacomo and Vardi 2013), the same applies to DHT satisfiability.

Boolean and LTL connectives can be defined as derived operators in the following way:

$$\varphi \wedge \psi \stackrel{def}{=} \langle \varphi ? \rangle\, \psi \qquad \varphi \rightarrow \psi \stackrel{def}{=} [\varphi ?]\, \psi$$

$$\varphi \vee \psi \stackrel{def}{=} \langle \varphi ? + \psi ? \rangle\, \top \qquad \neg\varphi \stackrel{def}{=} \varphi \rightarrow \bot$$

$$\Diamond\varphi \stackrel{def}{=} \langle \top^* \rangle\, \varphi \qquad \Box\varphi \stackrel{def}{=} [\top^*]\, \varphi$$

$$\varphi \,\mathsf{U}\, \psi \stackrel{def}{=} \langle (\varphi ?; \top)^* \rangle\, \psi \qquad \circ\varphi \stackrel{def}{=} \langle \top \rangle\, \varphi$$

$$\varphi \,\mathsf{R}\, \psi \stackrel{def}{=} (\psi \,\mathsf{U}\, (\varphi \wedge \psi)) \vee \Box\psi$$

A *propositional formula* $\phi$ is any combination of Boolean connectives $\{\bot, \top, \wedge, \vee, \rightarrow\}$ with atoms. We allow a propositional formula $\phi$ as a path expression actually standing for $(\phi?; \top)$. An LTL formula is a combination of Boolean connectives with atoms and $\{\circ, \Box, \Diamond, \mathsf{U}, \mathsf{R}\}$.

If we apply Definition 1 of DHT satisfaction to these derived connectives, we obtain the following result.

**Theorem 1.** *Let $M = \langle H, T \rangle$ be an* HT *trace and $k$ a time point $k \in \mathbb{N}$. Given the respective definitions of derived operators, we get the following satisfaction conditions:*

1. $M, k \models \varphi \wedge \psi$ *iff* $M, k \models \varphi$ *and* $M, k \models \psi$
2. $M, k \models \varphi \vee \psi$ *iff* $M, k \models \varphi$ *or* $M, k \models \psi$
3. $M, k \models \varphi \rightarrow \psi$ *iff* $\langle H', T \rangle, k \not\models \varphi$ *or* $\langle H', T \rangle, k \models \psi$, *for all* $H' \in \{H, T\}$
4. $M, k \models \neg\varphi$ *iff* $\langle T, T \rangle, k \not\models \varphi$
5. $M, k \models \circ\varphi$ *iff* $M, k{+}1 \models \varphi$
6. $M, k \models \Diamond\varphi$ *iff for some* $j \geq k$, *we have* $M, j \models \varphi$
7. $M, k \models \Box\varphi$ *iff for all* $j \geq k$, *we have* $M, j \models \varphi$
8. $M, k \models \varphi \,\mathsf{U}\, \psi$ *iff for some* $j \geq k$, *we have* $M, j \models \psi$ *and* $M, i \models \varphi$ *for all* $i \in \{k, \ldots, j{-}1\}$
9. $M, k \models \varphi \,\mathsf{R}\, \psi$ *iff for all* $j \geq k$, *we have* $M, j \models \psi$ *or* $M, i \models \varphi$ *for some* $i \in \{k, \ldots, j{-}1\}$

*as well as the relation:*

10. $\| \phi \|^M = \{ (i, i+1) \mid M, i \models \phi \}$ ⊠

An important observation is that the satisfaction conditions that we obtained in the previous result coincide with the satisfaction relation of THT defined in (Aguado et al. 2013). This immediately implies the following result.

**Corollary 1.** *Let $\varphi$ be an* LTL *formula, $M$ an* HT *trace and $k \geq 0$. Then, $M, k \models \varphi$ under* THT *satisfaction iff $M, k \models \varphi$ under* DHT *satisfaction.* ⊠

Given that both monotonic logics share the same semantic structures (HT traces) and coincide for the common syntactic fragment (LTL theories), it seems natural to maintain the same definition of *temporal equilibrium models* as in (Aguado et al. 2013), simply by using LDL formulas under the DHT satisfaction relation instead.

**Definition 2** (Temporal Equilibrium/Stable model)**.** *An* HT *trace of the form $\langle T, T \rangle$ is a* temporal equilibrium model *of an* LDL *theory $\Gamma$ iff it is a model of $\Gamma$ and there is no other model $\langle H, T \rangle$ of $\Gamma$ with $H < T$.* ⊠

If this is the case, we also say that $T$ is a *temporal stable model* of $\Gamma$.

*(Linear) Dynamic Equilibrium Logic* (DEL) is the non-monotonic logic induced by temporal equilibrium models of LDL theories. To illustrate non-monotonicity, consider the formula $[(\neg p)^*]\, q$ has a unique temporal stable model where $p$ is false and $q$ true in all the states, i.e., we conclude $\Box(\neg p \wedge q)$ in the temporal stable model. Intuitively, $\neg p$ acts as default negation and $[(\neg p)^*]$ behaves as a conditional, checking the consecutive states in which $p$ cannot be proved. Then, in all those states, $q$ is derived. Since there is no evidence for $p$ at all, $q$ is proved true in all states. Now, if we include a second formula $\Diamond p$ in the theory, we get infinite temporal stable models following the pattern $\{q\}^*\ \{p\}\ \emptyset^*$. This is because $\Diamond p$ adds one occurrence of $p$ in an arbitrary state $k \geq 0$ ($p$ is left false by default in all the rest) and, for all previous states, since they satisfy $\neg p$, the formula $[(\neg p)^*]\, q$ allows deriving $q$. As a result, the previous conclusion $\Box(\neg p \wedge q)$ is not derivable any more once we added formula $\Diamond p$.

Since Definition 2 in DEL applied to LTL theories collapses to *Temporal Equilibrium Logic*, TEL (Aguado et al. 2013), and since (Bozzelli and Pearce 2015) proved that TEL satisfiability is EXPSPACE-complete, we immediately obtain a lower bound on computational complexity:

**Corollary 2.** DEL *satisfiability is* EXPSPACE*-hard.* ⊠

As explained before, any DHT tautology is also an LDL tautology, but not vice versa. In some cases, however, we can guarantee the other direction, as stated below.

**Proposition 2.** *Let $\varphi$ and $\psi$ be* LDL *formulas without $[\cdot]$ other than those of the form $[\top^*]$ (that is, derived operator $\Box$). Remember that this rules out operators $\rightarrow$ and $\neg$, as they are derived from $[\cdot]$. Then, $\varphi \equiv \psi$ in* LDL *iff $\varphi \equiv \psi$ in* DHT*.* ⊠

Notice that this covers the whole LTL language but without implications or negation (the *release* operator R is defined in terms of $\langle\cdot\rangle$ and $\Box$). As an example dealing with LDL formulas, the following equivalences:

$$\begin{aligned} \langle \rho + \rho' \rangle\, \varphi &\equiv \langle \rho \rangle\, \varphi \vee \langle \rho' \rangle\, \varphi \\ \langle \rho; \rho' \rangle\, \varphi &\equiv \langle \rho \rangle\, \langle \rho' \rangle\, \varphi \\ \langle \rho^* \rangle\, \psi &\equiv \psi \vee \langle \rho \rangle\, \langle \rho^* \rangle\, \psi \end{aligned}$$

hold in LDL, and so, in DHT too. For the analogous properties, however, we have that:

$$\begin{aligned} [\rho + \rho']\, \varphi &\equiv [\rho]\, \varphi \wedge [\rho']\, \varphi \\ [\rho; \rho']\, \varphi &\equiv [\rho]\, [\rho']\, \varphi \\ [\rho^*]\, \psi &\equiv \psi \wedge [\rho]\, [\rho^*]\, \psi \end{aligned}$$

also hold in DHT, but we cannot use Proposition 2 to prove them, since they contain $[\cdot]$ operators. As a counterexample showing that, indeed, these operators behave differently in LDL and DHT, note that $p \vee [p?]\, \bot \equiv \top$ in LDL, since the left formula is another way of writing $p \vee \neg p$, while in DHT, as we said before, this equivalence does not hold.

One important logical feature that emerges when dealing with a non-monotonic logic is the concept of *strong equivalence* (Lifschitz, Pearce, and Valverde 2001). Under a non-monotonic inference relation, the fact that two theories $\Gamma_1$

and $\Gamma_2$ yield the same consequences is too weak to consider that one can be "safely" replaced by the other, since the addition of new information $\Gamma$ may make them behave in a different way. Instead, we normally define a stronger notion of equivalence, requiring that $\Gamma_1 \cup \Gamma$ and $\Gamma_2 \cup \Gamma$ have the same behavior, for any additional theory $\Gamma$ (providing a context). An important property proved in (Lifschitz, Pearce, and Valverde 2001) is that strong equivalence of propositional logic programs (and in fact, of arbitrary propositional theories) corresponds to regular equivalence in the monotonic logic of HT. This result reinforces the adequacy of the logic of HT as a monotonic basis for equilibrium logic and Answer Set Programming. Now, considering our setting, we can still prove that DHT plays a similar role with respect to DEL. Formally, we say that two LDL theories $\Gamma_1, \Gamma_2$ are *strongly equivalent* if $\Gamma_1 \cup \Gamma$ and $\Gamma_2 \cup \Gamma$ have the same temporal equilibrium models, for any additional LDL theory $\Gamma$. Then, we get the following result.

**Theorem 2.** *Two* LDL *theories* $\Gamma_1$ *and* $\Gamma_2$ *are strongly equivalent iff* $\Gamma_1 \equiv \Gamma_2$ *in* DHT.

*Proof.* Suppose $\Gamma_1$ and $\Gamma_2$ are DHT-equivalent and let $\Gamma$ be an arbitrary theory. Then $\Gamma_1 \cup \Gamma$ and $\Gamma_2 \cup \Gamma$ are DHT-equivalent, too. Therefore $\Gamma_1 \cup \Gamma$ and $\Gamma_2 \cup \Gamma$ have the same DHT models, and so, the same temporal equilibrium models, too.

Reciprocally, suppose that $\Gamma_1$ and $\Gamma_2$ are not DHT-equivalent.

- **First case:** $\Gamma_1$ and $\Gamma_2$ are not LDL-equivalent. Without loss of generality, there exists a total HT-trace $M = \langle T, T \rangle$ such that $M, 0 \models \Gamma_1$ but $M, 0 \not\models \Gamma_2$. Let $\Gamma_0 \stackrel{def}{=} \{[\top^*]\,(p \vee \neg p) \mid p \in \mathcal{P}\}$. It can be checked that $M$ is a temporal equilibrium model of $\Gamma_1 \cup \Gamma_0$ but not of $\Gamma_2 \cup \Gamma_0$.
- **Second case:** $\Gamma_1$ and $\Gamma_2$ are LDL-equivalent. Without loss of generality, there exists a DHT-model $M = \langle H, T \rangle$ such that $M, 0 \models \Gamma_1$ but $M, 0 \not\models \Gamma_2$. The latter means that there exists $\varphi \in \Gamma_2$ such that $M, 0 \not\models \varphi$. Let $\Gamma \stackrel{def}{=} \{\varphi \to [\top^*]\,(p \vee \neg p) \mid p \in \mathcal{P}\}$. It follows that $M, 0 \models \Gamma_1 \cup \Gamma$ because $M, 0 \not\models \varphi$ and $\langle T, T \rangle, 0 \models [\top^*]\,(p \vee \neg p)$ for all $p \in \mathcal{P}$. As a consequence, $\langle T, T \rangle$ is not an equilibrium model of $\Gamma_1 \cup \Gamma$. Since $\Gamma_1$ and $\Gamma_2$ are strongly equivalent $\langle T, T \rangle$ is not a temporal equilibrium model of $\Gamma_2 \cup \Gamma$. On the other hand, as $\Gamma_1$ and $\Gamma_2$ are LDL-equivalent, we get that $\langle T, T \rangle, 0 \models \Gamma_2 \cup \Gamma$. Since $\langle T, T \rangle$ is not a temporal equilibrium model, there exists an HT-trace $M' = \langle H', T \rangle$, with $H' < T$ such that $M', 0 \models \Gamma_2 \cup \Gamma$. Therefore, $M', 0 \models \Gamma_2$ and $M', 0 \models \Gamma$. From this, we conclude that $M', 0 \models \Gamma_0$ which amounts to requiring $H = T$ and we reach a contradiction. ⊠

## Automata-based computation of Temporal Stable Models

Connections between temporal logic and automata theory have been widely studied in the literature (Demri, Goranko, and Lange 2016). More specifically, connections between LTL and Büchi automata (Büchi 1962) have been used in (Cabalar and Demri 2011) to compute the temporal stable models of LTL formulas. This connection led to several results in complexity too. In this section, we extend the same method to the setting of DEL. We begin by extending the translation defined in (Aguado et al. 2008) to the case of LDL. Given a set of propositional variables $\mathcal{P}$ we define the *extended signature* as the set $\mathcal{P}' \stackrel{def}{=} \mathcal{P} \cup \{p' \mid p \in \mathcal{P}\}$. The idea behind this extension is that while $p \in \mathcal{P}$ encodes the satisfiability of $a$ in the "there" part ($p$ is potentially true), $p'$ would encode the satisfiability of $p$ in the "here" component ($p$ is certainly true). Both DHT formulas $\varphi$ and path expressions $\rho$ built on $\mathcal{P}$ can be encoded in ordinary LDL on $\mathcal{P}'$ by applying a rewriting transformation $(\cdot)^\bullet$ defined as follows:

- $\bot^\bullet \stackrel{def}{=} \bot$ and $\top^\bullet \stackrel{def}{=} \top$
- $p^\bullet \stackrel{def}{=} p'$, for any atom $p \in \mathcal{P}$
- $([\rho]\,\psi)^\bullet \stackrel{def}{=} [\rho]\,\psi \;\wedge\; [\rho^\bullet]\,\psi^\bullet$ and, as a consequence, $(\varphi \to \psi)^\bullet \stackrel{def}{=} (\varphi \to \psi) \;\wedge\; (\varphi^\bullet \to \psi^\bullet)$
- Finally, $(\cdot)^\bullet$ is homomorphic for the rest of connectives in formulas and in path expressions.

**Lemma 1.** *Let* $\langle H, T \rangle$ *be an* HT *trace and* $T'$ *be the* LDL *trace* $T'(i) = T(i) \cup \{p' \mid p \in H(i)\}$ *for all* $i \geq 0$. *Then, for any* $k \in \mathbb{N}$, *any* LDL *formula* $\varphi$ *and any path expression* $\rho$, *the following properties hold:*

1. $\langle H, T \rangle, k \models \varphi$ *iff* $T', k \models \varphi^\bullet$,
2. $\langle T, T \rangle, k \models \varphi$ *iff* $T', k \models \varphi$,
3. $\|\rho\|^{\langle H, T \rangle} = \|\rho^\bullet\|^{T'}$,
4. $\|\rho\|^{\langle T, T \rangle} = \|\rho\|^{T'}$. ⊠

Note that, although Lemma 1 starts from an HT trace and defines an LDL trace $T'$, we can also get the other direction, that is, we can obtain arbitrary DHT models of $\varphi$ by considering LDL models $T'$ of the formula:

$$\varphi' \stackrel{def}{=} \varphi^\bullet \wedge \bigwedge_{p \in \mathcal{P}} [\top^*]\,(p' \to p) \tag{1}$$

and making the converse mapping $H(i) = \{p \mid p' \in T'(i)\}$ and $T(i) = T'(i) \cap \mathcal{P}$. This one-to-one correspondence allows us to use several results from the literature, like the connection with Büchi automata. A Büchi automaton $\mathfrak{A}$ is a tuple $\mathfrak{A} = (Q, \Sigma, Q_0, \delta, F)$ where

1. $Q$ is a finite set of states;
2. $\Sigma$ is the language alphabet;
3. $Q_0 \subseteq Q$ is the set of initial states;
4. $\delta \subseteq Q \times \Sigma \times Q$ is the transition relation; and
5. $F \subseteq Q$ denotes the set of accepting states.

A *run* on an infinite word $w = s_1 s_2 \ldots$ (where $s_i \in \Sigma$) is a sequence $(q_0, s_0), (q_1, s_1), \ldots$ such that $(q_i, s_i, q_{i+1}) \in \delta$ for every $i \geq 0$. We say that a run $w$ is *accepting* if it starts with some initial state $q_0 \in Q_0$ and there exists at least $q \in F$ which is visited infinitely often during the run sequence. The language accepted by a Büchi automaton $\mathfrak{A}$, denoted by $L(\mathfrak{A})$, consists of the set of infinite words, $w$, for which there exists an accepting run of $\mathfrak{A}$ on $w$.

A related type of construction is the so-called *Alternating Büchi automata*, having the form $\mathfrak{A} = (Q, \Sigma, Q_0, \delta, F)$

where $Q$, $\Sigma$, $Q_0$ and $F$ are as before, but $\delta$ is now a transition function:

$$\delta : Q \times \Sigma \mapsto \mathcal{B}^+(Q)$$

that maps a state and an input symbol to some element in $\mathcal{B}^+(Q)$, which denotes the set of all positive Boolean formulas formed with states of $Q$ as atoms (that is, we can combine them with connectives $\wedge$, $\vee$, $\top$ and $\bot$). A *run* of $\mathfrak{A}$ on $w = s_0 s_1 \ldots$ has the form of a tree, a directed graph $(V, E)$ with $V \subseteq Q \times \mathbb{N}$ satisfying the following conditions:

1. it has an initial state $q_0 \in Q_0$ with $(q_0, 0) \in V$ as root
2. for all $((q, n), (q', n')) \in E$ we have $n' = n + 1$
3. for all $(q, n) \in V$, the set of successor states
$$\{q' \mid ((q, n), (q', n + 1)) \in E\}$$
satisfies the formula $\delta(q, s_n)$ specified by the transition function.

A run $(V, E)$ is *accepting* if all infinite paths (projected to $Q$) through $(V, E)$ includes infinitely many states in $F$. The language $L(\mathfrak{A})$ consists of all words that have an accepting run of $\mathfrak{A}$.

Extending the work of De Giacomo and Vardi (2013), Faymonville and Zimmermann (2017) presented a translation from LDL into (non-deterministic) Büchi Automata, we recall next:

**Lemma 2** (De Giacomo and Vardi 2013, Faymonville and Zimmermann 2017)**.** *Given an* LDL *formula $\varphi$, there is an Alternating Büchi automaton $\mathfrak{A}_\varphi$ with linearly many states in $|\varphi|$, whose accepting language corresponds to the set of models of $\varphi$.* $\boxtimes$

This result is important because it allows us to adapt the automata-based computation of temporal stable models for LTL theories (Cabalar and Demri 2011) to the case of LDL theories as described next. We begin strengthening the $\varphi'$ formula defined in (1) so we precisely capture *non-total* DHT models of $\varphi$ in LDL (under the extended signature $\mathcal{P}'$). To this aim, we define the formula:

$$\varphi_< \stackrel{def}{=} \varphi^\bullet \wedge \bigvee_{p \in \mathcal{P}} \langle \top^* \rangle \, (\neg p' \wedge p).$$

Then, the following lemmas can be easily proved.

**Lemma 3.** *The set of* LDL *models of any* LDL *formula $\varphi$ corresponds to the set of total* DHT *models of $\varphi$.* $\boxtimes$

**Lemma 4.** *Let $\varphi$ be an* LDL *formula. The set of* LDL *models of $\varphi_<$ corresponds to the set of non-total* DHT *models of $\varphi$.* $\boxtimes$

Now, we can use the automata construction from (Faymonville and Zimmermann 2017) together with Lemmas 3 and 4 to define the following two Büchi automata:

- $\mathfrak{A}_\varphi$ is built on the alphabet $\Sigma \stackrel{def}{=} 2^{\mathcal{P}}$ (i.e. propositional interpretations for signature $\mathcal{P}$) and accepts the DHT total models of $\varphi$;
- $\mathfrak{A}_{\varphi_<}$ is built on the alphabet $\Sigma' \stackrel{def}{=} 2^{\mathcal{P}'}$ (i.e. propositional interpretations for the extended signature $\mathcal{P}'$) and accepts the non-total DHT models models of $\varphi$.

In order to combine both automata, we project the language of $\mathfrak{A}_{\varphi_<}$ to $\mathcal{P}$ using the following mapping:

**Lemma 5** (Cabalar and Demri 2011)**.** *Let $h : \Sigma' \mapsto \Sigma$ be a mapping between two finite alphabets such that $h(s) = s \cap \mathcal{P}$ for any input symbol (propositional interpretation) $s \in \Sigma'$. Let us denote by $h(\mathfrak{A}_{\varphi_<})$ the result of applying $h$ to $\mathfrak{A}_{\varphi_<}$. Mapping $h(\mathfrak{A}_{\varphi_<})$ captures the* DHT *total models of $\varphi$ having an strictly smaller* DHT *model.* $\boxtimes$

Then, we obtain the following result adapted from (Cabalar and Demri 2011):

**Theorem 3.** *The set of temporal stable models of an* LDL *formula $\varphi$ corresponds to the accepting language of the Büchi automata resulting from the expression:*

$$\mathfrak{A}_\varphi \otimes \overline{h(\mathfrak{A}_{\varphi_<})},$$

*where $\overline{h(\mathfrak{A}_{\varphi_<})}$ stands for the complement of the automaton $h(\mathfrak{A}_{\varphi_<})$ and $\otimes$ represents the product of automata.* $\boxtimes$

This automata-based method can be used to determine the complexity of the DEL satisfiability problem.

**Proposition 3** (Upper bound)**.** *Checking whether an* LDL *formula has a temporal stable model can be done in* ExpSpace*.* $\boxtimes$

With this proposition and Corollary 2, we can fix the complexity of DEL satisfiability to ExpSpace.

## Yale Shooting Problem

Now that we have established the formal foundations of our approach, let us illustrate it by means of a well-known example, namely, a variation of the Yale Shooting Problem (Hanks and McDermott 1987).

A shooter can kill a turkey by loading and then shooting a gun. Action $load$ makes the gun to become $loaded$ while action $shoot$ unloads the gun and may kill the turkey or, in this version of the problem, may fail instead. A possible representation of this problem in DEL could be:

$$\begin{array}{rcll}
 & & \Box[load]loaded & (2)\\
\Box(loaded & \rightarrow & [shoot]unloaded) & (3)\\
\Box(loaded & \rightarrow & [shoot](dead \vee \neg dead)) & (4)\\
 & & \Box(load \vee \neg load) & (5)\\
 & & \Box(shoot \vee \neg shoot) & (6)
\end{array}$$

$$\begin{array}{rcll}
\Box(shoot \wedge load & \rightarrow & \bot) & (7)\\
\Box(loaded \wedge \neg \circ unloaded & \rightarrow & \circ loaded) & (8)\\
\Box(unloaded \wedge \neg \circ loaded & \rightarrow & \circ unloaded) & (9)\\
\Box(alive \wedge \neg \circ dead & \rightarrow & \circ alive) & (10)\\
\Box(dead \wedge \neg \circ alive & \rightarrow & \circ dead) & (11)\\
\Box(loaded \wedge unloaded & \rightarrow & \bot) & (12)\\
\Box(alive \wedge dead & \rightarrow & \bot) & (13)\\
alive & \wedge & unloaded & (14)
\end{array}$$

Notice that formulas of the form $p \vee \neg p$ are not tautologies, but behave as a non-deterministic choice that allow for including $p$ or not in the temporal stable model. The formulas

(5)-(7) describe the generation of actions: we can perform $load$, $shoot$ or perhaps none of them, whereas concurrent actions are not allowed. If we apply the automata-construction method on the previous theory, we get that its temporal stable models are captured by the automaton in Figure 1.

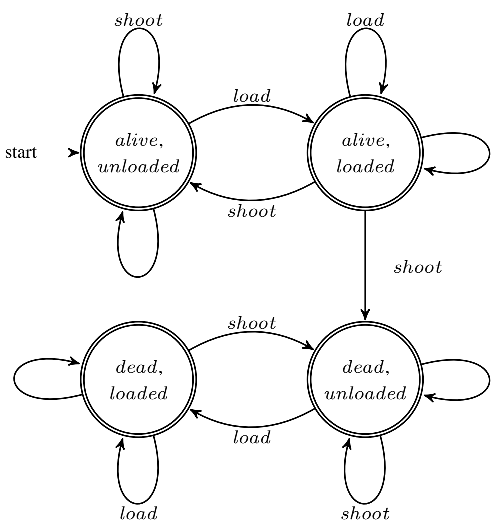


Figure 1: Büchi automaton capturing the temporal stable models of Yale Shooting example.

As an example showing the expressive capabilities of LDL expressions under DEL semantics, suppose that we replace the actions generation (5)-(7) by a program that instructs the shooter to consecutively load and shoot the gun while the turkey is $alive$. We can achieve this behavior by replacing (5)-(7) by the formula:

$$[alive^*]\,[unloaded?]\,\langle load; shoot\rangle\,\top \tag{15}$$

and the temporal stable models are displayed in Figure 2.

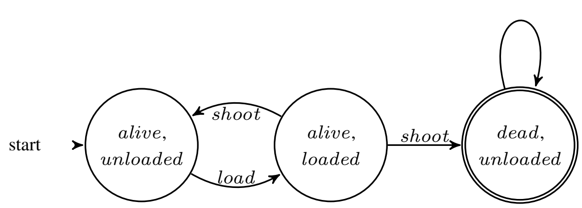


Figure 2: Büchi automaton capturing the temporal stable models of our Yale Shooting example replacing actions generation by (15).

In order to show that DEL is more expressive than TEL, suppose now that we want the shooter to shoot in (at least) every even time instant. To this aim, we keep the original theory (2)-(14) (including actions generation) but add this time the formula $[(\top;\top)^*]\, shoot$. The resulting automaton is displayed in Figure 3. To obtain a similar behavior in TEL, we would need to introduce auxiliary atoms.

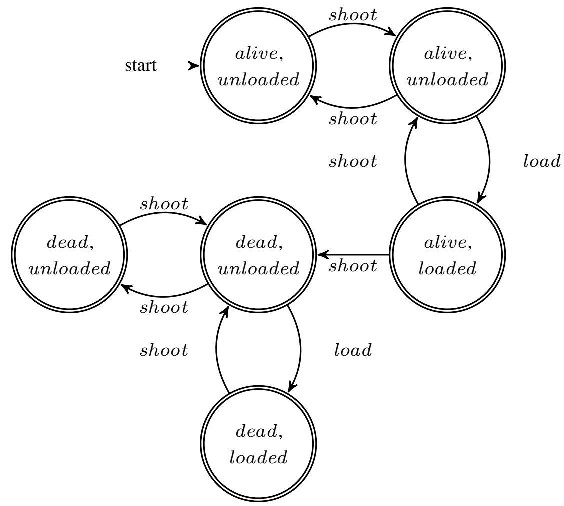


Figure 3: Büchi automaton capturing the temporal stable models of our Yale Shooting example plus the formula $[(\top;\top)^*]\, shoot$.

## Related Work

Extending logic programming (LP) with modal temporal operators is not new. This research area dates back to the late 1980s with several approaches (Gabbay 1987; Abadi and Manna 1989; Orgun and Wadge 1992; Baudinet 1992) enriching Prolog with temporal modalities from LTL – see the survey in (Orgun and Ma 1994). However, most of them impose some syntactic restrictions and disregard the use of default negation. The already mentioned TEL (Cabalar and Vega 2007) constituted the first non-monotonic semantics applicable to arbitrary LTL theories with a strong connection to LP and, in particular, to Answer Set Programming (ASP). As proved in (Cabalar, Diéguez, and Vidal 2015), propositional TEL can be seen as a (monadic) fragment of Quantified Equilibrium Logic (Pearce and Valverde 2006), the logical foundation of ASP for first order theories and can also be encoded into propositional Infinitary Equilibrium Logic (Harrison et al. 2017), also used for describing the semantics of the ASP input language (Harrison, Lifschitz, and Yang 2014). Moreover, (Cabalar, Diéguez, and Vidal 2015) also proved that TEL coincides with the temporal Prolog extension TEMPLOG (Baudinet 1992) for the common language fragment in which the latter is defined. Since we have shown that TEL is simply a fragment of DEL, all these results are still applicable to our current proposal.

Apart from approaches relying on LTL modalities, the combination of Dynamic Logic operators with ASP was already proposed in (Giordano, Martelli, and Theseider Dupré 2013). In this work, however, the underlying temporal logic was *Dynamic Linear Time Temporal Logic* (Henriksen and Thiagarajan 1999) (dLTL[3]) which, although it shares with LDL the linear structure of time points, differs from the latter in keeping the separation between actions/programs in-

[3]Since acronyms are very similar, we use initial lowercase 'd' for all formalisms related to (Henriksen and Thiagarajan 1999).

side modalities versus atoms/formulas outside them, as in the original formulation of Dynamic Logic. Additionally, dLTL does not allow the test '?' construct but allows extending the until U operator for dealing with programs. Although the approach of Giordano, Martelli, and Theseider Dupré (2013) was not defined for arbitrary dLTL theories (its semantics relied on a syntactic reduct and a fixpoint computation), Aguado, Pérez, and Vidal (2013) introduced an extension based on equilibrium logic, called *Dynamic Linear Time Equilibrium Logic* (dTEL), that covered any dLTL theory. Due to the close relation between dTEL and DEL, both in their syntax using Dynamic Logic operators and in their equilibrium-based semantics, an obvious question is whether a formal relation can be established. In the rest of the section, we show that, in fact, dTEL can also be encoded into DEL, as happened with TEL, proving in this way the high expressive power of our current proposal.

The definition of dTEL follows similar steps to the other modal extensions of equilibrium logic. It starts from a modal extension of HT, called in this case dTHT, and then defines a model selection criterion to obtain the corresponding equilibrium models. The language of dLTL is defined with respect to two disjoint sets of atoms $\mathcal{P}$ and $\mathcal{A}$, corresponding to (truth) propositions and actions. The former are used to build temporal formulas respecting the grammar:

$$\varphi ::= \bot \mid p \mid \varphi \vee \psi \mid \varphi \wedge \psi \mid \varphi \rightarrow \psi \mid \varphi \, \mathsf{U}^\pi \, \psi \mid \varphi \, \mathsf{R}^\pi \, \psi,$$

where $p \in \mathcal{P}$, $\varphi, \psi$ are dLTL formulas and $\pi$ is a test-free program as in Dynamic Logic, that is, $\pi$ is built from the grammar:

$$\pi ::= a \mid \pi_1 + \pi_2 \mid \pi_1 ; \pi_2 \mid \pi_1^*.$$

where $a \in \mathcal{A}$ and $\pi_1, \pi_2$ are programs in their turn. We write $prg(\mathcal{A})$ to denote the set of possible programs that can be built from $\mathcal{A}$. When excluding the use of binary modal operators such as $\varphi \mathsf{U}^\pi \psi$ and $\varphi \mathsf{R}^\pi \psi$, we denote the resulting sublanguage[4] as $\mathrm{dLTL}^-$.

Rather than relying on traces, the semantics of dTHT is interpreted on words built over $\mathcal{A}$. We denote by $\mathcal{A}^*$ its finite words, by $\mathcal{A}^\omega$ its infinite words, and let $\mathcal{A}^\infty = \mathcal{A}^* \cup \mathcal{A}^\omega$. Given an infinite word $\sigma \in \mathcal{A}^\omega$ of the form $\sigma = a_0 a_1 \ldots$, we denote its $i$-th symbol as $\sigma(i) \stackrel{def}{=} a_i$ and its finite prefix of length $i$ as $pref_i(\sigma) \stackrel{def}{=} a_0 a_1 \ldots a_{i-1}$. The 0-length prefix $pref_0(\sigma)$ corresponds to the empty word $\epsilon$. The set of all finite prefixes of $\sigma$ is defined as:

$$prefs(\sigma) \stackrel{def}{=} \bigcup_{i \in \mathbb{N}} pref_i(\sigma).$$

The mapping $||\cdot||$ associates to each program $\pi$ a set of finite words, viz. $||\cdot|| : prg(\mathcal{A}) \rightarrow 2^{\mathcal{A}^*}$, and is defined as follows:

- $||a|| = \{a\}$
- $||\pi_0 + \pi_1|| = ||\pi_0|| \cup ||\pi_1||$
- $||\pi_0 \, ; \pi_1|| = \{\tau_0 \tau_1 \mid \tau_0 \in ||\pi_0|| \text{ and } \tau_1 \in ||\pi_1||\}$

[4]In (Henriksen and Thiagarajan 1999) it is proved that, in the non-HT case, $\varphi \mathsf{U}^\pi \psi$ can be omitted from the language without losing expressiveness. However, no translation is provided.

- $||\pi^*|| = \bigcup_{i \in \mathbb{N}} ||\pi^i||$ where
  $||\pi^0|| = \{\epsilon\}$ and for every $i \in \mathbb{N}$
  $||\pi^{i+1}|| = \{\tau_0 \tau_1 \mid \tau_0 \in ||\pi|| \text{ and } \tau_1 \in ||\pi^i||\}$.

A dTHT *interpretation* is a structure $\langle \sigma, V_h, V_t \rangle$ where $\sigma$ denotes an infinite word and, in this setting, $V_h, V_t : prefs(\sigma) \mapsto 2^{\mathcal{P}}$ are two valuation functions satisfying $V_h(\tau) \subseteq V_t(\tau)$ for all $\tau \in prefs(\sigma)$. Let $M = \langle \sigma, V_h, V_t \rangle$ be a dTHT interpretation, $\tau \in prefs(\sigma)$ a prefix of $\sigma$ and let $\varphi$ be a dLTL formula. Then, the satisfaction relation $M, \tau \models \varphi$ is defined as follows (Henriksen and Thiagarajan 1999):

- $M, \tau \models p$ if $p \in V_h(\tau)$ for any $p \in \mathcal{P}$;
- $M, \tau \models \varphi \wedge \psi$ if $M, \tau \models \varphi$ and $M, \tau \models \psi$;
- $M, \tau \models \varphi \vee \psi$ if $M, \tau \models \varphi$ or $M, \tau \models \psi$;
- $\langle \sigma, V_h, V_t \rangle, \tau \models \varphi \rightarrow \psi$ if for all $w \in \{h, t\}$, if $\langle \sigma, V_w, V_t \rangle, \tau \models \varphi$ then $\langle \sigma, V_w, V_t \rangle, \tau \models \psi$;
- $M, \tau \models \varphi \, \mathsf{U}^\pi \, \psi$ if there exists $\tau' \in ||\pi||$ such that $\tau\tau' \in prefs(\sigma)$ and $M, \tau\tau' \models \psi$. Moreover, $M, \tau\tau'' \models \varphi$ for every $\tau''$ such that $\epsilon \leq \tau'' < \tau'$.
- $M, \tau \models \varphi \, \mathsf{R}^\pi \, \psi$ if for every $\tau' \in ||\pi||$ such that $\tau\tau' \in prefs(\sigma)$, either $M, \tau\tau' \models \psi$ or there exists $\tau''$ such that $\epsilon \leq \tau'' < \tau'$ and $M, \tau\tau'' \models \varphi$.

The following formulas are dTHT valid.

$$\neg\varphi \quad \leftrightarrow \quad \varphi \rightarrow \bot \tag{16}$$
$$\langle \rho \rangle \, \psi \quad \leftrightarrow \quad \top \mathsf{U}^\rho \psi \tag{17}$$
$$[\rho] \, \psi \quad \leftrightarrow \quad \bot \mathsf{R}^\rho \psi \tag{18}$$
$$\circ\psi \quad \leftrightarrow \quad \bigvee_{a \in \mathcal{A}} \langle a \rangle \, \psi \tag{19}$$
$$\varphi \, \mathsf{U} \, \psi \quad \leftrightarrow \quad \varphi \mathsf{U}^{\mathcal{A}^*} \psi \tag{20}$$
$$\varphi \, \mathsf{R} \, \psi \quad \leftrightarrow \quad \varphi \mathsf{R}^{\mathcal{A}^*} \psi \tag{21}$$
$$\Box\psi \quad \leftrightarrow \quad [\mathcal{A}^*] \, \psi \leftrightarrow \bot \, \mathsf{R}^{\mathcal{A}^*} \, \psi \tag{22}$$
$$\Diamond\psi \quad \leftrightarrow \quad \langle \mathcal{A}^* \rangle \, \psi \leftrightarrow \top \, \mathsf{U}^{\mathcal{A}^*} \, \psi \tag{23}$$

where, in (19), we are assuming a finite number of actions in $\mathcal{A}$.

As done before with traces, the following relations are defined between $V_h$ and $V_t$:

1. $M$ is said to be *total* if $V_h = V_t$;
2. $V_h \leq V_t$ if $V_h(\tau) \subseteq V_t(\tau)$ for all $\tau \in prefs(\sigma)$;
3. $V_h < V_t$ if $V_h \leq V_t$ and $V_h(\tau) \neq V_t(\tau)$ for some $\tau \in prefs(\sigma)$.

**Definition 3** (Aguado, Pérez, and Vidal 2013)**.** *A total temporal interpretation $M = (\sigma, V_t, V_t)$ is a* Dynamic Linear Time Equilibrium Model *of a formula $\varphi$ if $M \models \varphi$ and there is no $V_h < V_t$ such that $\langle \sigma, V_h, V_t \rangle \models \varphi$.* ⊠

Although this temporal extension of ASP has been less studied than TEL, most of the properties already considered in this paper such as the theorem of strong equivalence, complexity or automata-based computation methods can be adapted from TEL. Moreover, the following result can be easily obtained.

**Lemma 6.** *Star-free* dTEL *is equivalent to* TEL. ⊠

We define next a translation $g(\varphi)$ that transforms a dLTL formula $\varphi$ into an LDL formula and that allows us to encode dTEL in DEL. This translation is defined in combination with a second one, $g'$ that takes a dLTL program $\pi$ on $\mathcal{A}$ and an dLTL formula $\varphi$ as arguments and returns an LDL path expression $g'(\pi, \varphi)$.

**Definition 4** ($g$- and $g'$-translation)**.** *We define the translations $g$ and $g'$ as follows.*

$$
\begin{aligned}
g(p) &\overset{def}{=} p \text{ for any } p \in \mathcal{P}\\
g(\varphi \wedge \psi) &\overset{def}{=} g(\varphi) \wedge g(\psi)\\
g(\varphi \vee \psi) &\overset{def}{=} g(\varphi) \vee g(\psi)\\
g(\varphi \rightarrow \psi) &\overset{def}{=} g(\varphi) \rightarrow g(\psi)\\
g(\varphi \,\mathsf{U}^{\pi}\, \psi) &\overset{def}{=} \langle\, g'(\pi, \varphi)\,\rangle\, g(\psi)\\
g(\varphi \,\mathsf{R}^{\pi}\, \psi) &\overset{def}{=} \langle\, g'(\pi, \psi)\,\rangle\, g(\psi \wedge \varphi)\\
&\quad\ \vee [\, g'(\pi, \top)\,]\, g(\psi)
\end{aligned}
$$

$$
\begin{aligned}
g'(a, \varphi) &\overset{def}{=} g(\varphi)?; a \text{ where } a \in \mathcal{A}\\
g'(\pi_1 + \pi_2, \varphi) &\overset{def}{=} g'(\pi_1, \varphi) + g'(\pi_2, \varphi)\\
g'(\pi_1; \pi_2, \varphi) &\overset{def}{=} g'(\pi_1, \varphi); g'(\pi_2, \varphi)\\
g'(\pi^*, \varphi) &\overset{def}{=} g'(\pi, \varphi)^*
\end{aligned}
$$

Let us define now the model correspondence. Given a dTHT interpretation $M = (\sigma, V_h, V_t)$ on $\mathcal{A}$ and $\mathcal{P}$, being $\sigma$ the infinite word $a_0 a_1 \ldots$, we associate the DHT interpretation $M' = \langle H', T'\rangle$ on signature $\mathcal{A} \cup \mathcal{P}$ defined as:

$$
\begin{aligned}
H'(i) &\overset{def}{=} \{\, \sigma(i)\,\} \cup V_h(\, \mathit{pref}_i(\sigma)\,)\\
T'(i) &\overset{def}{=} \{\, \sigma(i)\,\} \cup V_t(\, \mathit{pref}_i(\sigma)\,).
\end{aligned}
$$

for all $i \in \mathbb{N}$. Note that any action $a \in \mathcal{A}$ at any situation $i \in \mathbb{N}$ will either be true both in $H'(i)$ and $T'(i)$, because it happens to be $\sigma(i) = a$, or it will be false in $H'(i)$ and $T'(i)$. As a result, the excluded middle axiom is satisfied for all actions, that is, $M', i \models a \vee \neg a$ for any $i \in \mathbb{N}$ and for all $a \in \mathcal{A}$.

This model correspondence is one to one: given any DHT interpretation $M' = \langle H', T'\rangle$ we can obtain back its corresponding dTHT interpretation $M = (\sigma, V_h, V_t)$:

$$
\begin{aligned}
\sigma(i) &\overset{def}{=} H(i) \cap \mathcal{A};\\
V_h(\mathit{pref}_i(\sigma)) &\overset{def}{=} H(i) \cap \mathcal{P};\\
V_t(\mathit{pref}_i(\sigma)) &\overset{def}{=} T(i) \cap \mathcal{P}.
\end{aligned}
$$

Based on this model correspondence, we prove the following lemmas:

**Lemma 7.** *Given a* dTHT *interpretation $M = (\sigma, V_h, V_t)$ and the corresponding* DHT *interpretation $M' = \langle H', T'\rangle$ as defined above, and given $\tau = (a_0; \cdots; a_i) \in \mathit{prefs}(\sigma)$, then, for all test-free path expressions $\rho$, the following conditions are equivalent:*

1. $(a_{i+1}; \cdots; a_j) \in \|\rho\|$ *and* $(a_0; \cdots; a_i; a_{i+1}; \cdots; a_j) \in \mathit{prefs}(\sigma)$
2. $(i, j) \in \|\,\rho\,\|^{M'}$. ⊠

**Lemma 8.** *Let $M = \langle H, T\rangle$ be a* DHT *interpretation and let $\psi$ be an* LDL *formula. For all test-free path expressions $\rho$ and for all $i, j$ satisfying $0 \leq i \leq j$, the following conditions are equivalent:*

1. $(i, j) \in \|\,\rho\,\|^{M}$ *and for all* $i \leq k < j$, $M, k \models \psi$,
2. $(i, j) \in \|\, g'(\rho, \psi)\,\|^{M}$. ⊠

**Theorem 4.** *Given a* dTHT *interpretation $M = (\sigma, V_h, V_t)$ and the corresponding* DHT *interpretation $M' = \langle H', T'\rangle$, and given $\tau_i = \mathit{pref}_i(\sigma)$, we have:*
$M, \tau_i \models \varphi$ *in* dTHT *iff* $M', i \models g(\varphi)$ *in* DHT. ⊠

This result shows that we can embed dTHT into DHT thanks to the combined $g/g'$ translation. In other words, given a dTHT formula $\varphi$ we can obtain its dTHT models by looking at the DHT models of $g(\varphi)$ using the correspondence we have established between both types of models. It is not difficult to see that if we further select the DEL-equilibrium models of $g(\varphi)$ we obtain precisely the original dTEL equilibrium models of $\varphi$. The only particularity in the models selection performed in DHT and $g(\varphi)$ is that actions $a \in \mathcal{A}$ may occur as regular atoms in $H$ or $T$ but, as we said before, they satisfy the excluded middle action (an action $a$ is both in $H(i)$ and $T(i)$ or in none of them) so they do not interfere in the model minimization. As a result, we get:

**Corollary 3.** *The set of* dTEL *equilibrium models of a* dTHT *formula $\varphi$ coincides with the set of (the corresponding)* DEL *equilibrium models of $g(\varphi)$.* ⊠

## Conclusions

We have introduced an extension of the Logic of Here-and-There with a linear version of Dynamic Logic (LDL) due to De Giacomo and Vardi (2013). The resulting *Dynamic logic of Here-and-There* (DHT for short) serves us as the (monotonic) basis of its non-monotonic extension, called *Dynamic Equilibrium Logic* (DEL). Our formal elaboration of DHT and DEL greatly benefits from our result showing that former temporal extensions of the Logic of Here-and-There and Equilibrium Logic, viz. THT and TEL, constitute fragments of our new logics. Along this paper, we thus developed an automata-based computation method for DEL and also obtained key results on strong equivalence and computational complexity. As a consequence, it turns out that DEL is more expressive than TEL while sharing the same complexity.

The obvious practical interest of our work stems from its relation to Answer Set Programming (ASP). Today's ASP solvers can be seen as implementations of Equilibrium Logic. Any propositional formula can be translated into a (disjunctive) logic program, whose answer sets correspond to the equilibrium models of the original formula. The analogous approach led recently to the temporal ASP solver *telingo* (Cabalar et al. 2018) that computes finite traces in TEL. In view of the presented results, our next steps thus foresee the definition of a normal form like in (Cabalar

2010), which is very close to logic programming, and also extend the recent result in (Cabalar et al. 2018) on TEL on finite traces to the DEL setting. This will result in a powerful system for representing and reasoning about dynamic domains, not only providing an effective implementation of DEL but, furthermore, a platform for action and control languages, like $\mathcal{A}, \mathcal{B}, \mathcal{C}$ (Gelfond and Lifschitz 1998; Giunchiglia et al. 2004) or GOLOG (Levesque et al. 1997). An encoding of GOLOG in ASP was proposed in (Son et al. 2006).

Moreover, our approach paves the way for further extensions stemming from the area of (propositional) Dynamic Logic such as Propositional Assignments (Balbiani, Herzig, and Troquard 2013), Dynamic Logic with repeating and looping (Harel and Sherman 1982), or the use of timed regular expressions (Asarin, Caspi, and Maler 2002) that could potentially be incorporated into our approach together with other variants of LDL (Weinert and Zimmermann 2016; Faymonville and Zimmermann 2017). In fact, LDL is the core of `ForSpec` (Armoni et al. 2002), a specification language widely used for model checking in industrial domains. By a simple replacement of LDL by DEL, it should be possible to extend `ForSpec` with default negation in a very natural way. This would provide, in principle, a simple method for introducing non-monotonicity in the specifications.

Following a more theoretical line of research, we also see a more extensive study of the Here-and-There basis supporting DEL as an interesting avenue of future research. Note that DHT lacks the symmetry inherent in THT because the model is involved in the DHT accessibility relation. A sound and complete axiomatic system would help us to have an alternative view of the properties inherent to DHT.

**Acknowledgments.** This work was partially supported by MINECO, Spain, grant TIC2017-84453-P, Xunta de Galicia, Spain (GPC ED431B 2016/035 and 2016-2019 ED431G/01, CITIC), the DFG grant SCHA 550/9 and the ANR-12-ASTR-0020 project.

## References

Abadi, M., and Manna, Z. 1989. Temporal logic programming. *Journal of Symbolic Computation* 8:277–295.

Aguado, F.; Cabalar, P.; Pérez, G.; and Vidal, C. 2008. Strongly Equivalent Temporal Logic Programs. In *Proc. of the 11th European Conference on Logics in Artificial Intelligence (JELIA'08)*, 8–20.

Aguado, F.; Cabalar, P.; Diéguez, M.; Pérez, G.; and Vidal, C. 2013. Temporal equilibrium logic: a survey. *Journal of Applied Non-Classical Logics* 23(1-2):2–24.

Aguado, F.; Pérez, G.; and Vidal, C. 2013. Integrating temporal extensions of answer set programming. In Cabalar, P., and Son, T., eds., *Proceedings of the Twelfth International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'13)*, volume 8148 of *Lecture Notes in Artificial Intelligence*, 23–35. Springer-Verlag.

Armoni, R.; Fix, L.; Flaisher, A.; Gerth, R.; Ginsburg, B.; Kanza, T.; Landver, A.; Mador-Haim, S.; Singerman, E.; Tiemeyer, A.; Vardi, M. Y.; and Zbar, Y. 2002. The forspec temporal logic: A new temporal property-specification language. In *TACAS*, volume 2280 of *Lecture Notes in Computer Science*, 296–211. Springer.

Asarin, E.; Caspi, P.; and Maler, O. 2002. Timed regular expressions. *J. ACM* 49(2):172–206.

Balbiani, P.; Herzig, A.; and Troquard, N. 2013. Dynamic logic of propositional assignments: A well-behaved variant of PDL. In *Proceedings of the 2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '13, 143–152. IEEE Computer Society.

Baudinet, M. 1992. A simple proof of the completeness of temporal logic programming. In Fariñas del Cerro, L., and Penttonen, M., eds., *Intensional Logics for Programming*. Clarendon Press. 51–83.

Bozzelli, L., and Pearce, D. 2015. On the complexity of temporal equilibrium logic. In *Proceedings of the Thirtieth Annual Symposium on Logic in Computer Science (LICS'15)*, 645–656. IEEE Computer Society Press.

Büchi, J. R. 1962. On a Decision Method in Restricted Second Order Arithmetic. In *International Congress on Logic, Methodology, and Philosophy of Science*, 1–11.

Cabalar, P., and Demri, S. 2011. Automata-based computation of temporal equilibrium models. In Vidal, G., ed., *Proceedings of the Twenty-first International Symposium on Logic-Based Program Synthesis and Transformation (LOPSTR'11)*, volume 7225 of *Lecture Notes in Computer Science*, 57–72. Springer-Verlag.

Cabalar, P., and Vega, G. P. 2007. Temporal equilibrium logic: A first approach. In Moreno-Díaz, R.; Pichler, F.; and Quesada-Arencibia, A., eds., *Eleventh International Conference on Computer Aided Systems Theory (EUROCAST'17)*, volume 4739 of *Lecture Notes in Computer Science*, 241–248. Springer-Verlag.

Cabalar, P.; Kaminski, R.; Schaub, T.; and Schuhmann, A. 2018. Temporal answer set programming on finite traces. *Theory and Practice of Logic Programming* 18(3-4):406–420.

Cabalar, P.; Diéguez, M.; and Vidal, C. 2015. An infinitary encoding of temporal equilibrium logic. *Theory and Practice of Logic Programming* 15(4-5):666–680.

Cabalar, P. 2010. A normal form for linear temporal equilibrium logic. In Janhunen, T., and Niemelä, I., eds., *Proceedings of the Twelfth European Conference on Logics in Artificial Intelligence (JELIA'10)*, volume 6341 of *Lecture Notes in Artificial Intelligence*, 64–76. Springer-Verlag.

De Giacomo, G., and Vardi, M. 2013. Linear temporal logic and linear dynamic logic on finite traces. In Rossi, F., ed., *Proceedings of the Twenty-third International Joint Conference on Artificial Intelligence (IJCAI'13)*, 854–860. IJCAI/AAAI Press.

Demri, S.; Goranko, V.; and Lange, M. 2016. *Temporal Logics in Computer Science: Finite-State Systems*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press.

Faymonville, P., and Zimmermann, M. 2017. Parametric linear dynamic logic. *Inf. Comput.* 253:237–256.

Gabbay, D. M. 1987. Modal and temporal logic programming. In Galton, A., ed., *Temporal Logics and their Applications*. Academic Press. chapter 6, 197–237.

Gelfond, M., and Lifschitz, V. 1998. Action languages. *Electronic Transactions on Artificial Intelligence* 3(6):193–210.

Giordano, L.; Martelli, A.; and Theseider Dupré, D. 2013. Reasoning about actions with temporal answer sets. *Theory and Practice of Logic Programming* 13(2):201–225.

Giunchiglia, E.; Lee, J.; Lifschitz, V.; McCain, N.; and Turner, H. 2004. Nonmonotonic causal theories. *Artificial Intelligence* 153(1-2):49–104.

Hanks, S., and McDermott, D. 1987. Nonmonotonic logic and temporal projection. *Artificial Intelligence* 33(3):379–412.

Harel, D., and Sherman, R. 1982. Looping vs. repeating in dynamic logic. *Information and control* 55(1-3):175–192.

Harel, D.; Tiuryn, J.; and Kozen, D. 2000. *Dynamic Logic*. MIT Press.

Harrison, A.; Lifschitz, V.; Pearce, D.; and Valverde, A. 2017. Infinitary equilibrium logic and strongly equivalent logic programs. *Artificial Intelligence* 246:22–33.

Harrison, A.; Lifschitz, V.; and Yang, F. 2014. The semantics of gringo and infinitary propositional formulas. In Baral, C.; De Giacomo, G.; and Eiter, T., eds., *Proceedings of the Fourteenth International Conference on Principles of Knowledge Representation and Reasoning (KR'14)*. AAAI Press.

Henriksen, J., and Thiagarajan, P. 1999. Dynamic linear time temporal logic. *Annals Pure and Applied Logic* 96(1-3):187–207.

Heyting, A. 1930. Die formalen Regeln der intuitionistischen Logik. In *Sitzungsberichte der Preussischen Akademie der Wissenschaften*. Deutsche Akademie der Wissenschaften zu Berlin. 42–56. Reprint in Logik-Texte: Kommentierte Auswahl zur Geschichte der Modernen Logik, Akademie-Verlag, 1986.

Levesque, H.; Reiter, R.; Lespérance, Y.; Lin, F.; and Scherl, R. 1997. GOLOG: A logic programming language for dynamic domains. *Journal of Logic Programming* 31(1-3):59–83.

Lifschitz, V.; Pearce, D.; and Valverde, A. 2001. Strongly equivalent logic programs. *ACM Transactions on Computational Logic* 2(4):526–541.

Lifschitz, V. 1999. Answer set planning. In de Schreye, D., ed., *Proceedings of the International Conference on Logic Programming (ICLP'99)*, 23–37. MIT Press.

Orgun, M. A., and Ma, W. 1994. An overview of temporal and modal logic programming. In Gabbay, D. M., and Ohlbach, H. J., eds., *Temporal Logic*, volume 827 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 445–479.

Orgun, M. A., and Wadge, W. W. 1992. Theory and practice of temporal logic programming. In Fariñas del Cerro, L., and Penttonen, M., eds., *Intensional Logics for Programming*. Oxford: Clarendon Press. 21–50.

Pearce, D., and Valverde, A. 2006. Quantified equilibrium logic and the first order logic of here-and-there. Technical Report MA-06-02, University of Málaga.

Pnueli, A. 1977. The temporal logic of programs. In *Proceedings of the Eight-teenth Symposium on Foundations of Computer Science (FOCS'77)*, 46–57. IEEE Computer Society Press.

Pratt, V. 1976. Semantical consideration on floyd-hoare logic. In *Proceedings of the Seventeenth Annual Symposium on Foundations of Computer Science (SFCS'76)*, 109–121. IEEE Computer Society Press.

Son, T.; Baral, C.; Nam, T.; and McIlraith, S. 2006. Domain-dependent knowledge in answer set planning. *ACM Transactions on Computational Logic* 7(4):613–657.

Weinert, A., and Zimmermann, M. 2016. Visibly linear dynamic logic. In *36th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS'16*, 28:1–28:14.