Exclusion mutuelle (suite)

- Exclusion mutuelle: Jamais les deux processus ne peuvent se trouver en SC au même moment.

$$\neg\, \mathbf{F}\,(SC_1 \wedge SC_2) \qquad \mathbf{G}\,(\neg\, SC_1 \vee \neg\, SC_2)$$

- Il n'y a jamais de blocage.

$$\mathbf{G}\,(\mathbf{X}\,\top\,)$$ (NB: toujours vrai si $\rightarrow$ est totale)

- Absence de famine: Si un processus demande l'accès à la SC, il y arrivera un jour.

$$\mathbf{G}\,(D_1 \Rightarrow \mathbf{F}\ SC_1) \ \wedge\ \mathbf{G}\,(D_2 \Rightarrow \mathbf{F}\ SC_2)$$

- Attente bornée: Si un processus demande l'accès à la SC, l'autre processus ne peut pas passer avant lui plus d'une fois.

il nous manque encore un opérateur… patience !

LTL

Syntaxe: $P \in AP$

$$\phi,\psi ::= P \mid \neg\phi \mid \phi\vee\psi \mid \phi\wedge\psi \mid \mathbf{X}\,\phi \mid \psi\mathbf{U}\phi \mid \mathbf{X}^{-1}\,\phi \mid \psi\mathbf{S}\phi$$

U = until **S** = since

$\rho,i \models \mathbf{X}\,\phi$ ssi $\rho,i+1 \models \phi$

$\rho,i \models \psi\mathbf{U}\phi$ ssi ($\exists\, j \geq i.$ ($\rho,j \models \phi$ et $\forall\, i\leq k<j$ on a $\rho,k \models \psi$)

$\rho,i \models \mathbf{X}^{-1}\,\phi$ ssi ($i>0$ et $\rho,i-1 \models \phi$)

$\rho,i \models \psi\mathbf{S}\phi$ ssi ($\exists\, j \leq i.$ ($\rho,j \models \phi$ et $\forall\, j<k\leq i$ on a $\rho,k \models \psi$)

Exemples de formules

$\mathbf{F}\ \phi = ?$

$\mathbf{F}\ \phi = \top\ \mathbf{U}\ \phi$

def:

$\rho,i \models \mathbf{F}\ \phi$ ssi ($\exists\ j \geq i.\ \ \rho,j \models \phi$)

$\rho,i \models\ \psi \mathbf{U} \phi$ ssi ($\exists\ j \geq i.$ ($\rho,j \models \phi$ et $\forall\ i \leq k < j$ on a $\rho,k \models \psi$)

$\rho,i \models\ \top \mathbf{U} \phi$ ssi ($\exists\ j \geq i.$ ($\rho,j \models \phi$ et $\forall\ i \leq k < j$ on a $\rho,k \models \top$)

$\Longleftrightarrow$

($\exists\ j \geq i.\ \ \rho,j \models \phi$)

$\Longleftrightarrow$

$\rho,i \models\ \mathbf{F} \phi$

Exemples de formules

$\mathbf{F^{-1}}\ \phi = \top\ \mathbf{S}\ \phi$

def:

$\rho,i \models \mathbf{F^{-1}}\ \phi$ ssi ($\exists\ j \leq i.\ \ \rho,j \models \phi$)

$\rho,i \models\ \psi \mathbf{S} \phi$ ssi ($\exists\ j \leq i.$ ($\rho,j \models \phi$ et $\forall\ j < k \leq j$ on a $\rho,k \models \psi$)

$\rho,i \models\ \top \mathbf{S} \phi$ ssi ($\exists\ j \leq i.$ ($\rho,j \models \phi$ et $\forall\ j < k \leq i$ on a $\rho,k \models \top$)

$\Longleftrightarrow$

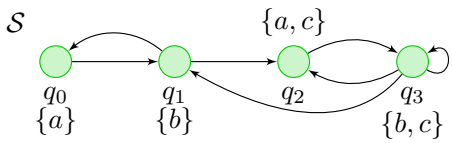
($\exists\ j \leq i.\ \ \rho,j \models \phi$)

$\Longleftrightarrow$

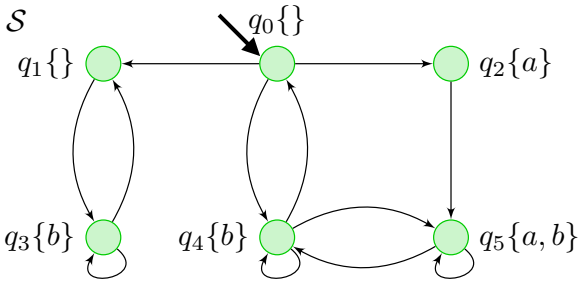
$\rho,i \models \mathbf{F^{-1}}\ \phi$

Exercice 2 : *Evaluer les formules*

On considère le STE $\mathcal{S}$ de la figure ci-contre (q_0 est l'état initial). Pour chacune des formules suivantes, dire si la formule est vraie pour $\mathcal{S}$ (*ie* pour toutes ses exécutions). Justifier les réponses.



$\mathbf{X}a \quad \mathbf{X}b \quad \mathbf{F}a \quad \mathbf{G}a \quad \mathbf{GF}a \quad \mathbf{GF}c \quad \mathbf{G}(a \Rightarrow \mathbf{X}b)$
$\mathbf{FG}c \Rightarrow \mathbf{G}(a \Rightarrow \mathbf{X}b) \qquad \mathbf{FG}c \Rightarrow \mathbf{G}(b \Rightarrow \mathbf{X}a) \qquad \mathbf{GF}b$



1. $\Psi_1 = \mathbf{G}\Big(b \Rightarrow (\mathbf{X}(a \vee b))\Big)$
2. $\Psi_2 = \mathbf{G}\Big(a \Rightarrow (a \,\mathbf{U}\, b)\Big)$
3. $\Psi_3 = \mathbf{GF}a$
4. $\Psi_4 = \mathbf{GF}b$
5. $\Psi_5 = (\mathbf{FG}b) \Rightarrow (\mathbf{G}(a \Rightarrow \mathbf{X}b))$
6. $\Psi_6 = (\mathbf{FG}b) \Rightarrow (\mathbf{GF}a)$
7. $\Psi_7 = ((\neg a)\,\mathbf{U}\,(a \wedge \mathbf{XG}\neg a)) \Rightarrow \mathbf{FG}b$

Pourquoi utiliser la logique temporelle ?

- une bonne expressivité

→ on peut exprimer beaucoup de choses

- une sémantique naturelle,

→ facilement et succinctement

- de bonnes propriétés de décision

→ des algorithmes et des outils

- beaucoup d'extensions

→ pour les systèmes probabilistes, temps-réel, les jeux, les données,…

Problèmes de vérification

Model-checking:
input: un modèle (STE) $\mathbf{S}$ et une formule φ
output: oui ssi $\mathbf{S} \models \varphi$.

Satisfaisabilité:
input: une formule φ
output: oui ssi il existe un modèle $\mathbf{S}$ t.q. $\mathbf{S} \models \varphi$.
(+ $\mathbf{S}$ si il existe !)

Synthèse de contrôleur:
input: un modèle partiel $\mathbf{S}$ une formule φ
output: un « controleur » $\boldsymbol{C}$ t.q. $\mathbf{S} \mathrm{x} \boldsymbol{C} \models \varphi$.

Spécifier un système réactif

On distingue plusieurs grandes familles de propriétés:

Propriétés de sûreté (safety):
“une mauvaise chose n’arrive jamais”.
Ex: il y a au plus un processus en section critique.

Propriétés de vivacité (liveness):
“de bonnes choses arrivent un jour”.
Ex: chaque demande d’accès à la SC est satisfaite un jour.

Propriétés d’équité (fairness):
→ Vérification d’exécution équitable.
Ex: Chaque processus doit « avancer » infiniment souvent.