13

Introduction à la vérification - DM

Moro Benjamin

27 mars 2021

1 Exercice 1 :

On considère la formule CTL suivante :

$$\Phi = \mathbf{AF}(b \wedge \neg a) \rightarrow \mathbf{AFA}(b\mathbf{W}(\neg a \wedge \neg b))$$

1.1 Question 1 :

Mettez Φ en forme normale existentielle.

Dans un premier temps nous allons décomposer cette formule :

1. On commence par mettre le weak-until en forme normale existentielle :

$$\neg\Phi_W = \mathbf{A}(b\mathbf{W}(\neg a \wedge \neg b))$$
$$\neg\Phi_W = \neg\mathbf{E}(((a \vee b) \wedge b)\ \mathbf{U}\ (\neg b \wedge (\neg(\neg a \wedge \neg b))))$$
$$\neg\Phi_W = \neg\mathbf{E}(b\ \mathbf{U}\ (\neg b \wedge a))$$

   On a donc au final :

$$\Phi = \mathbf{AF}(b \wedge \neg a) \rightarrow \mathbf{AF}\neg\Phi_W$$

2. Ensuite on casse l'implication :

$$\Phi = \neg(\mathbf{AF}(b \wedge \neg a)) \vee \mathbf{AF}\neg\Phi_W$$

3. Enfin, on peut passer en forme normale existentielle. Prenons $\Phi_1 = \neg(\mathbf{AF}(b \wedge \neg a))$ et $\Phi_2 = \mathbf{AF}\neg\Phi_W$. On a donc $\Phi = \Phi_1 \vee \Phi_2$.
   Soit $\Psi_1 = ENF(\Phi_1)$ et $\Psi_2 = ENF(\Phi_2)$:

$$\Psi_1 = \mathbf{EG}(\neg b \vee a)$$
$$\Psi_2 = \neg\mathbf{EG}(\neg\neg\Phi_W))$$
$$\Psi_2 = \neg\mathbf{EG}(\Phi_W))$$

   On obtient Ψ :

$$\Psi = \Psi_1 \vee \Psi_2$$
$$\Psi = \mathbf{EG}(\neg b \vee a) \vee \neg\mathbf{EGE}(b\ \mathbf{U}\ (\neg b \wedge a))$$

1.2 Question 2 :

Calculez Sat(Ψ) en suivant l'algorithme vu en cours sur le système de transitions.

Découpons Ψ en sous formule tel que $\Psi = \Psi_1 \vee \neg\mathbf{EG}\Psi_2$

1. $\Psi_1 = \mathbf{EG}(\neg b \vee a)$, Calculons Sat($\Psi_1$). Ψ_1 est de la forme $\mathbf{EG}\ x$. Par conséquent, on recherche le plus grand ensemble $T \subseteq S$ tel que $T \subseteq Sat(x)$ et que si $s \in T$ avec s un sommet, alors il existe un successeur de s qui appartient à T.

   $$Sat(\neg b \vee a) = \{s_0, s_1, s_4, s_5\}$$

   Utilisons l'algorithme vu en cours :

T	V
s_0	s_2
s_1	s_3
s_4	
s_5	

TABLE 1 – Initialisation de l'algorithme pour $Sat(\Psi_1)$

   Dans un premier temps, on regarde si s_0 a au moins un successeur dans T. La réponse est oui. s_0 et s_1 sont tous les deux un des successeurs de l'autre. Maintenant observons s_4, son seul successeur est s_3 et s_3 n'est pas dans T par conséquent, s_4 n'est pas dans T. Du côté de s_5, le problème est le même, son seul successeur est s_4 or, on vient de montrer que s_4 n'est pas dans T, donc s_5 n'est pas dans T non-plus.

T	V
s_0	s_2
s_1	s_3
	s_4
	s_5

TABLE 2 – État final de l'algorithme pour $Sat(\Psi_1)$

   On a donc : $Sat(\Psi_1) = \{s_0, s_1\}$

2. $\Psi_2 = \mathbf{E}(b\ \mathbf{U}\ (\neg b \wedge a))$, comme pour Ψ_1, calculons Sat(Ψ_2). Ici, Ψ_2 contient un *Until*. On recherche donc le plus petit ensemble $T \subseteq S$ tel que $Sat(\neg b \wedge a) \subseteq T$ et pour tout sommet $s \in Sat(b)$, si s détient un successeur dans T alors il est lui aussi dans T.
   Calculons $Sat(\neg b \wedge a)$:

   $$Sat(\neg b \wedge a) = \{s_5\}$$

   Calculons $Sat(b)$:

   $$Sat(b) = \{s_0, s_2, s_3\}$$

T	$Sat(b)$
s_5	s_0
	s_2
	s_3

TABLE 3 – État initial de l'algorithme pour $Sat(\Psi_2)$

   Prenons les éléments de la seconde colonne et observons ceux qui ont, dans un premier temps, s_5 comme successeur. Seul s_2 a s_5 comme successeur, s_2 est donc dans l'ensemble. Regardons maintenant les sommets ayant s_2 comme successeur, s_0 est dans cette catégorie, s_0 est donc dans l'ensemble. s_3 n'a aucun successeur dans l'ensemble, il n'appartient donc pas à celui-ci.

3. Ensuite on constate que cet ensemble est dans un $\mathbf{EG}$, cela veut dire que tous les éléments présent dans cette ensemble détienne un successeur dedans. s_5 et s_2 forment un circuit, s_0 détient s_2 comme successeur, ces 3 états sont donc dans l'ensemble final.
4. Ici, le $\mathbf{EG}$ est précédé d'un $\neg$ on récupère donc l'ensemble complémentaire. On obtient l'ensemble $\{s_1, s_3, s_4\}$.
5. Enfin, on effectue l'union des 2 ensembles (car leurs propositions logique sont séparées par un $\vee$). La solution est donc : $S = \{s_0, s_1, s_3, s_4\}$

Ce résultat est cohérent, car lorsqu'on observe le système de transition, les seuls états vérifiant $\mathbf{AF}(b \wedge \neg a)$ sont les sommets s_2, s_3, s_4 et s_5, les autres états ne vérifie pas cette propriété. Or cette propriété est la condition d'une implication et $\bot \rightarrow \Phi = \top$. On a donc la propriété vrai pour tous les sommets sauf $\{s_2, s_3, s_4, s_5\}$.

Vérifions si la deuxième partie de l'implication est vrai pour $\{s_2, s_3, s_4, s_5\}$. $\mathbf{AFA}(b\mathbf{W}(\neg a \wedge \neg b))$, ici on peut interpréter la formule ainsi : "Tous les successeur de tous les chemin issu des états vérifiant $\mathbf{AF}(b \wedge \neg a)$ vérifient un jour $(b\mathbf{W}(\neg a \wedge \neg b))$".

Le sommet s_4 vérifie le second membre du weak-until $(\neg a \wedge \neg b)$, il vérifie donc le weak-until.

Le sommet s_3 peut soit boucler sur lui-même, or s_3 vérifie b donc il vérifie le premier membre du weak-until, soit aller vers s_4, or s_4 vérifie le weak-until. On peut donc conclure que s_3 vérifie aussi le weak-until.

Par contre, s_2 et s_5 forment un circuit entre eux, et ce circuit ne vérifie pas $\mathbf{AFA}(b\mathbf{W}(\neg a \wedge \neg b))$, la seconde proposition est donc fausse, l'implication n'est pas vérifiée. s_2 et s_5 ne sont donc pas dans l'ensemble des solutions.

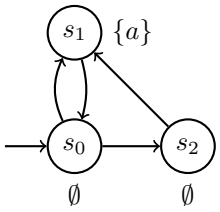
Nous avons donc l'ensemble de solution : $S = \{s_0, s_1, s_3, s_4\}$

2 Exercice 2 :

Pour chacune des paires de formules LTL/CTL ci-dessous, déterminez si elles sont équivalentes. Justifiez votre réponse soit par une preuve d'équivalence, ou par un système de transitions qui satisfait une des formules, mais pas l'autre.

2.1 $\mathbf{FX}a$ et $\mathbf{AFAX}a$:

Prenons ce système de transition :



Toutes les exécutions vérifie la formule LTL $\mathbf{FX}a$ car elles passent toutes par s_1 qui vérifie a (et passe donc par un état vérifiant $\mathbf{X}a$) Cependant, la formule CTL n'est pas vérifiée car on peut interpréter la formule $\mathbf{AFAX}a$ comme "Toutes les exécutions possible détiennent un jour un état ayant tous ses successeurs vérifiant a". Or, l'exécution $((s_0, s_1)^\omega)$ ne vérifie pas la formule car ni s_0, ni s_1 ont tous leurs successeurs vérifiant a. Les formules ne sont donc pas équivalente.

2.2 $\mathbf{F}(a \wedge \mathbf{F}b)$ et $\mathbf{AF}(a \wedge \mathbf{AF}b)$:

Montrons que $\mathbf{F}(a \wedge \mathbf{F}b)_{LTL} \equiv \mathbf{AF}(a \wedge \mathbf{AF}b)_{CTL}$:

Un système de transition qui vérifie la formule LTL est un système dont toutes les exécutions vérifient cette formule, on peut donc obtenir des chemins de la forme $(s_{init}, ..., s_i, ..., s_j, ...)$ où s_i vérifie a et s_j vérifie b. Ces chemins vérifient donc la formule LTL.

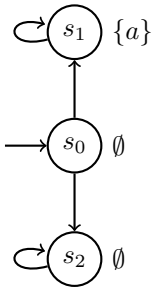
D'un point de vue CTL, l'arbre formé par l'ensemble de ces chemins détient notamment 2 coupes, une première coupe uniquement composé des états s_i vérifiant a, et un seconde coupe, en dessous de la première uniquement composé des états s_j vérifiant b. Le fait qu'on obtienne 2 coupes nous permet de conclure que la formule CTL vérifie $\mathbf{AF}a$ et $\mathbf{AF}b$. Le fait que les coupes soient ordonnées nous permet de conclure que lorsqu'on vérifie a, on vérifiera b plus tard. Et donc on obtient bien la formule $\mathbf{AF}(a \wedge \mathbf{AF}b)$, on vérifie d'abord a, puis ensuite b.

Montrons maintenant que $\mathbf{AF}(a \wedge \mathbf{AF}b)_{CTL} \equiv \mathbf{F}(a \wedge \mathbf{F}b)_{LTL}$:
La méthode est ici la même que précédemment, on observe l'arbre issu de l'exécution d'un système de transition vérifiant notre formule CTL. Les mêmes coupes dans cette arbre nous permettes de conclure que la formule LTL est bien équivalente à la formule CTL.

Les 2 formules sont donc équivalente.

2.3 $\mathbf{GX}a$ et $\mathbf{EGAX}a$:

Prenons ce système de transition :



Ici on peut observer que la formule CTL est vérifiée puisqu'il existe un chemin où tous les états se succédant vérifient a, c'est le chemin (s_0, s_1^ω). Cependant, la formule LTL n'est pas vérifiée car le chemin (s_0, s_2^ω) ne vérifie jamais a. Or pour qu'un système de transition vérifie une formule LTL, l'ensemble des exécutions possible doit vérifier la formule en question. Les formules ne sont donc pas équivalente.

3 Exercice 3 :

Soit ST un système de transitions avec un ensemble d'états S. Justifiez les assertions suivantes :

3.1 Assertion 1 :

$Sat(\mathbf{A}(\Phi\ \mathbf{U}\ \Psi))$ *est le plus petit ensemble* $T \subseteq S$ *qui contient* $Sat(\Psi)$ *et qui satisfait :*

$$\{s \in Sat(\Phi)\ :\ post(s)\ \subseteq\ T\}\ \subseteq\ T$$

Par définition, $Sat(\mathbf{E}(\alpha\ \mathbf{U}\ \beta))$ est le plus petit ensemble $T \subseteq S$ tel que $Sat(\beta) \subseteq T$ et que pour tous sommets $s \in Sat(\alpha)$ si s détient un successeur dans T alors $s \in T$.
Et, $Sat(\mathbf{EG}\alpha)$ est le plus grand ensemble $T \subseteq S$ tel que $T \subseteq Sat(\alpha)$ et que pour tous sommets $s \in T \rightarrow \exists\ post(s) \in T$
On sait que

$$Sat(\mathbf{A}(\Phi\ \mathbf{U}\ \Psi)) = Sat(\neg(\mathbf{E}(\neg\Psi\ \mathbf{U}\ (\neg\Phi \wedge \neg\Psi)) \vee \mathbf{EG}\neg\Psi))$$
$$Sat(\mathbf{A}(\Phi\ \mathbf{U}\ \Psi)) = Sat(\neg\mathbf{E}(\neg\Psi\ \mathbf{U}\ (\neg\Phi \wedge \neg\Psi))) \cap Sat(\neg\mathbf{EG}\neg\Psi))$$

cette réécriture avec E U, EG ne mène pas au résultat escompté ...

Calculons $Sat(\mathbf{E}(\neg\Psi\ \mathbf{U}\ (\neg\Phi \wedge \neg\Psi)))$: l'ensemble obtenu contient par propriété $Sat(\neg\Phi \wedge \neg\Psi)$, il est possible qu'il contienne des éléments supplémentaire en fonction du système de transition. Notre ensemble $T_{\mathbf{U}}$ est donc compris entre :

$$Sat(\neg\Phi \wedge \neg\Psi) \subseteq T_{\mathbf{U}} \subseteq Sat(\neg\Phi \wedge \neg\Psi) \cup Sat(\neg\Psi)$$

L'ensemble qui nous intéresse est le complémentaire de cette ensemble :

$$Sat(\Phi \vee \Psi) \supseteq S \setminus T_{\mathbf{U}} \supseteq (Sat(\Phi \vee \Psi) \cap Sat(\Psi))$$
$$((Sat(\Phi) \cup Sat(\Psi)) \cap Sat(\Psi)) \subseteq S \setminus T_{\mathbf{U}} \subseteq (Sat(\Phi) \cup Sat(\Psi))$$
$$Sat(\Psi) \subseteq S \setminus T_{\mathbf{U}} \subseteq (Sat(\Phi) \cup Sat(\Psi))$$

Dans une deuxième partie, calculons $Sat(\mathbf{EG}\neg\Psi))$: là aussi, par propriété, on peut obtenir les inclusions suivantes :

$$\emptyset \subseteq T_{\mathbf{G}} \subseteq Sat(\neg\Psi)$$

Ici aussi, l'ensemble qui nous intéresse est le complémentaire :

$$Sat(\Psi) \subseteq S \setminus T_{\mathbf{G}} \subseteq S$$

On a donc, pour $Sat(\mathbf{A}(\Phi\ \mathbf{U}\ \Psi))$ l'ensemble suivant $T_{\mathbf{A}}$:

$$(Sat(\Psi) \cap Sat(\Psi)) \subseteq (S \setminus T_{\mathbf{U}} \cap S \setminus T_{\mathbf{G}}) \subseteq ((Sat(\Phi) \cup Sat(\Psi)) \cap S)$$
$$Sat(\Psi) \subseteq (S \setminus T_{\mathbf{U}} \cap S \setminus T_{\mathbf{G}}) \subseteq (Sat(\Phi) \cup Sat(\Psi))$$
$$Sat(\Psi) \subseteq T_{\mathbf{A}} \subseteq (Sat(\Phi) \cup Sat(\Psi))$$

On constate donc que $Sat(\Psi)$ est bien inclus dans $T_{\mathbf{A}}$ et que, les éléments supplémentaire dans $T_{\mathbf{A}}$ n'étant pas dans $Sat(\Psi)$ sont obtenus lorsqu'on ajoute des éléments de $Sat(\neg\Phi)$ dans $T_{\mathbf{U}}$. Ces éléments, avec le passage au complémentaire, seront dans $Sat(\Phi)$ et ne seront dans $T_{\mathbf{A}}$ que si $s \in Sat(\Phi) \rightarrow \exists\ post(s) \subseteq T$.

3.2 Assertion 2 :

$Sat(\mathbf{AG}\Phi)$ *est le plus grand ensemble* $T \subseteq S$ *qui satisfait :*

$$T \subseteq \{s \in Sat(\Phi) : post(s) \subseteq T\}$$

On sait que

$$Sat(\mathbf{AG}\Phi) = Sat(\neg\mathbf{EF}\neg\Phi)$$
$$Sat(\mathbf{AG}\Phi) = Sat(\neg\mathbf{E}(\top\ \mathbf{U}\ \neg\Phi))$$

De manière similaire à l'assertion précédente, calculons l'encadrement de $T_{\mathbf{U}}$ l'ensemble des états de S vérifiant $\mathbf{E}(\top\ \mathbf{U}\ \neg\Phi)$:

$$Sat(\neg\Phi) \subseteq T_{\mathbf{U}} \subseteq (Sat(\neg\Phi) \cup Sat(\top))$$
$$Sat(\neg\Phi) \subseteq T_{\mathbf{U}} \subseteq S$$

Prenons les complémentaires de ces ensembles :

$$\emptyset \subseteq S \setminus T_{\mathbf{U}} \subseteq Sat(\Phi)$$

On obtient donc un encadrement pour $T_{\mathbf{A}}$:

$$\emptyset \subseteq T_{\mathbf{A}} \subseteq Sat(\Phi)$$

On a recherché le plus petit ensemble complémentaire vérifiant $Sat(\neg\mathbf{E}(\top\ \mathbf{U}\ \neg\Phi))$, par complémentarité, on obtient le plus grand ensemble pour $Sat(\mathbf{AG}\Phi)$. Cet ensemble est bien inclus dans S puisqu'il est inclus dans $Sat(\Phi)$ et que $Sat(\Phi)$ est inclus dans S. De plus, par construction, les éléments présent à l'intérieur de $T_{\mathbf{A}}$ ne sont présent que si il détiennent un successeur également dans $T_{\mathbf{A}}$.

(*) c'est le même chose que ce que vous avez trouvé pour AW, ce qui montre (aussi) que ce n'est pas suffisant pour montrer les assertions demandées.

3.3 Assertion 3 :

$Sat(\mathbf{A}(\Phi\ \mathbf{W}\ \Psi)$ *est le plus grand ensemble* $T \subseteq S$ *qui satisfait :*

$$T \subseteq Sat(\Psi) \cup \{s \in Sat(\Phi) : post(s) \subseteq T\}$$

On sait que :

$$Sat(\mathbf{A}(\Phi\ \mathbf{W}\ \Psi) = Sat(\neg\mathbf{E}((\Phi \wedge \neg\Psi)\ \mathbf{U}\ (\neg\Phi \wedge \neg\Psi)))$$

Ré-appliquons la procédure précédente pour encadrer l'ensemble solution $T_{\mathbf{W}}$ sur

$$Sat(\mathbf{E}((\Phi \wedge \neg\Psi)\ \mathbf{U}\ (\neg\Phi \wedge \neg\Psi)))$$

On notera $T_{\mathbf{U}}$ l'ensemble satisfaisant la formule ci-dessus. Par complémentarité, $T_{\mathbf{W}} = S \setminus T_{\mathbf{U}}$

$$\begin{aligned} Sat(\neg\Phi \wedge \neg\Psi) &\subseteq T_{\mathbf{U}} \subseteq Sat(\neg\Phi \wedge \neg\Psi) \cup Sat(\Phi \wedge \neg\Psi) \\ Sat(\neg\Phi) \cap Sat(\neg\Psi) &\subseteq T_{\mathbf{U}} \subseteq (Sat(\neg\Phi) \cap Sat(\neg\Psi)) \cup (Sat(\Phi) \cap Sat(\neg\Psi)) \\ Sat(\neg\Phi) \cap Sat(\neg\Psi) &\subseteq T_{\mathbf{U}} \subseteq Sat(\neg\Psi) \end{aligned}$$

$T_{\mathbf{U}}$ est donc le plus petit ensemble satisfaisant la négation de $Sat(\mathbf{A}(\Phi\ \mathbf{W}\ \Psi)$, par complémentarité, on peut donc obtenir $T_{\mathbf{W}}$:

(*) $$Sat(\Psi) \subseteq T_{\mathbf{W}} \subseteq Sat(\Phi) \cup Sat(\Psi)$$

D'après ce calcul, $Sat(\mathbf{A}(\Phi\ \mathbf{W}\ \Psi)$ correspond au plus grand ensemble (car on a recherché le plus petit ensemble complémentaire) incluant $Sat(\Psi)$ et ayant des éléments de $Sat(\Phi)$ si ceux-ci ont des successeurs dans $T_{\mathbf{W}}$.

On a donc $Sat(\mathbf{A}(\Phi\ \mathbf{W}\ \Psi))$ est le plus grand ensemble $T \subseteq S$ qui contient $Sat(\Psi)$ et qui satisfait :

$$\{s \in Sat(\Phi)\ :\ post(s)\ \subseteq\ T\}\ \subseteq\ T$$

Le résultat est plus fort ici que l'hypothèse qu'on souhaitait montrer car on sait que $Sat(\Psi)$ est inclus dans T grâce à la première partie de l'encadrement $Sat(\Psi) \subseteq T$. La seconde partie de l'encadrement $T_{\mathbf{W}} \subseteq Sat(\Phi) \cup Sat(\Psi)$ exprime que les éléments supplémentaire qui serait dans l'ensemble résultat appartiennent à $Sat(\Phi)$. Sauf que ces éléments sont ajoutés grâce à l'algorithme permettant de déterminé l'ensemble des états satisfaisant $Sat(\mathbf{E}(\alpha\ \mathbf{U}\ \beta))$. Cet algorithme applique la contrainte $\{s \in Sat(\Phi)\ :\ post(s)\ \subseteq\ T\}$ pour ajouter les sommets dans l'ensemble T. Nous avons donc au final la propriété qui est vérifiée.

4 Exercice 4 :

On considère le fragment ECTL de CTL définie par la grammaire :

$$\begin{aligned} \Phi &:= a \mid \neg a \mid \Phi \wedge \Phi \mid \mathbf{E}\varphi \\ \varphi &:= \mathbf{X}\Phi \mid \mathbf{G}\Phi \mid \Phi\ \mathbf{U}\ \Phi \end{aligned}$$

4.1 Question 1 :

Montrez que pour toute formule ECTL Φ *et* $S1$, $S2$ *tels que* $S1 \subseteq S2$ *:*

$$S1 \models \Phi \rightarrow S2 \models \Phi.$$

On sait que $S1$ est inclus dans $S2$, ce qui veut dire que $S1$ est un sous-système de $S2$. Par conséquent, si $S1 \models \Phi$ alors, $S2$, qui est un système plus grand incluant $S1$, vérifie aussi Φ.

il faut être plus précis : faire réc. sur la forme des formules + dire que tous les deniés de S_1 sont aussi des deniés de S_2 ($+ I_1 = I_2$), etc.

4.2 Question 2 :

Montrez qu'il existe des formules CTL qui ne sont équivalentes à aucune formule ECTL.

Prenons la formule CTL $(\neg a \vee \neg b)$. Cette formule n'est pas valide en ECTL puisque le $\vee$ ne fait pas partie de la grammaire ECTL. Essayons de transformer ce $\vee$ en $\wedge$. Grâce aux règles de De Morgan on peut obtenir le $\wedge$ par négation :

$$(\neg a \vee \neg b) = \neg(a \wedge b)$$

Ici, nous avons bien un $\wedge$ mais un autre problème survient avec la négation, celle-ci n'est autorisé que devant une variable élémentaire $\Phi = a \mid \neg a \mid ...$ ($\neg\Phi$ n'est pas autorisé). Essayons donc de trouver une formule ECTL, satisfaisant la table de vérité de $(\neg a \vee \neg b)$:

a	b	$(\neg a \vee \neg b)$
0	0	1
0	1	1
1	0	1
1	1	0

TABLE 4 – Table de vérité de $(\neg a \vee \neg b)$

Une telle formule n'existe pas puisque le $\vee$ n'appartient pas à la grammaire ECTL et que la seule autre manière d'exprimer une disjonction est de mettre un $\neg$ en facteur d'une formule contenant un $\wedge$ ($\neg(a \wedge b)$ par exemple est équivalent à $\neg a \vee \neg b$).

$(\neg a \vee \neg b)$ ne marche pas, car elles parlent que des états initiaux, or $I_1 = I_2$ et ils ont les mêmes étiquettes.
Il faut trouver une formule CTL ϕ tq. $S_1 \models \phi$, $S_2 \not\models \phi$!