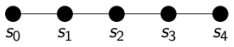
Lógica LTL

Aula 11

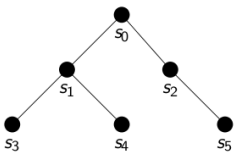
1 Noção de Tempo

Tempo

Linear: O tempo é um conjunto de caminhos, onde um caminho é uma sequência de estados.



Ramificado: O tempo é representado em árvore com raíz no momento presente.

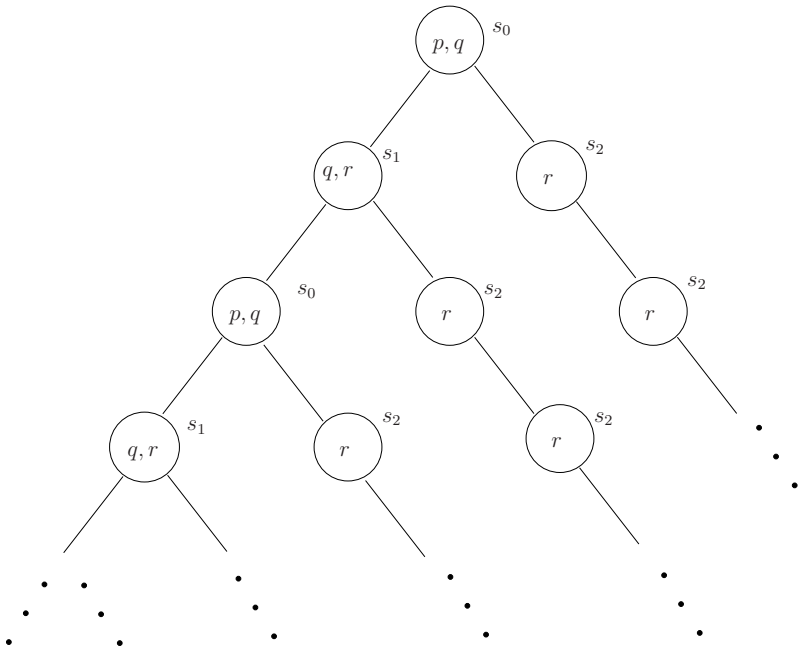


Árvore Infinita de Computação

Caminho (de Computação)

$T = (S, Act, \longrightarrow, AP, I, L)$ um sistema de transição, um caminho é uma sequência infinita de estados $s_1, s_2, s_3, \ldots$ em S, tal que para todo $i \geq 1$, $s_i \in Pre(s_{i+1})$.

O conjunto de caminhos a partir dum estado s pode ser visto como uma árvore infinita de computação.



Lógicas Temporais

- São extensões da lógica proposicional e da lógica de primeira ordem
- Usam modalidades para referir o comportamento das execuções dos sistemas reactivos
- a lógica temporal linear (LTL) permite a quantificação ao longo de um caminho.
- A lógica CTL (*Computation Tree Logic*) permite quantificação (existencial) sobre os caminhos dessa árvore.

2 Lógica Temporal Linear, LTL

Lógica Temporal Linear, LTL

AP, conjunto de variáveis proposicionais $p, q, r, s, \ldots$

$$\begin{aligned}\varphi \quad ::= \quad & \mathsf{false} \mid \mathsf{true} \mid p \mid (\neg\varphi) \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid (\varphi \rightarrow \varphi) \mid \\ & (\mathrm{X}\varphi) \mid (\mathrm{F}\varphi) \mid (\mathrm{G}\varphi) \mid (\varphi \mathrm{U} \varphi) \mid (\varphi \mathrm{W} \varphi) \mid (\varphi \mathrm{R} \varphi)\end{aligned}$$

$\mathbf{X}\varphi$ φ verifica-se no estado seguinte (*neXt*) (ou $\bigcirc\varphi$)

$\mathbf{F}\varphi$ φ verifica-se nalgum estado futuro (ou $\Diamond\varphi$, *eventually*)

$\mathbf{G}\varphi$ φ verifica-se em todos os estados (ou $\Box\varphi$, sempre)

$\varphi\mathbf{U}\psi$ φ verifica-se até ψ se verificar (*Until*)

$\varphi\mathbf{W}\psi$ φ verifica-se até ψ se verificar ou sempre (*Weak until*)

$\varphi\mathbf{R}\psi$ ψ verifica-se até φ se verificar (inclusivé!) ou sempre (*Release*)

Lógica Temporal Linear, LTL

$$(((\mathrm{F}p) \wedge (\mathrm{G}p)) \rightarrow (p\mathrm{W}r))$$
$$((\mathrm{G}(\mathrm{F}p)) \rightarrow (\mathrm{F}(q \vee p)))$$
$$(p\mathrm{W}(q\mathrm{W}r))$$

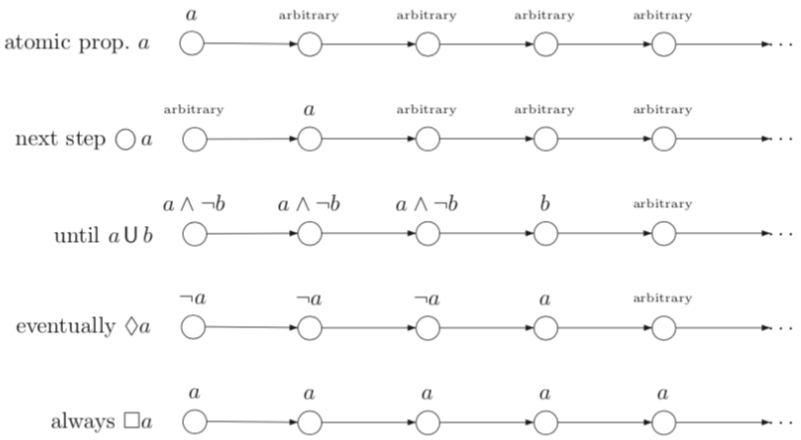
Convensão de prioridades (omissão de parêntesis)

- As conectivas unárias ($\neg$, X, F,G) têm prioridade mais alta
- Depois as conectivas U, W e R.
- Depois as conectivas $\wedge$ e $\vee$.
- Depois a conectiva $\rightarrow$.

$$\mathrm{F}p \wedge \mathrm{G}p \rightarrow p\mathrm{W}r$$
$$\mathrm{GF}p \rightarrow \mathrm{F}(q \vee p)$$
$$p\mathrm{W}(q\mathrm{W}r)$$

Semântica do LTL

$\bigcirc$ é X; $\Diamond$ é F e $\Box$ é G



GF e FG

- GFφ significa *um número infinito de vezes*
- FGφ significa *a partir de um certo momento, para sempre*

Exemplos

- Exclusão mútua (*segurança*): $\{A_0A_1A_2\ldots \mid \{c_1, c_2\} \not\subseteq A_i,\ i \geq 0, A_i \subseteq AP\}$

$$\mathrm{G}(\neg c_1 \vee \neg c_2)$$

- P_i acede à seccão critica um n.i.d.v (*vivacidade/liveness*):

$$\{A_0A_1A_2\ldots \mid (\overset{\infty}{\exists}\ j.\ c_1 \in A_j) \wedge (\overset{\infty}{\exists}\ j.\ c_2 \in A_j)\}$$

$$\mathrm{GF}c_1 \wedge \mathrm{GF}c_2$$

- *starvation freedom* (vivacidade):

$$\{A_0A_1A_2\ldots \mid \forall i \in \{1,2\}, (\overset{\infty}{\exists}\ j.\ w_i \in A_j) \rightarrow (\overset{\infty}{\exists}\ j.\ c_i \in A_j)\}$$

$$(\mathrm{GF}w_1 \rightarrow \mathrm{GF}c_1) \wedge (\mathrm{GF}w_2 \rightarrow \mathrm{GF}c_2)$$

Semântica do LTL

Seja $T = (S, Act, \longrightarrow, I, AP, L)$ e um traço

$$\sigma = A_0 A_1 A_2 \ldots,$$

A propriedade LT induzida por φ é

$$Words(\varphi) = \{\ \sigma \in (2^{AP})^\omega \mid \sigma \models \varphi\ \}$$

onde a relação de satisfabilidade $\sigma \models \varphi$ define-se indutivamente por:

Satisfazibilidade

1. $\sigma \models \mathsf{true}$
2. $\sigma \not\models \mathsf{false}$
3. $\sigma \models p$ sse $p \in A_0$ (ou se $\pi = s_0 \ldots$ um caminho, $p \in L(S_0)$)
4. $\sigma \models \neg\varphi$ sse $\sigma \not\models \varphi$
5. $\sigma \models \varphi \wedge \psi$ sse $\sigma \models \varphi$ e $\sigma \models \psi$
6. $\sigma \models \varphi \vee \psi$ sse $\sigma \models \varphi$ ou $\sigma \models \psi$
7. $\sigma \models \varphi \rightarrow \psi$ sse sempre que $\sigma \models \varphi$ então $\sigma \models \psi$
8. $\sigma \models \mathrm{X}\varphi$ sse $\sigma[1 \ldots] \models \varphi$
9. $\sigma \models \mathrm{G}\varphi$ sse $\forall i \geq 0, \sigma[i \ldots] \models \varphi$
10. $\sigma \models \mathrm{F}\varphi$ sse $\exists i \geq 0, \sigma[i \ldots] \models \varphi$
11. $\sigma \models \varphi \mathrm{U} \psi$ sse $\exists i \geq 0, \sigma[i \ldots] \models \psi$ e $\forall\, 0 \leq j < i,\ \sigma[j \ldots] \models \varphi$
12. $\sigma \models \varphi \mathrm{W} \psi$ sse ou $\exists i \geq 0, \sigma[i \ldots] \models \psi$ e $\forall\, 0 \leq j < i, \sigma[j \ldots] \models \varphi$ ou $\forall k \geq 0,\ \sigma[k \ldots] \models \varphi$
13. $\sigma \models \varphi \mathrm{R} \psi$ sse ou $\exists i \geq 0, \sigma[i \ldots] \models \varphi$ e $\forall\, 0 \leq j \leq i, \sigma[j \ldots] \models \psi$ ou $\forall k \geq 0,\ \sigma[k \ldots] \models \psi$

E também...

- $\sigma \models \mathrm{GF}\varphi$ sse $\overset{\infty}{\exists}\, j.\sigma[j \ldots] \models \varphi$
- $\sigma \models \mathrm{FG}\varphi$ sse $\overset{\infty}{\forall}\, j.\sigma[j \ldots] \models \varphi$

onde

$$\overset{\infty}{\exists}\, j \equiv \forall k \geq 0 \exists j \geq k$$

e

$$\overset{\infty}{\forall}\, j \equiv \exists k_0 \geq 0 \forall j \geq k_0$$

Traços (revisão)

- $T = (S, Act, \longrightarrow, I, AP, L)$ sistema de transições

$$\begin{aligned} \rho &= s_0 \xrightarrow{\alpha_1} s_1 \xrightarrow{\alpha_2} s_2 \cdots \\ \pi &= s_0 s_1 s_2 \ldots \\ trace(\pi) &= L(s_0)L(s_1)L(s_2)\ldots \end{aligned}$$

- Π conjunto de caminhos, $trace(\Pi) = \{trace(\pi) \mid \pi \in \Pi\}$.
- $trace(s) = \{trace(\pi) \mid s = first(\pi)\}$
- $Traces(s) = trace(Paths(s))$
- $Traces(T) = \bigcup_{s \in I} Traces(s)$

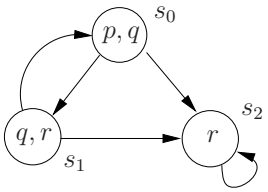
Semântica do LTL para caminhos, estados e sistemas

Seja $T = (S, Act, \longrightarrow, I, AP, L)$, π um fragmento de caminho infinito e $s \in S$.

- $\pi \models \varphi$ sse $trace(\pi) \models \varphi$
- $s \models \varphi$ sse *para todos* os caminhos de computação π começando em s, se tem $\pi \models \varphi$, i.e.
$$\forall \pi \in Paths(s), \pi \models \varphi.$$
- T satisfaz φ sse $Traces(T) \subseteq Words(\varphi)$, ou seja, sse
$$s_0 \models \varphi, \forall s_0 \in I.$$

Semântica do LTL

Exercício 11.1. *Considera o sistema* $T = (S = \{s_0, s_1, s_2\}, \{s_0 \longrightarrow s_1, s_0 \longrightarrow s_2, s_1 \longrightarrow s_2, s_1 \longrightarrow s_0, s_2 \longrightarrow s_2\}, L(s_0) = \{p, q\}, L(s_1) = \{q, r\}, L(s_2) = \{r\})$.

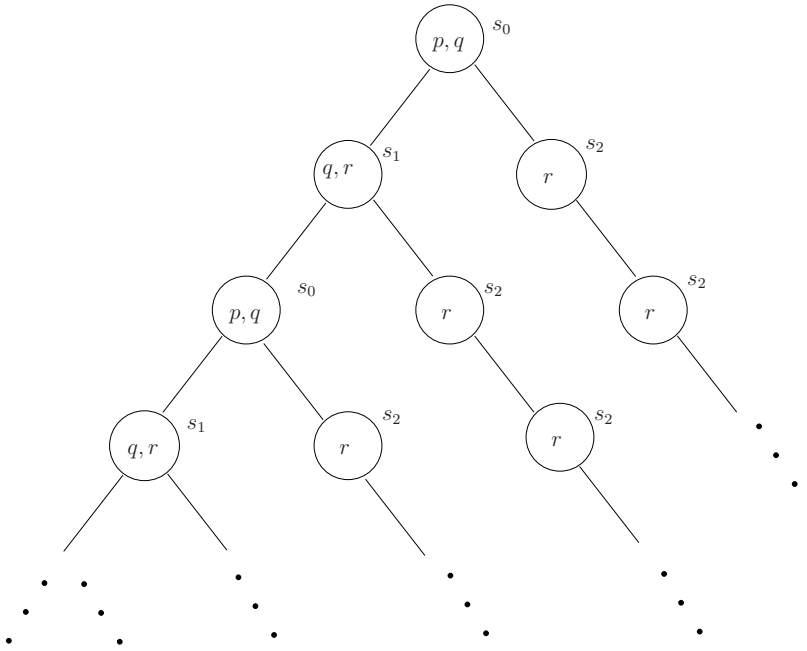


Determina quais destas relações são verdadeiras:

1. $s_0 \models p \wedge q$

2. $s_0 \models Xr$
3. $s_0 \models X(q \wedge r)$
4. $s_0 \models G\neg(p \wedge r)$
5. $s_0 \models GFp$
6. $s_0 \models GFp \rightarrow GFr$

$\diamond$



Algumas especificações práticas (padrões)

Supõe que ass variáveis proposicionais correspondem a estados de uma sistema reactivo real.

- Não é possível chegar a um estado em que `started` se verifica e `ready` não se verifica: G$\neg$(`started` $\wedge \neg$`ready`)
- Para qualquer estado, se `request` se verifica, então no futuro também se irá verificar `ack`: G(`request` $\rightarrow$ F`ack`)
- Um processo é activado (`enabled`) um número infinito de vezes em todos os caminhos de computação: GF`enabled`
- Um processo irá ficar permanentemente em `deadlock`: FG`deadlock`

- Se um processo é activado (`enabled`) um número infinito de vezes, então ele executa (`run`) um número infinito de vezes: GF`enabled` $\rightarrow$ GF`run`
- Para todos os estados, existe uma caminho para um estado que satisfaz `restart`: Não se pode exprimir em LTL!

Em LTL não se pode exprimir a existência de um caminho de um estado com uma dada propriedade.

Exercício 11.2. *Considera o sistema* $\mathcal{M} = (S = \{q_1, q_2, q_3, q_4\}, \{q_1 \rightarrow q_2, q_2 \rightarrow q_2, q_3 \rightarrow q_1, q_3 \rightarrow q_2, q_3 \rightarrow q_4, q_4 \rightarrow q_3\}, L(q_1) = \{\}, L(q_2) = \{b\}, L(q_3) = \{a\}, L(q_4) = \{a, b\})$.

Para cada uma das fórmulas seguintes:

a) Ga;

b) aUb;

c) $aUX(a \wedge \neg b)$;

d) $X\neg b \wedge G(\neg a \vee \neg b)$;

e) $X(a \wedge b) \wedge F(\neg a \wedge \neg b)$;

- *encontra um caminho a partir do estado* q_3 *que satisfaz* φ;
- *determina se* $q_3 \models \varphi$.

$\diamond$

3 Especificações

Equivalência semântica

Equivalência de fórmulas

Duas fórmulas em LTL são semânticamente equivalentes, $\varphi_1 \equiv \varphi_2$, sse $Words(\varphi_1) = Words(\varphi_2)$

Exercício

Mostra que

$$\begin{array}{rcl} \mathrm{F}(\varphi \vee \psi) & \equiv & \mathrm{F}\varphi \vee \mathrm{F}\psi \\ \mathrm{F}(\varphi \wedge \psi) & \not\equiv & \mathrm{F}\varphi \wedge \mathrm{F}\psi \\ \mathrm{G}(\varphi \wedge \psi) & \equiv & \mathrm{G}\varphi \wedge \mathrm{G}\psi \\ \mathrm{G}(\varphi \vee \psi) & \not\equiv & \mathrm{G}\varphi \vee \mathrm{G}\psi \end{array}$$

Equivalência semântica

Teorema 11.1. *Temos as seguintes equivalências:*

$$\begin{aligned}
\neg(\varphi \wedge \psi) &\equiv \neg\varphi \vee \neg\psi \\
\neg(\varphi \vee \psi) &\equiv \neg\varphi \wedge \neg\psi \\
\neg G\varphi &\equiv F\neg\varphi \\
\neg F\varphi &\equiv G\neg\varphi \\
\neg X\varphi &\equiv X\neg\varphi \\
\neg(\varphi\, U\psi) &\equiv \neg\varphi R\neg\psi \\
\neg(\varphi R\psi) &\equiv \neg\varphi\, U\neg\psi \\
X(\varphi\, U\psi) &\equiv (X\varphi)\, U(X\psi) \\
F(\varphi \vee \psi) &\equiv F\varphi \vee F\psi \\
G(\varphi \wedge \psi) &\equiv G\varphi \wedge G\psi
\end{aligned}$$

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

Se $\sigma \not\models \varphi \mathrm{U} \psi$ então

- ou $\forall\, i$, $\sigma[i \ldots] \models \neg\psi$
- ou $\exists i \geq 0, \sigma[i \ldots] \models \neg\varphi$ e $\forall\, 0 \leq j \leq i$, $\sigma[j \ldots] \models \neg\psi$
- isto é $\sigma \models \neg\varphi \mathrm{R} \neg\psi$

Mais equivalências

$$\begin{aligned}
\mathrm{F}\varphi &\equiv \mathsf{true}\mathrm{U}\varphi \\
\mathrm{G}\varphi &\equiv \mathsf{false}\mathrm{R}\varphi \\
\varphi \mathrm{U} \psi &\equiv \varphi \mathrm{W} \psi \wedge \mathrm{F}\psi \\
\varphi \mathrm{W} \psi &\equiv \varphi \mathrm{U} \psi \vee \mathrm{G}\varphi \\
\varphi \mathrm{W} \psi &\equiv \psi \mathrm{R}(\varphi \vee \psi) \\
\varphi \mathrm{R} \psi &\equiv \psi \mathrm{W}(\varphi \wedge \psi) \\
\varphi \mathrm{U} \psi &\equiv \psi \vee (\varphi \wedge \mathrm{X}(\varphi \mathrm{U} \psi)) \\
\mathrm{F}\varphi &\equiv \varphi \vee \mathrm{XF}\varphi \\
\mathrm{G}\varphi &\equiv \varphi \wedge \mathrm{XG}\varphi
\end{aligned}$$

Exercício 11.3. *Mostra as equivalências anteriores.* $\diamond$

Conjuntos completos de conectivas

Um conjunto de conectivas é completo se as restantes se podem obter em termos dos seus elementos

Teorema 11.2. *Considerando as conectivas temporais, os seguintes conjuntos são completos:* $\{U, X\}$, $\{R, X\}$ *e* $\{W, X\}$.

Exercício 11.4. *Mostre que* $\{U, X\}$ *é completo para LTL.* $\diamond$

Exercícios

Exercício 11.5. *Demonstra a veracidade ou falsidade das afirmações seguintes com uma prova ou um contra-exemplo.*

a) $G(\varphi \vee \psi) \equiv G(\varphi) \vee G(\psi)$

b) $G(\varphi \wedge \psi) \equiv G(\varphi) \wedge G(\psi)$

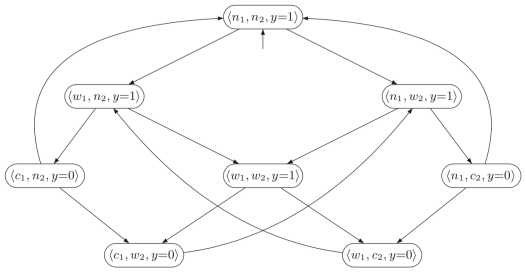
$\diamond$

Exercício 11.6. *Indica uma fórmula em LTL que é semânticamente equivalente a*

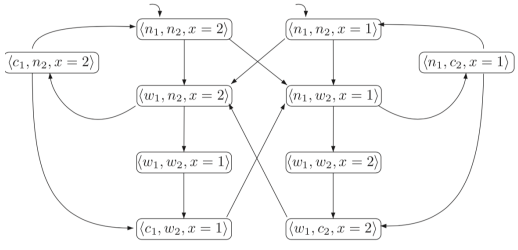
$$\neg G(c_1 \rightarrow c_1 W(\neg c_1 \wedge \neg c_1 W c_2))$$

mas não utiliza a conectiva W. $\diamond$

T_{sem}



T_{pet}



Especificação das propriedades em LTL

Considera os sistemas para a exclusão mútua T_{sem} (com semáforo) e T_{pet} (de Peterson). E as propriedades

de Segurança Já vimos que $T_{sem} \models \mathrm{G}\neg(c_1 \wedge c_2)$

de Vivacidade $T_{sem} \not\models \mathrm{G}(w_1 \rightarrow \mathrm{F}c_1)$, mas $T_{pet} \models \mathrm{G}(w_1 \rightarrow \mathrm{F}c_1)$

Non-blocking Cada estado em que n_i se verifica existe um sucessor tal que w_i se verifica. *Não se pode exprimir em LTL.*

Not strict sequencing Existe um caminho com dois estados que satisfazem c_1, tal que nenhum estado entre eles verifica c_2. *Isto não se pode exprimir directamente em LTL.* Mas, usando o complementar: Todos os caminhos com um período em que c_1 é satisfeito,mas que termina, não podem ter c_1 antes de ter c_2:

$$G(c_1 \rightarrow c_1 \mathrm{W}(\neg c_1 \wedge \neg c_1 W c_2))$$

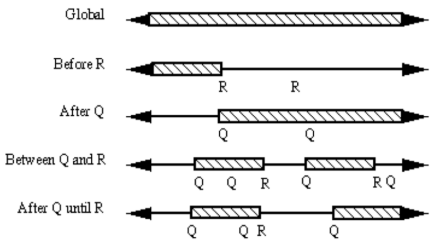
Complementando temos a propriedade pretendida, e que se verifica em T_{sem}. Porquê?

Especificações de Propriedades

- Ocorrência: existência ou não de certos estados numa dada região
  - Ausência
  - Universalidade
  - Existência
  - Existência limitada
- Ordem: relaciona pares de estados numa dada região
  - Precedência
  - Sequência (*)

Regiões

- Global
- Antes de r
- Depois de q
- Entre q e r
- Depois de q até r



Especificações em LTL

Ausência/Universalidade

φ:$\neg p$ / p

Global	$\mathrm{G}\varphi$
Antes de r	$\mathrm{F}r \rightarrow \varphi \mathrm{U} r$
Depois de q	$\mathrm{G}(q \rightarrow \mathrm{G}\varphi)$
Entre q e r	$\mathrm{G}((q \wedge \neg r \wedge \mathrm{F}r) \rightarrow (\varphi \mathrm{U} r))$
Depois de q até r	$\mathrm{G}(q \wedge \neg r \rightarrow (\varphi \mathrm{W} r))$

Existência

p passa a ser verdade

Global	$\mathrm{F}p$
Antes de r	$\neg r \mathrm{W}(p \wedge \neg r)$
Depois de q	$\mathrm{G}(\neg q) \vee \mathrm{F}(q \wedge \mathrm{F}p)$
Entre q e r	$\mathrm{G}(q \wedge \neg r \rightarrow (\neg r \mathrm{W}(p \neg r)))$
Depois de q até r	$\mathrm{G}(q \wedge \neg r \rightarrow (\neg r \mathrm{U}(p \wedge \neg r))$

Especificações em LTL

Existência limitada

Ex:*transições para estados com p ocorrem no máximo 2 vezes*

Global	$\neg p \mathrm{W}(p \mathrm{W}(\neg p \mathrm{W}(p \mathrm{W} \mathrm{G} \neg p)))$
Antes de r	$\mathrm{F}r \rightarrow ((\neg p \wedge \neg r)\mathrm{U}(r \vee ((p \wedge \neg r)\mathrm{U}$ $(r \vee (\neg p \wedge \neg r)\mathrm{U}(r \vee ((p \wedge \neg r)$ $\mathrm{U}(r \vee (\neg p \mathrm{U} r))))))))$
Depois de q	$\mathrm{F}q \rightarrow (\neg q \mathrm{U}(q \wedge \neg p \mathrm{W}(p \mathrm{W}(\neg p \mathrm{W}(p \mathrm{W} \mathrm{G} \neg p)))$
Entre q e r	$\mathrm{G}((q \wedge \neg r) \rightarrow ((\neg p \wedge \neg r)\mathrm{U}(r \vee ((p \wedge \neg r)\mathrm{U}$ $(r \vee (\neg p \wedge \neg r)\mathrm{U}(r \vee ((p \wedge \neg r)$ $\mathrm{U}(r \vee (\neg p \mathrm{U} r)))))))))$
Depois de q até r	$\mathrm{G}(q \rightarrow ((\neg p \wedge \neg r)\mathrm{U}(r \vee ((p \wedge \neg r)\mathrm{U}$ $(r \vee (\neg p \wedge \neg r)\mathrm{U}(r \vee ((p \wedge \neg r)$ $\mathrm{U}(r \vee (\neg p \mathrm{W} r) \vee \mathrm{G}p))))))))$

Especificações em LTL

Precedência

s precede p

Global	$\neg p \mathrm{W} s$
Antes de r	$\mathrm{F} r \rightarrow (\neg p \mathrm{U} (s \vee r))$
Depois de q	$\mathrm{G}(\neg q) \vee \mathrm{F}(q \wedge (\neg p \mathrm{W} s))$
Entre q e r	$\mathrm{G}((q \wedge \neg r \wedge \mathrm{F} r) \rightarrow (\neg p \mathrm{U}(s \vee r)))$
Depois de q até r	$\mathrm{G}(q \wedge \neg r \rightarrow (\neg p \mathrm{W}(s \vee r)))$

Justeza baseada em ações

- Dado $T = (S, Act, \longrightarrow, AP, I, L)$
- $Act(s) = \{ \alpha \in Act \mid \exists s' \in S.\, s \xrightarrow{\alpha} s' \}$, ações executáveis em s
- Seja $A \subseteq Act$
- $\rho = s_0 \xrightarrow{\alpha_0} s_1 \xrightarrow{\alpha_1} \ldots,$
- ρ é *incondicionalmente* A-justo se

$$\overset{\infty}{\exists}\, j.\, \alpha_j \in A.$$

- ρ é *fortemente* A-justo se

$$(\overset{\infty}{\exists}\, j.\, Act(s_j) \cap A \neq \emptyset) \rightarrow \overset{\infty}{\exists}\, j.\, \alpha_j \in A.$$

- ρ é *fracamente* A-justo se

$$(\overset{\infty}{\forall}\, j.\, Act(s_j) \cap A \neq \emptyset) \rightarrow \overset{\infty}{\exists}\, j.\, \alpha_j \in A.$$

- onde $\overset{\infty}{\forall}\, j$ significa todos j excepto um número finito.

Restrições de Fairness/Justeza em LTL

Sejam Φ e Φ duas fórmulas proposicionais sobre AP. Uma restrição de justeza

- *incondicional* é da forma $\mathrm{GF}\Phi$ (ufair)
- *forte* é da forma $\mathrm{GF}\Phi \rightarrow \mathrm{GF}\Psi$ (sfair)
- *fraca* é da forma $\mathrm{FG}\Phi \rightarrow \mathrm{GF}\Psi$ (wfair)

uma assumção de justeza é uma conjunção de restrições de justeza

$$fair = ufair \wedge sfair \wedge wfair$$

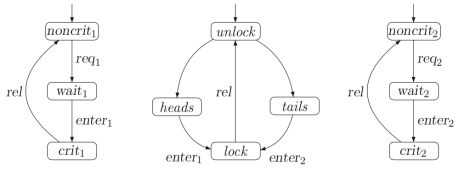
Satisfazibilidade para LTL com Fairness

$$\begin{aligned} FairPaths(s) &= \{\pi \in Paths(s) \mid \pi \models fair\} \\ FairTraces(s) &= \{trace(\pi) \mid \in FairPaths(s)\} \end{aligned}$$

- $s \models_{fair} \varphi$ sse $\forall \pi \in FairPaths(s), \pi \models \varphi$
- $T \models_{fair} \varphi$ sse $\forall s_0 \in I.s_0 \models_{fair} \varphi$

Exclusão mútua com árbitro aleatório

O árbitro decide quem entra na região crítica lançando uma moeda ao ar, cujo resultado é simulado pelas ações *heads* (cara) e *tails* (coroa).



- Tem-se que $T_1||Arbiter||T_2 \not\models \mathrm{GF}c_1$. Porquê?
- Agora se $fair = \mathrm{GF}heads \wedge \mathrm{GF}tails$
- então teríamos $T_1||Arbiter||T_2 \models_{fair} \mathrm{GF}c_1$.
- em geral $T \models_{fair} \varphi$ sse $T \models (fair \rightarrow \varphi)$ (Prova!)