Automated Reasoning

Lecture 13: Linear Temporal Logic I

Jacques Fleuriot
jdf@inf.ed.ac.uk

Overview

- Linear Temporal Logic
  - Some motivation
  - Syntax
  - Semantics
  - Equivalences

Specifications

We are interested in specifying behaviours of systems over time.

- Use **Temporal Logic**

Specifications

We are interested in specifying behaviours of systems over time.

- Use **Temporal Logic**

Specifications are built from:

1. Primitive properties of individual states
   e.g., "is on", "is off", "is active", "is in region", "is at position";
2. propositional connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$;
3. and **temporal** connectives: *e.g.*,
   - At **all times**, the system is not simultaneously *reading* and *writing*.
   - If a *request* signal is asserted **at some time**, a corresponding *grant* signal will be asserted **within 10 time units**.
   - The robot's *position* will **eventually** be at **1 distance unit** from the shelf.

Specifications

We are interested in specifying behaviours of systems over time.

- Use **Temporal Logic**

Specifications are built from:

1. Primitive properties of individual states
   e.g., "is on", "is off", "is active", "is in region", "is at position";
2. propositional connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$;
3. and **temporal** connectives: *e.g.*,
   - At **all times**, the system is not simultaneously *reading* and *writing*.
   - If a *request* signal is asserted **at some time**, a corresponding *grant* signal will be asserted **within 10 time units**.
   - The robot's *position* will **eventually** be at **1 distance unit** from the shelf.

The exact set of temporal connectives differs across temporal logics.
Logics can differ in how they treat time:

- **Linear time** vs. **Branching time**

These differ in reasoning about *non-determinism*.

LTL – Syntax

LTL = Linear(-time) Temporal Logic

Assume some set *Atom* of atomic propositions

Syntax of LTL formulas ϕ:

$$\phi ::= p \mid \neg\phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid \phi \rightarrow \phi \mid \mathbf{X}\phi \mid \mathbf{F}\phi \mid \mathbf{G}\phi \mid \phi\mathbf{U}\phi$$

where $p \in Atom$.

Pronunciation:

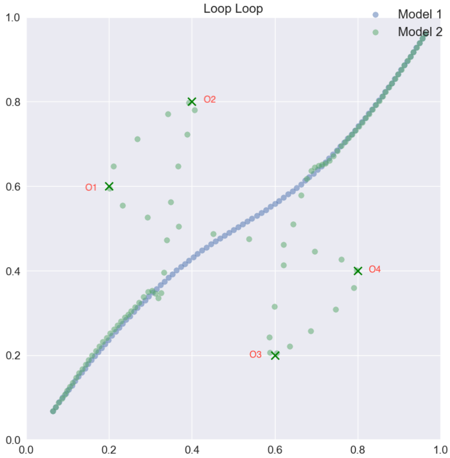
- $\mathbf{X}\phi$ – neXt ϕ
- $\mathbf{F}\phi$ – Future ϕ; Eventually ϕ
- $\mathbf{G}\phi$ – Globally ϕ; Always ϕ
- $\phi\mathbf{U}\psi$ – ϕ Until ψ

Other common connectives: **W** (weak until), **R** (release).

Precedence high-to-low: $(\mathbf{X}, \mathbf{F}, \mathbf{G}, \neg)$, $(\mathbf{U})$, $(\wedge, \vee)$, $\rightarrow$.

- E.g. Write $\mathbf{F}p \wedge \mathbf{G}q \rightarrow p\,\mathbf{U}\,r$ instead of $((\mathbf{F}p) \wedge (\mathbf{G}q)) \rightarrow (p\,\mathbf{U}\,r)$.

Example: Trajectory Specification



$$\mathbf{F}(\|p - o_2\| = 0 \wedge \mathbf{F}(\|p - o_1\| = 0 \wedge \mathbf{F}(\|p - o_4\| = 0 \wedge \mathbf{F}(\|p - o_3\| = 0))))$$

LTL – Informal Semantics

LTL formulas are evaluated at a position i along a path π through the system (a path is a sequence of states connected by transitions)

LTL – Informal Semantics

LTL formulas are evaluated at a position i along a path π through the system (a path is a sequence of states connected by transitions)

- An atomic p holds if p is true the state at position i.
- The propositional connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$ have their usual meanings.
- Semantics is also commonly given in terms of suffixes of paths (words).

LTL – Informal Semantics

Meaning of LTL connectives:

- $\mathbf{X}\phi$ holds if ϕ holds at the next position;
- $\mathbf{F}\phi$ holds if there exists a future position where ϕ holds;
- $\mathbf{G}\phi$ holds if, for all future positions, ϕ holds;
- $\phi\mathbf{U}\psi$ holds if there is a future position where ψ holds, and ϕ holds for all positions prior to that.
- $\phi\mathbf{R}\psi$ holds at a position if ψ holds for ever from that position onwards *or* ϕ holds at some future position, and ψ holds from the current position to up to and including when ϕ holds
  - It is equivalent to $\neg(\neg\phi\mathbf{U}\neg\psi)$.
  - Thus $\mathbf{R}$ is the dual of $\mathbf{U}$.

This will be made more formal in the next few slides.

LTL – Formal Semantics: Transition Systems and Paths

Definition (Transition System)

A *transition system* (or model) $\mathcal{M} = \langle S, \rightarrow, L \rangle$ consists of:

$$\begin{array}{cl} S & \text{a finite set of states} \\ \rightarrow \subseteq S \times S & \text{transition relation} \\ L : S \rightarrow \mathcal{P}(Atom) & \text{a labelling function} \end{array}$$

such that $\forall s_1 \in S.\ \exists s_2 \in S.\ s_1 \rightarrow s_2$

Note: $Atom$ is a fixed set of atomic propositions, $\mathcal{P}(Atom)$ is the powerset of $Atom$.

LTL – Formal Semantics: Transition Systems and Paths

Definition (Transition System)

A *transition system* (or model) $\mathcal{M} = \langle S, \rightarrow, L \rangle$ consists of:

S	a finite set of states
$\rightarrow \subseteq S \times S$	transition relation
$L : S \rightarrow \mathcal{P}(Atom)$	a labelling function

such that $\forall s_1 \in S.\ \exists s_2 \in S.\ s_1 \rightarrow s_2$

Note: *Atom* is a fixed set of atomic propositions, $\mathcal{P}(Atom)$ is the powerset of *Atom*.

Thus, $L(s)$ is the set of atomic propositions that is true in state s.

LTL – Formal Semantics: Transition Systems and Paths

Definition (Transition System)

A *transition system* (or model) $\mathcal{M} = \langle S, \rightarrow, L \rangle$ consists of:

S	a finite set of states
$\rightarrow \subseteq S \times S$	transition relation
$L : S \rightarrow \mathcal{P}(Atom)$	a labelling function

such that $\forall s_1 \in S.\ \exists s_2 \in S.\ s_1 \rightarrow s_2$

Note: $Atom$ is a fixed set of atomic propositions, $\mathcal{P}(Atom)$ is the powerset of $Atom$.

Thus, $L(s)$ is the set of atomic propositions that is true in state s.

Definition (Path)

A *path* π in a transition system $\mathcal{M} = \langle S, \rightarrow, L \rangle$ is an infinite sequence of states $s_0, s_1, \ldots$ such that $\forall i \geq 0.\ s_i \rightarrow s_{i+1}$.

Paths are written as: $\pi = s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \ldots$

LTL – Formal Semantics: Satisfaction by Path

Satisfaction: $\pi \models^i \phi$ – "path at position i satisfies formula ϕ"

$$
\begin{array}{ll}
\pi \models^i \top & \\
\pi \not\models^i \bot & \\
\pi \models^i p & \text{iff } p \in L(s_i) \\
\pi \models^i \neg\phi & \text{iff } \pi \not\models^i \phi \\
\pi \models^i \phi \wedge \psi & \text{iff } \pi \models^i \phi \text{ and } \pi \models^i \psi \\
\pi \models^i \phi \vee \psi & \text{iff } \pi \models^i \phi \text{ or } \pi \models^i \psi \\
\pi \models^i \phi \rightarrow \psi & \text{iff } \pi \models^i \phi \text{ implies } \pi \models^i \psi \\
\pi \models^i \mathbf{X}\,\phi & \text{iff } \pi \models^{i+1} \phi \\
\pi \models^i \mathbf{F}\,\phi & \text{iff } \exists j \geq i.\ \pi \models^j \phi \\
\pi \models^i \mathbf{G}\,\phi & \text{iff } \forall j \geq i.\ \pi \models^j \phi \\
\pi \models^i \phi_1\ \mathbf{U}\ \phi_2 & \text{iff } \exists j \geq i.\ \pi \models^j \phi_2 \text{ and } \forall k \in \{i..j-1\}.\ \pi \models^k \phi_1
\end{array}
$$

Note: $\pi \not\models^i \psi$ means not $\pi \models^i \psi$. Also, the expected equivalences of FOL hold for the formulae on the RHS of the definitions e.g. ϕ implies $\psi \equiv$ (not ϕ) or ψ and not $\exists i.\phi \equiv \forall i.$ not ϕ.

LTL – Formal Semantics: Alternative Satisfaction by Path

Alternatively, we can define $\pi \models \phi$ using the notion of ith suffix $\pi^i = s_i \rightarrow s_{i+1} \rightarrow \ldots$ of a path $\pi = s_0 \rightarrow s_1 \rightarrow \ldots$.

For example, the **alternative definition** of satisfaction for **G** would be:

$$\pi \models \mathbf{G}\,\phi \qquad \text{iff} \qquad \forall j \geq 0.\ \pi^j \models \phi$$

instead of

$$\pi \models^0 \mathbf{G}\,\phi \qquad \text{iff} \qquad \forall j \geq 0.\ \pi \models^j \phi$$

What about $\pi \models \mathbf{X}\,\phi$?

- $\pi \models^i \phi$ is better for understanding, and needed for past-time operators.
- $\pi \models \phi$ is needed for the semantics of branching-time logics, like CTL (Computation Tree Logic).
- Exercise: Work out satisfaction in terms of $\models$ for the other connectives.

Expanding Formulas

We can expand formulas by using the LTL semantics: e.g.

$$\pi \models^0 \mathbf{F}\,\mathbf{G}\, \mathit{at\_table} \equiv \exists i \geq 0. \forall j \geq i.\ \mathit{at\_table} \in L(s_j)$$

LTL – A Few Practical Specification Pattern

1. $\pi \models^i \mathbf{G}\ \mathit{invariant}$

   invariant is true for all future positions

   $\forall j \geq i.\ \pi \models^j \mathit{invariant}$

   $\forall j \geq i.\ \mathit{invariant} \in L(s_j)$

LTL – A Few Practical Specification Pattern

1. $\pi \models^i \mathbf{G}\ \mathit{invariant}$

   invariant is true for all future positions

   $\forall j \geq i.\ \pi \models^j \mathit{invariant}$

   $\forall j \geq i.\ \mathit{invariant} \in L(s_j)$

2. $\pi \models^i \mathbf{G}\ \neg(\mathit{read} \wedge \mathit{write})$

   In all future positions, it is not the case that *read* and *write*

   $\forall j \geq i.\ \mathit{read} \notin L(s_j) \vee \mathit{write} \notin L(s_j)$

LTL – A Few Practical Specification Pattern

1. $\pi \models^i \mathbf{G}\ \mathit{invariant}$

   invariant is true for all future positions

   $\forall j \geq i.\ \pi \models^j \mathit{invariant}$

   $\forall j \geq i.\ \mathit{invariant} \in L(s_j)$

2. $\pi \models^i \mathbf{G}\ \neg(\mathit{read} \wedge \mathit{write})$

   In all future positions, it is not the case that *read* and *write*

   $\forall j \geq i.\ \mathit{read} \notin L(s_j) \vee \mathit{write} \notin L(s_j)$

3. $\pi \models^i \mathbf{G}(\mathit{request} \rightarrow \mathbf{F}\mathit{grant})$

   At every position in the future, a *request* implies that there exists a future point where *grant* holds.

   $\forall j \geq i.\ \mathit{request} \in L(s_j)$ implies $\exists k \geq j.\ \mathit{grant} \in L(s_k)$.

LTL – A Few Practical Specification Pattern

1. $\pi \models^i \mathbf{G}\ invariant$

   invariant is true for all future positions

   $\forall j \geq i.\ \pi \models^j invariant$

   $\forall j \geq i.\ invariant \in L(s_j)$

2. $\pi \models^i \mathbf{G}\ \neg(read \wedge write)$

   In all future positions, it is not the case that *read* and *write*

   $\forall j \geq i.\ read \notin L(s_j) \vee write \notin L(s_j)$

3. $\pi \models^i \mathbf{G}(request \rightarrow \mathbf{F} grant)$

   At every position in the future, a *request* implies that there exists a future point where *grant* holds.

   $\forall j \geq i.\ request \in L(s_j)$ implies $\exists k \geq j.\ grant \in L(s_k)$.

4. $\pi \models^i \mathbf{G}(request \rightarrow (request\, \mathbf{U}\, grant))$

   At every position in the future, a *request* implies that there exists a future point where *grant* holds, and *request* holds up until that point.

   $\forall j \geq i.\ request \in L(s_j)$ implies

   $\exists k \geq j.\ grant \in L(s_k)$ and $\forall l \in \{j, k-1\}.\ request \in L(s_l)$.

Weak Until (W) and Release (R)

- The semantics of $\phi_1 \mathbf{W} \phi_2$ does not require a state to be reached for which ϕ_2 holds, unlike $\phi_1 \mathbf{U} \phi_2$. Thus, it is defined as:

$$\phi_1 \mathbf{W} \phi_2 \stackrel{\text{def}}{=} \phi_1 \mathbf{U} \phi_2 \vee \mathbf{G} \phi_1$$

- The Release operator $\mathbf{R}$ is defined as follows:

$$\phi_1 \mathbf{R} \phi_2 \stackrel{\text{def}}{=} \neg(\neg\phi_2 \mathbf{U} \neg\phi_1)$$

  Its intuitive interpretation is as follows: $\phi_1 \mathbf{R} \phi_2$ holds for a path if ϕ_2 always holds, a requirement that is released as soon as ϕ_1 becomes valid.

- Exercise: Work out the semantics of the Weak Until and Release operators. How do they compare to that of the (Strong) Until operator?

LTL Equivalences 1

$$\phi \equiv \psi \ \stackrel{\text{def}}{=}\ \forall \mathcal{M}. \forall \pi \in \mathcal{M}. \forall i.\, \pi \models^{i} \phi \leftrightarrow \pi \models^{i} \psi$$

LTL Equivalences 1

$$\phi \equiv \psi \stackrel{\text{def}}{=} \forall \mathcal{M}. \forall \pi \in \mathcal{M}. \forall i.\, \pi \models^i \phi \leftrightarrow \pi \models^i \psi$$

Dualities from Propositional Logic:

$$\neg(\phi \wedge \psi) \equiv \neg\phi \vee \neg\psi \qquad\qquad \neg(\phi \vee \psi) \equiv \neg\phi \wedge \neg\psi$$

LTL Equivalences 1

$$\phi \equiv \psi \stackrel{\text{def}}{=} \forall \mathcal{M}. \forall \pi \in \mathcal{M}. \forall i.\, \pi \models^i \phi \leftrightarrow \pi \models^i \psi$$

Dualities from Propositional Logic:

$$\neg(\phi \wedge \psi) \equiv \neg\phi \vee \neg\psi \qquad\qquad \neg(\phi \vee \psi) \equiv \neg\phi \wedge \neg\psi$$

Dualities from LTL:

$$\neg\mathbf{X}\phi \equiv \mathbf{X}\neg\phi \qquad\qquad \neg\mathbf{G}\phi \equiv \mathbf{F}\neg\phi \qquad\qquad \neg\mathbf{F}\phi \equiv \mathbf{G}\neg\phi$$

$$\neg(\phi \;\mathbf{U}\; \psi) \equiv \neg\phi \;\mathbf{R}\; \neg\psi$$

LTL Equivalences 1

$$\phi \equiv \psi \stackrel{\text{def}}{=} \forall \mathcal{M}. \forall \pi \in \mathcal{M}. \forall i.\, \pi \models^i \phi \leftrightarrow \pi \models^i \psi$$

Dualities from Propositional Logic:

$$\neg(\phi \land \psi) \equiv \neg\phi \lor \neg\psi \qquad \neg(\phi \lor \psi) \equiv \neg\phi \land \neg\psi$$

Dualities from LTL:

$$\neg\mathbf{X}\phi \equiv \mathbf{X}\neg\phi \qquad \neg\mathbf{G}\phi \equiv \mathbf{F}\neg\phi \qquad \neg\mathbf{F}\phi \equiv \mathbf{G}\neg\phi$$

$$\neg(\phi \;\mathbf{U}\; \psi) \equiv \neg\phi \;\mathbf{R}\; \neg\psi$$

Distributive laws:

$$\mathbf{G}(\phi \land \psi) \equiv \mathbf{G}\phi \land \mathbf{G}\psi \qquad \mathbf{F}(\phi \lor \psi) \equiv \mathbf{F}\phi \lor \mathbf{F}\psi$$

LTL Equivalences 2

Inter-definitions:

$$\mathbf{F}\phi \equiv \neg\mathbf{G}\neg\phi \qquad \mathbf{G}\phi \equiv \neg\mathbf{F}\neg\phi \qquad \mathbf{F}\phi \equiv \top \ \mathbf{U} \ \phi \qquad \mathbf{G}\phi \equiv \bot \ \mathbf{R} \ \phi$$

LTL Equivalences 2

Inter-definitions:

$$\mathbf{F}\phi \equiv \neg\mathbf{G}\neg\phi \qquad \mathbf{G}\phi \equiv \neg\mathbf{F}\neg\phi \qquad \mathbf{F}\phi \equiv \top\ \mathbf{U}\ \phi \qquad \mathbf{G}\phi \equiv \bot\ \mathbf{R}\ \phi$$

Idempotency:

$$\mathbf{FF}\phi \equiv \mathbf{F}\phi \qquad \mathbf{GG}\phi \equiv \mathbf{G}\phi$$

LTL Equivalences 2

Inter-definitions:

$$\mathbf{F}\phi \equiv \neg\mathbf{G}\neg\phi \qquad \mathbf{G}\phi \equiv \neg\mathbf{F}\neg\phi \qquad \mathbf{F}\phi \equiv \top\ \mathbf{U}\ \phi \qquad \mathbf{G}\phi \equiv \bot\ \mathbf{R}\ \phi$$

Idempotency:

$$\mathbf{FF}\phi \equiv \mathbf{F}\phi \qquad \mathbf{GG}\phi \equiv \mathbf{G}\phi$$

Weak and strong until:

$$\phi\ \mathbf{U}\ \psi \equiv \phi\ \mathbf{W}\ \psi \wedge \mathbf{F}\psi$$

LTL Equivalences 2

Inter-definitions:

$$\mathbf{F}\phi \equiv \neg\mathbf{G}\neg\phi \qquad \mathbf{G}\phi \equiv \neg\mathbf{F}\neg\phi \qquad \mathbf{F}\phi \equiv \top \ \mathbf{U} \ \phi \qquad \mathbf{G}\phi \equiv \bot \ \mathbf{R} \ \phi$$

Idempotency:

$$\mathbf{FF}\phi \equiv \mathbf{F}\phi \qquad \mathbf{GG}\phi \equiv \mathbf{G}\phi$$

Weak and strong until:

$$\phi \ \mathbf{U} \ \psi \equiv \phi \ \mathbf{W} \ \psi \wedge \mathbf{F}\psi$$

Some more surprising equivalences:

$$\mathbf{GFG}\phi \equiv \mathbf{FG}\phi \qquad \mathbf{FGF}\phi \equiv \mathbf{GF}\phi \qquad \mathbf{G}(\mathbf{F}\phi \vee \mathbf{F}\psi) \equiv \mathbf{GF}\phi \vee \mathbf{GF}\psi$$

Summary

- Linear Temporal Logic (H&R 3.2)
  - Syntax
  - Semantics
  - A Few Specification Patterns
  - Equivalences
- Exercise: Prove the equivalences in the previous slides using the semantics of LTL (try this for both the given semantics and the one involving the ith suffix of a path).