# The Technical Framework of Linear Temporal Logic


Author: E. Allen Emerson

Presented By: Maulik Patel

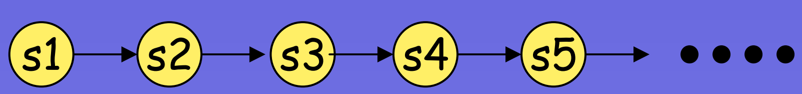
# Timelines

- Properties of time:
  - Is discrete.
  - Has an initial moment with no predecessors.
  - Is infinite into the future.

# Timelines (cont.)

- AP: set of atomic proposition symbols
  - P, Q, $P_1$, $Q_1$ etc.

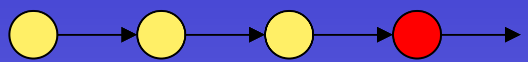
- Linear time structure: M=(S,x,L)
  - S : set of states
  - x : an infinite sequence of states ($N \rightarrow S$)
  - L : labeling of each state



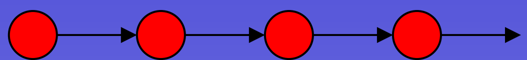
# PLTL

(Propositional Linear Temporal Logic)

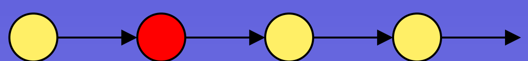
- Basic temporal operators:
  - Fp (◊p): "sometimes p" or "eventually p"



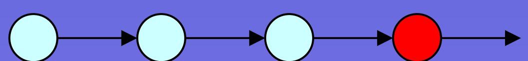
  - Gp (□p) : "always p" or "henceforth p"



  - Xp (οp): "nexttime p"



  - p U q : "p until q"



# PLTL- Syntax

(Propositional Linear Temporal Logic)

- Rules for formula:
  - Each atomic proposition P is a formula
  - If p and q are formulae:
    1) $p \wedge q$ and $\neg p$ are formulae.
    2) $p \, U \, q$ and $Xp$ are formulae.

- Abbreviations:
  - $p \vee q = \neg(\neg p \wedge \neg q)$
  - $p \rightarrow q = \neg p \wedge q$
  - $p \leftrightarrow q = (p \rightarrow q) \wedge (q \rightarrow p)$

# PLTL- Semantics

(Propositional Linear Temporal Logic)

- $M,x \models p$ : "in structure M formula p is true of timeline x"

- $\models$ is defined inductively:
  - $x \models P$ iff P is in $L(s_0)$
  - $x \models (p \wedge q)$ iff $x \models p$ and $x \models q$
    $x \models \neg p$ iff it is not the case that $x \models p$
  - $x \models (p \cup q)$ iff $\exists j(x^j \models q$ and $\forall k<j(x^k \models p))$
    $x \models Xp$ iff $x^1 \models p$

$x^i$=the suffix path $S_i$, $S_{i+1}$, $S_{i+2}$..........

# PLTL- Semantics(cont.)

(Propositional Linear Temporal Logic)

- p U q = “p until q”
- Xp = “nexttime p”
- Fq = “sometimes q”
- Gq = “always q”
- p B q = “p precedes q”
- $F^{\infty}p$ = “infinitely often p” = GFp
- $G^{\infty}p$ = “almost everywhere p” = FGp

# Satisfiable/Valid

- Satisfiable:
  - Exists $M=(S,x,L)$ such that $x \models P$
- Valid:
  - $\models P$ iff for all $M=(S,x,L)$ we have $x \models P$
  - $P$ is valid iff $\neg P$ is not satisfiable

# Variation of PLTL

- Until operator (U):
  - Strong until : (p $U_S$ q) or (p $U_\exists$ q)
  - Weak until: (p $U_W$ q) or (p $U_\forall$ q)
  - p $U_\exists$ q ≡ (p $U_\forall$ q) ∧ Fq
  - p $U_\forall$ q ≡ (p $U_\exists$ q) V Gq

- Does future include present?
  - Reflexive future : $F^{\geq}p \equiv p \vee XF^{>}p$ (similarly $G^{\geq}p$)
  - Strict future : $F^{>}p \equiv XF^{\geq}p$ (similarly $G^{>}p$)
  - Strict until : $(p\ U^{>}\ q) \equiv X(p\ U\ q)$

# Variation of PLTL(cont.)

- What if we have finite timeline (I)?
  - Gp = for all subsequent times in I, p holds.
  - Fp = for some subsequent times in I, p holds.
  - p U q = for some subsequent time in I, q holds and p holds at all subsequent times until them.
  - $X_{\forall}p$ = weak nexttime
  - $X_{\exists}p$ = strong nexttime

# Variation of PLTL(cont.)

- Adding past tense:
  - Fp : $F^{+}p$ (future) ; $F^{-}p$ (past)
  - Gp : $G^{+}p$ (future) ; $G^{-}p$ (past)
  - Xp : $X^{+}p$ (future) ; $X^{-}p$ (past)
  - p U q : p $U^{+}$ q (future) ; p $U^{-}$ q (past)

- PLTLF = future
- PLTLP = past
- PLTLB = both

# Variation of PLTL(cont.)

- Past tense:
  - $M,(x,i) \models p$: "in structure M along timeline x at time i formula p holds true"

- Future tense:
  - $x \models p \equiv (x,0) \models p$

# Example

**Temporal Connectives:**
1. Next (O)
2. Always (□)
3. Eventually (◊)
4. Until (U)

- Pass by room 5
  - ◊ at(room5)
- Go to room 5 and stay there
  - ◊□ at(room5)
- Go to room 5 and stay there, but don't ever get hit
  - ◊□ at(room5) ∧ □ (¬ hit)
- Go to room 5 and stay there, but don't get hit until then
  - (¬ hit) U ◊ at(room5)

Example copied form : "Introduction to Linear Temporal Logic(LTL) in Goal Specification", Jicheng Zhao

# Example(cont.)

- Go to Room 5 and stay there, and any time if the door is closed and you open it then you must eventually close it.
  - $\Diamond\Box\, at(room5) \wedge \Box\, ((closed \wedge \bigcirc\neg\, closed) \rightarrow \bigcirc\Diamond\, closed)$

# Question 1

- What is the need for the past tense in PLTL? Is it really useful?

# Question 2

- What properties of a program can be checked using linear time temporal logic?
  - Reachability: A particular state is reachable from present state
  - Safety: A bad property will never be satisfy.

Thank You