Linear temporal logic(LTL)

Syntax

1. alphabet: a finite set of propositional variables (atoms) denoted as AP , logical operators $\{\neg, \wedge\}$ (actually $\{\neg, \wedge, \vee, \rightarrow\}$) and the temporal modal operator {G,F,X,U}, with binding priority $\neg$, X,F,G > U > $\wedge, \vee > \rightarrow$.
2. grammar (including the grammar of propositional logic): structural induction
   if $p \in AP$, then p is a formula;
   if ϕ and ψ are formulas, then $\neg\phi, \phi \wedge \psi, \mathrm{X}\psi, \phi\mathrm{U}\psi, \mathrm{F}\phi, \mathrm{G}\phi$ are formulas.

Semantics

the model of LTL is defined as $\mathscr{M} = (S, \rightarrow, L)$, where S is the domain, representing a set of states; and instead of a function signature σ, there is only a transitive relation $\rightarrow$ on S^2 such that $\forall s \in S, \exists s' \in S, (s, s^{'}) \in \rightarrow$, and a function $L : S \mapsto AP$.
Note: Since $\top$ and $\bot$ are logical constants(null-ary logical connectives but not logical variables) , $\forall s \in S, \bot, \top \notin L(s)$.

Definition. (path and trace/word) A path in a model $\mathscr{M}(S, \rightarrow, L)$ is an infinite sequence of states $s_1, s_2, \ldots$ in S such that, for each $i \geq 1$, $s_i \rightarrow s_{i+1}$. A word/trace is an **infinite** sequence of subsets of AP. A word/trace on the path $\pi : s_1 \rightarrow s_2 \rightarrow \ldots$ is $w = L(s_1), L(s_2), \ldots$.
π^i means the path that starts from s_i (so $\pi^1 = \pi$), and w^i means the suffix $L(s_i), L(s_{i+1}), \ldots$; and p is used to mean an atom.

Definition. (Truth-assignment of LTL)

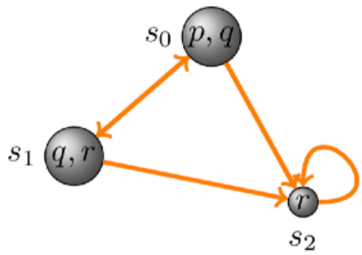
Let $\mathcal{M} = (S, \rightarrow, L)$ be a model and $\pi = s_1 \rightarrow s_2 \rightarrow \ldots$ be a path in $\mathcal{M}$. Let ϕ be an LTL formula. The notion $\pi \models \phi$ (π satisfies ϕ) is defined by structural induction on ϕ as follows:

- $\pi \models p$ iff $p \in L(s_1)$;
- $\pi \models \neg\phi$ iff $\pi \not\models \phi$
- $\pi \models \phi \wedge \psi$ iff $\pi \models \phi$ and $\pi \models \psi$
- $\pi \models X\phi$ iff $\pi^2 \models \phi$
- $\pi \models F\phi$ iff $\pi^i \models \phi$ for some $i \geq 1$
- $\pi \models G\phi$ iff $\pi^i \models \phi$ for all $i \geq 1$
- $\pi \models \phi U \psi$ iff $\pi^i \models \psi$ for some $i \geq 1$ and $\pi^j \models \phi$ for all $j = 1, \ldots, i-1$

$\pi \vDash p \vee q \Leftrightarrow \pi \vDash p$ or $\pi \vDash q$, $\pi \vDash p \rightarrow q \Leftrightarrow \pi \vDash \neg p$ or $\pi \vDash q$.
$\pi \vDash \top$ and $\pi \nvDash \bot$ for all path.

Suppose s is a state and ϕ is a LTL formula. We denote $s \vDash \phi$ iff for any path which starts from s, we have $\pi \vDash \phi$.

e.g.



- $\mathcal{M}, s_0 \models p \wedge q$
- $\mathcal{M}, s_0 \models Xr$
- $\mathcal{M}, s_0 \models G\neg(p \wedge r)$
- $\mathcal{M}, s_0 \models F(\neg q \wedge r) \rightarrow FGr$
- $\mathcal{M}, s_0 \models GFp \rightarrow GFr$

Definition. We say that two LTL formulas ϕ and ψ are equivalent, written $\phi \equiv \psi$, if for all models $\mathcal{M}$ and all paths π in $\mathcal{M}$, $\pi \models \phi$ iff $\pi \models \psi$. *e.g.*,

- $\neg F\phi \equiv G\neg\phi$, $\neg G\phi \equiv F\neg\phi$, $\neg X\phi \equiv X\neg\phi$
- $F(\phi \vee \psi) \equiv F\phi \vee F\psi$, $G(\phi \wedge \psi) \equiv G\phi \wedge G\psi$
- $F\phi \equiv \top U\phi$

recursion law:
$\psi_1 \mathrm{U} \psi_2 \equiv \psi_2 \vee (\psi_1 \wedge \mathrm{X}(\psi_1 \mathrm{U} \psi_2))$

As we can see, {X,U} is an adequate to express {G,F,X,U}; **{¬, ∨ ,X,U} is an adequate set.**
infinite, infinitely means GF ; eventually permanently means FG.

Limits of LTL: can not express AGEFϕ and its negation EFAG$\neg\phi$ as in CTL. But LTL can express single E : $\mathscr{M}, s \vDash \neg X \neg \phi$ is equal to $\mathscr{M}, s \vDash \mathrm{EX}\phi$ in CTL.

Computation tree logic (CTL)

Syntax

1. alphabet: a finite set of propositional variables (atoms) denoted as AP , logical operators $\{\neg, \wedge\}$ (actually $\{\neg, \wedge, \vee, \rightarrow\}$) and the temporal modal operator {G,F,X,U,E,A}.
2. grammar (well-formed formulas): structural induction

if $p \in AP$, then p is a formula;
if ϕ and ψ are formulas, then
$\neg\phi, \phi \vee \psi$, AXψ, EXϕ, AFϕ, EFϕ, EGϕ, A[ϕUψ], E[ϕUψ]AGϕ, EGϕ
are formulas. (Note: the formulas of LTL is not a subset of formulas of CTL)

Binding priorities: $\neg$, AG, EG,AF,EF,AX,EX $>$ $\wedge$, $\vee$ $>$ $\rightarrow$, AU,EU

Semantics:

The definition of model of CTL is the same as LTL.

In the following, $i \geq 1$.

Let $\mathcal{M} = (S, \rightarrow, L)$ be a model, $s \in S$, and ϕ a CTL formula. The notion $\mathcal{M}, s \models \phi$ is defined by structural induction on ϕ as follows:

- $\mathcal{M}, s \models p$ iff $p \in L(s)$;
- $\mathcal{M}, s \models \neg\phi$ iff $\mathcal{M}, s \not\models \phi$
- $\mathcal{M}, s \models \phi \wedge \psi$ iff $\mathcal{M}, s \models \phi$ and $\mathcal{M}, s \models \psi$
- $\mathcal{M}, s \models AX\phi$ iff for all s_1 such that $s \rightarrow s_1$, $\mathcal{M}, s_1 \models \phi$
- $\mathcal{M}, s \models EX\phi$ iff for some s_1 such that $s \rightarrow s_1$, $\mathcal{M}, s_1 \models \phi$
- $\mathcal{M}, s \models AF\phi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow \ldots$, where $s_1 = s$, there is some s_i such that $\mathcal{M}, s_i \models \phi$
- $\mathcal{M}, s \models EF\phi$ iff there is a path $s_1 \rightarrow s_2 \rightarrow \ldots$, where $s_1 = s$, and there is some s_i such that $\mathcal{M}, s_i \models \phi$

- $\mathcal{M}, s \models AG\phi$ iff for all paths $s_1 \to s_2 \to \ldots$, where $s_1 = s$, and all s_i, $\mathcal{M}, s_i \models \phi$
- $\mathcal{M}, s \models EG\phi$ iff there is a path $s_1 \to s_2 \to \ldots$, where $s_1 = s$, and for all s_i, $\mathcal{M}, s_i \models \phi$
- $\mathcal{M}, s \models A[\phi U \psi]$ iff for all paths $s_1 \to s_2 \to \ldots$, where $s_1 = s$, that path satisfies $\phi U \psi$, *i.e.*, there is some s_i such that $\mathcal{M}, s_i \models \psi$, and for all $j < i$, $\mathcal{M}, s_j \models \phi$
- $\mathcal{M}, s \models E[\phi U \psi]$ iff there is a path $s_1 \to s_2 \to \ldots$, where $s_1 = s$, and that path satisfies $\phi U \psi$

Definition. We say that two CTL formulas ϕ and ψ are equivalent, written $\phi \equiv \psi$, if for all models $\mathcal{M}$ and all states s in $\mathcal{M}$, $\mathcal{M}, s \models \phi$ iff $\mathcal{M}, s \models \psi$. *e.g.*,

- $\neg AF\phi \equiv EG\neg\phi$, $\neg EF\phi \equiv AG\neg\phi$, $\neg AX\phi \equiv EX\neg\phi$
- $AF\phi \equiv A[\top U \phi]$, $EF\phi \equiv E[\top U \phi]$

注：CTL 和 LTL equivalent expressions的不同定义导致了有互相不能表示的公式。

$A[\phi U \psi] \equiv \neg(E[\neg\psi U(\neg\psi \wedge \neg\phi)] \vee \neg AF\psi)$

EX 相当于表明存在某个 next state
AX 相当于对于任意的 next state

limit of CTL: we can not focus on an arbitrary path and state its properties in CTL.
e.g. CTL can not express $FG\phi$ and $Fp \to Fq$ as in LTL.

Computation tree logic* (CTL*)

superset of CTL and LTL; so include all the symbols of them.
there are two kinds of formulas in CTL*:

structural induction (p is any atom, ϕ, ψ are state formulas, α, β are path formulas):
state formulas: $p \mid \neg\phi \mid \phi \wedge \psi \mid A[\alpha] \mid E[\alpha]$ (based on CTL)
path formulas: $\phi \mid \neg\alpha \mid \alpha \wedge \beta \mid \mathrm{X}\alpha \mid \mathrm{F}\alpha \mid \mathrm{G}\alpha \mid \alpha \mathrm{U} \beta$ (based on LTL)

semantics see:
https://en.wikipedia.org/wiki/CTL*#Semantics

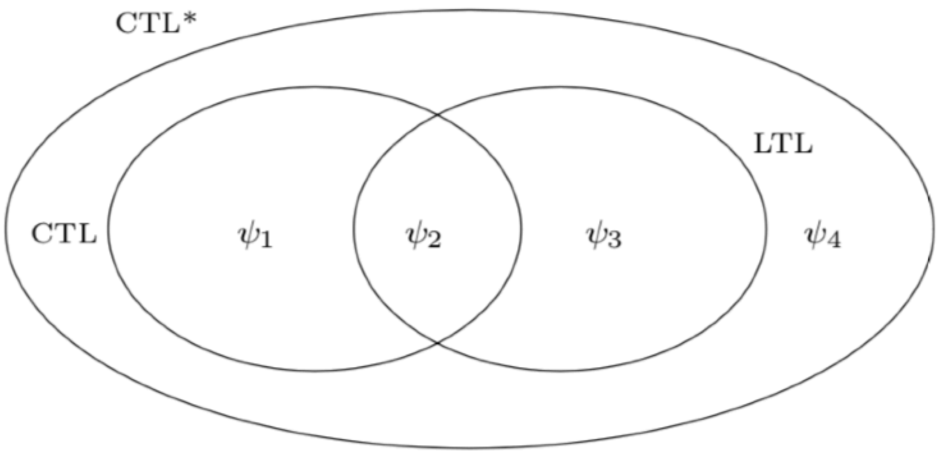


Figure 3.23. The expressive powers of CTL, LTL and CTL*.

- $\psi_1 = AGEFp$, $\psi_2 = G(p \to Fq)$
- $\psi_3 = Fp \to Fq$, $\psi_4 = E[GFp]$
- The proof that ψ_4 is not expressible in CTL is quite complex

即LTL不能表示存在，任意或者任意，存在; CTL不能聚焦于某一条特定的路上
ψ_2 在 CTL 中 相当于 $AG(p \to AFq)$.
$s \vDash Fp \to Fq \Leftrightarrow s \vDash \neg Fp$ or $s \vDash Fq \Leftrightarrow s \vDash G\neg p$ or $s \vDash Fq$ 考虑所有可能的情形，可以发现CTL无法表达。
ψ_4 相当于 $\exists \pi, \forall i, \exists j > i, \pi^j \vDash p$.

translation of properties:

LTL:
after any state satisfying p (after p) : $p \to$ X or $p \to$ XF
for all paths, p precedes q (if q occurs, then p must also occur and precede q):
$G(Fq \to (Fp \wedge G(q \to G\neg p)))$
p doesn't occur after q : $q \to XG\neg p$
p occurs at least twice: $F(p \to XFp)$

CTL:
after any state satisfying p (after p) : $p \to$ AX or $p \to$ AF
for all paths, p precedes q : $\neg E[\neg p U q]$
p doesn't occur after q : $q \to AXAG\neg p$
p occurs at least twice: $EF(p \wedge EXEFp)$

reference:
http://www.cs.toronto.edu/~chechik/courses07/csc2108/Assignments/assign1Sol.pdf

follow/after 意味着此状态之后的状态(要用X)
between 意味着包含两个端点

some comments.
LTL is a kind of temporal logic:
https://en.wikipedia.org/wiki/Temporal_logic
and it's also a kind of modal logic.

CTL is a kind of branching-time logic.