Q13

~~5.3~~ 5.2 Fixpoints

Modèle M, état s -> satisfaction : $M,s\models\phi$ vérifie si la formule ϕ est valide sur s
Les algorithmes de model checking fixent un modèle Krypke $M = \langle S,\rightarrow,AP,v\rangle$
et une formule ϕ et calculent $[\phi]_M=\{s\in S | M,s\models\phi\}$: dénotation de ϕ
AP = propositions atomiques, v indique quels états vérfient chaque AP
$[\phi]_M$ = l'ensemble d'états satisfaisant la formule
But : trouver un algorithme calculant $[\phi]_M$

$X = F(X)$

Fixpoints : (Une manière d'exprimer les algorithmes de model checking)
Soit une fonction $F\subseteq P(S)\rightarrow P(S)$ (lie des ensembles d'états à d'autres)
$X\subseteq S$ est un point fixe de F <=> $X=F(X)$
Si $[EF_n\phi] = [EF_{n+1}\phi]$ = $[\phi \text{ OR } EX\ EF_n\phi]$ = $[\phi]\cup pre_\exists([EF_n\phi])$ -> rien de nouveau
-> $[EF_n\phi]$= point fixe de $F=\lambda Y.[\phi]\cup pre_\exists(Y)$
($[EF_n\phi]=[EF\phi]$ => $[EF\phi]$ = point fixe de F)
$[EF_n\phi]$ (et $[EF\phi]$) est le plus petit point fixe de F car on part de l'ensemble vide ($EF_0\phi$)
et on ajoute de nouveaux ensembles d'états (chemins possibles de longueur i+1)

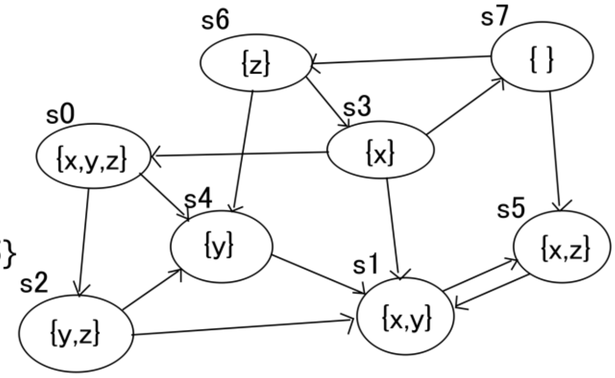
$pre_\exists([Y]) =\{s\in S| \exists s'\in S, s\rightarrow s' \text{ et } s'\in Y\}$
$pre_\forall([Y]) =\{s\in S| \forall s'\in S, s\rightarrow s' \Rightarrow s'\in Y\}$

l ensemble s'agrandit ou est é

Théorème :
Si $F\in P(S)\rightarrow P(S)$ est $\boxed{\text{monotone}}$ ($X\subseteq Y\rightarrow F(X)\subseteq F(Y)$) alors
- $F^n(\emptyset)$ est le plus petit point fixe de F (F^n est n compositions de F)
- $F^n(S)$ est le plus grand point fixe de F

Opérateurs CTL :
$\mu Y.F(Y)$ = plus petit point fixe de F, $\nu Y.F(Y)$ = plus grand point fixe de F
$[EF\phi] = \mu Y.[\phi]\cup pre_\exists(Y)$ (on commence par $Y=\emptyset$ et on atteint tous les états satisfaisants $EF\phi$)
$[EG\phi] = \nu Y.[\phi]\cap pre_\exists(Y)$ (on commence par Y=S et on réduit pour garder les états satisfaisants $EG\phi$)
$[AF\phi] = \mu Y.[\phi]\cup pre_\forall(Y)$ Tu dois vérifier ϕ et être le prédécesseur d'un état qui vérifie ϕ
$[AG\phi] = \nu Y.[\phi]\cap pre_\forall(Y)$
$[E[\phi U\psi]] = \mu Y.[\psi]\cup([\phi]\cap pre_\exists(Y))$
$[A[\phi U\psi]] = \mu Y.[\psi]\cup([\phi]\cap pre_\forall(Y))$

Méthode des points fixes pour [AF AG x]
i) $[x] = v(x)=\{s0, s1, s3, s5\}$
ii) $[AG\ x] = \nu Y.[x]\cap pre_\forall(Y)$
iii) $[AF\ AG\ x] = \mu Y.[AGx]\cup pre_\forall(Y)$



ii) $[AGx]=\nu Y.[x]\cap pre_\forall(Y)$:
(1) $\{s0,s1,s3,s5\}\cap pre_\forall(\{s0,s1,s2,s3,s4,s5,s6,s7\})$ (Y=S)
(1') $\{s0,s1,s3,s5\}\cap\{s0,s1,s2,s3,s4,s5,s6,s7\} = \{s0,s1,s3,s5\}$
(2) $\{s0,s1,s3,s5\}\cap pre_\forall(\{s0,s1,s3,s5\})$
(2') $\{s0,s1,s3,s5\}\cap\{s1,s4,s5\} = \{s1,s5\}$
(3) $\{s0,s1,s3,s5\}\cap pre_\forall(\{s1,s5\})$
(3') $\{s0,s1,s3,s5\}\cap\{s1,s4,s5\} = \{s1,s5\}$ -> point fixe
-> $[AGx]=\{s1,s5\}$

iii) $[AF\ AGx]=\mu Y.[AGx]\cup pre_\forall(Y)$:
(1) $\{s1,s5\}\cup pre_\forall(\emptyset)=[s1,s5]$ ($Y=\emptyset$)
(2) $\{s1,s5\}\cup pre_\forall(\{s1,s5\})$
(2') $\{s1,s5\}\cup\{s1,s4,s5\} = \{s1,s4,s5\}$
(3) $\{s1,s5\}\cup pre_\forall(\{s1,s4,s5\})$
(3') $\{s1,s5\}\cup\{s1,s2,s4,s5\} = \{s1,s2,s4,s5\}$
(4) $\{s1,s5\}\cup pre_\forall(\{s1,s2,s4,s5\})$
(4') $\{s1,s5\}\cup\{s0,s1,s2,s4,s5\} = \{s0,s1,s2,s4,s5\}$
(5) $\{s1,s5\}\cup pre_\forall(\{s0,s1,s2,s4,s5\})$
(5') $\{s1,s5\}\cup\{s0,s1,s2,s4,s5\} = \{s0,s1,s2,s4,s5\}$ -> point fixe
-> $[AF\ AGx]=\{s0,s1,s2,s4,s5\}$

[redacted] Operators implementation (algorithms)

But : trouver un algorithme calculant $[\phi]_M$, l'ensemble d'états satisfaisant la formule ϕ sur un modèle M

Définition de $[\phi]$ récursivement à partir de ϕ, ψ et $p \in AP$:
[t] = S (constante true est satisfaite dans tous les états)
[f] = Ø (false jamais satisfait)
[p]=v(p) ->v définit les états qui satisfont les APs
opérateurs logiques :
$[\neg\phi] = S - [\phi]$ (complément sur l'ensemble des états)
$[\phi \text{ OR } \psi] = [\phi] \cup [\psi]$
$[\phi \text{ AND } \psi] = [\phi] \cap [\psi]$
Plus complexe pour les autres formules ex : $[EX\phi]$, $[AX\phi]$, $[EF\phi]$, $[EG\phi]$, ...

Pour $[EX\phi]$: Calculer les prédessesseurs de ϕ : $pre([\phi])$
Donc $[EX\phi] = pre_\exists([\phi])$ et $[AX\phi] = pre_\forall([\phi])$ avec
$pre_\exists([Y]) = \{s \in S \mid \exists s' \in S, s \to s' \text{ et } s' \in Y\}$, Y = ensemble d'états
$pre_\forall([Y]) = \{s \in S \mid \forall s' \in S, s \to s' \Rightarrow s' \in Y\}$

Pour EF et EG le chemin n'est pas connu, il peut être de longueur variable
EF :
$[EF\phi] = \{s \in S \mid \exists \text{path } \pi \text{ tel que } \pi(0)=s, \exists i\ \pi(i) \models \phi\}$
Besoin d'une définition récursive : l'état satisfait ϕ ou le prochain
$EF_i\phi$ = il existe un chemin de longueur i dans lequel à un moment ϕ est satisfait
$EF_0\phi$ = f (false)
$EF_{i+1}\phi = \phi \text{ OR } EX\ EF_i\phi$
Donc :
$EF_1\phi = \phi$
$EF_2\phi = \phi \text{ OR } EX\ \phi$
$EF_3\phi = \phi \text{ OR } EX(\phi \text{ OR } EX\ \phi)$
Or, si $|S|=n$ et $k>n$ alors $[EF_k\phi] = [EF_n\phi]$
Donc, $[EF\phi]=[EF_n\phi]$ et
$[EF_0\phi] = \emptyset$
$[EF_{i+1}\phi] = [\phi] \cup pre_\exists([EF_i\phi])$
->Processus pour calculer $[EF\phi]$
Par contre nous devrions nous arrêter si nous atteignons un point fixe et ne pas calculer les n étapes (prendre le prédécesseur ne change rien)

EG : ϕ est satisfait sur tous le chemin
$EG_0\phi$ = t (true, chemin inexistant)
$EG_{i+1}\phi = \phi \text{ AND } EX\ EG_i\phi$ (conjonction)
Donc :
$EG_1\phi = \phi$
$EG_2\phi = \phi \text{ AND } EX\ \phi$
$EG_3\phi = \phi \text{ AND } EX(\phi \text{ AND } EX\ \phi)$
Si $k>n$: $[EG_k\phi]=[EG_n\phi]=[EG\phi]$

Les autres peuvent être trouvés par équivalence avec les opérateurs d'existence :
$AF\phi \equiv \phi \text{ OR } AX(AF\phi) \to [AF\phi]=\mu Y.[\phi] \cup pre_\forall([Y])$
$AG\phi \equiv \phi \text{ AND } AX(AG\phi) \to [AG\phi]=\nu Y.[\phi] \cap pre_\forall([Y])$
$E[\phi \cup \psi] \equiv \psi \text{ OR } (\phi \text{ AND } EX(E[\phi \cup \psi]) \to [E[\phi \cup \psi]]= \mu Y.[\psi] \cup ([\phi] \cap pre_\exists([Y])$
$A[\phi \cup \psi] \equiv \psi \text{ OR } (\phi \text{ AND } AX(A[\phi \cup \psi]) \to [A[\phi \cup \psi]]=\mu Y.[\psi] \cup ([\phi] \cap pre_\forall([Y])$

Avec la méthode des points fixes :
$\mu Y.F(Y)$ pour le plus petit point fixe de F, $\nu Y.F(Y)$ pour le plus grand point fixe de F