

Examen – Modélisation et spécification

Master Informatique

Partie Spécification

15 Janvier 2020

Durée : 1h.

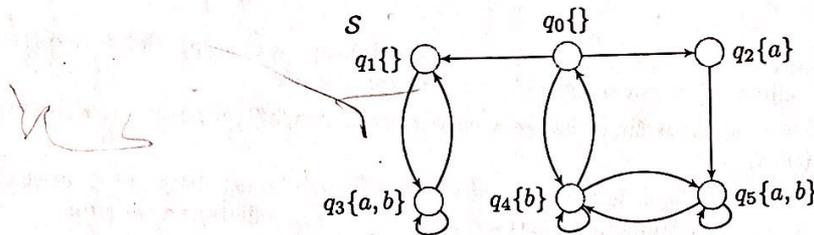
Documents autorisés : Une feuille A4 manuscrite recto-verso.

Rédaction : Il faut rendre deux copies : une pour les trois exercices de la partie spécification et une pour les exercices de la partie modélisation.

Exercice 1 :

Vérification de formules de CTL et LTL [2 points]

- On considère le système de transition \mathcal{S} de la figure ci-dessous (les propositions atomiques sont entre accolades) et la formule CTL $\Phi = \text{EF}(a \wedge \text{AXB})$. Indiquer pour chaque état les sous-formules de Φ vérifiées par l'état. Justifier vos réponses.



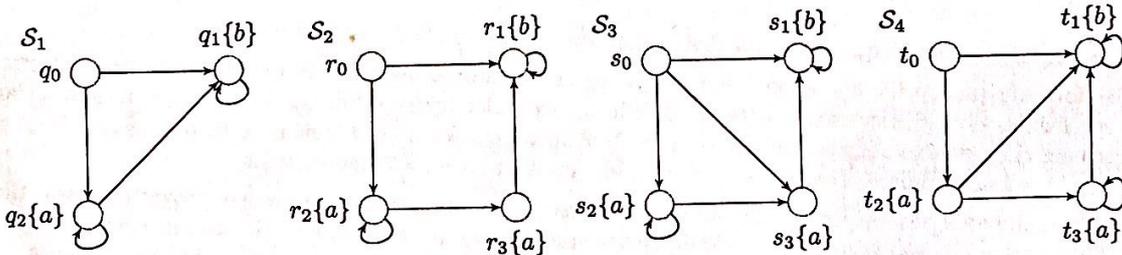
- Dire si les formules LTL suivantes $\Psi_1 = \text{G}(b \Rightarrow (\text{X}(a \vee b)))$ et $\Psi_2 = \text{G}(a \Rightarrow (a \text{ U } b))$ sont vraies pour le système \mathcal{S} en supposant que l'état initial est l'état q_0 . Justifier vos réponses.

Exercice 2 :

Logique temporelle et distinction de modèles [3 points]

On considère les trois modèles \mathcal{S}_1 , \mathcal{S}_2 et \mathcal{S}_3 ci-dessous. On suppose que les états initiaux sont q_0 , r_0 et s_0 .

- Peut-on distinguer \mathcal{S}_1 et \mathcal{S}_2 , \mathcal{S}_1 et \mathcal{S}_3 , et \mathcal{S}_2 et \mathcal{S}_3 avec CTL? Justifier vos réponses.
- Même question avec LTL. Justifier vos réponses.



Exercice 3 :

On s'intéresse ici à la spécification en LTL d'un lave-vaisselle. Le tableau de bord de la machine comporte quatre voyants lumineux (« on », « rinçage », « lavage », « séchage ») un sélecteur de programme avec deux positions (« éco » et « intensif »), et deux boutons (« start-reset » et « on/off »). La figure 1 représente ce tableau de bord.

On considère les propositions atomiques suivantes : chaque voyant a une proposition atomique associée qui est vraie lorsque le voyant est allumé (Von, Vrinçage, Vlavage, Vséchage), Péco est vraie lorsque le sélecteur de programme est sur « éco » (et Pintensif est vrai lorsqu'il est sur « intensif »). Bstart-reset est vrai au moment où le bouton « start-reset » est enfoncé, et Bon/off est vrai lorsque c'est le bouton « on/off » qui est enfoncé.

La machine peut se trouver dans deux états différents : allumé (proposition Pon) ou éteinte (proposition Poff). Lorsqu'elle est allumée, elle peut être dans trois modes différents : attente, marche, ou fin. A chacun de ces modes on associe une proposition atomique : attente, marche, ou fin.

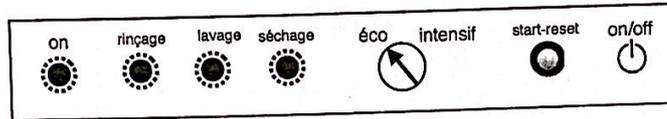


FIGURE 1 – Tableau de bord du lave-linge.

Écrire les formules suivantes :

1. Une formule exprimant que la machine est toujours dans un seul état (allumée ou éteinte). Ecrire une formule qui spécifie que le voyant Von est allumé si et seulement si la machine est allumée.
2. Une formule pour indiquer que l'état de la machine change avec le bouton Bon/off (appuyer sur le bouton la fait changer d'état à l'instant suivant).
3. Une formule qui spécifie qu'à chaque instant, si la machine est allumée, alors elle est dans un et un seul des trois modes (attente, marche, fin). Et une formule pour spécifier qu'à l'allumage, elle est en attente.
4. Une formule qui énonce qu'appuyer sur le bouton Bstart-reset lorsque la machine est en attente, la fait passer dans le mode marche.
5. Et une formule pour dire que si le bouton Bstart-reset est appuyé longtemps (= pendant 3 instants successifs), alors la machine passe en mode attente (quel que soit le mode dans lequel elle était auparavant).
6. Une formule $\phi(V)$ qui indique qu'un voyant V clignote (ici la proposition V est vraie ssi le voyant est allumé) : c'est-à-dire qu'il est allumé, puis éteint, puis allumé, etc. changeant à chaque instant (ou éteint puis allumé, puis éteint...). On supposera que ce clignotement ne s'arrête qu'avec l'arrêt de la machine (état Poff).
7. Une formule qui énonce que les voyants « rinçage », « lavage » et « séchage » clignotent lorsque la machine arrive dans le mode « fin ».

Bonus :

8. Une formule spécifiant qu'à tout moment un programme (et un seul) est sélectionné.
9. Une formule qui spécifie que lorsque la machine passe en mode « marche » et que le programme sélectionné est « intensif », alors elle va exécuter un cycle de lavage complet (c'est-à-dire que le voyant « rinçage » s'allumera pendant une certaine période, puis ce sera au tour du voyant « lavage » puis au tour du voyant « séchage ») à moins de s'interrompre en cas de retour au mode « attente » ou à l'état « éteint ».
10. Une formule qui spécifie que lorsque la machine passe en mode « marche » et que le programme sélectionné est « éco », alors elle va exécuter un cycle de lavage sans rinçage (c'est-à-dire que le voyant « lavage » s'allumera pendant une certaine période, puis ce sera au tour du voyant « séchage ») à moins de s'interrompre en cas de retour au mode « attente » ou à l'état « éteint ».

Examen – Modélisation et spécification

Master Informatique

Partie Modélisation

15 Janvier 2020

Durée : 1h.

Documents autorisés : Une feuille A4 manuscrite recto-verso.

Rédaction : Il faut rendre les exercices de cette partie sur une copie différente de la partie Spécification

Exercice 1 :

Analyse de réseaux de Petri [3 points]

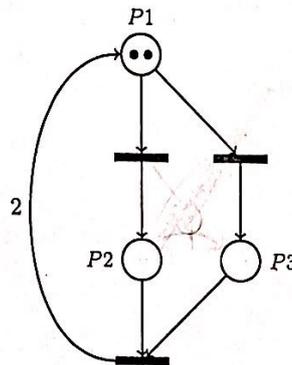


FIGURE 1 – Réseau de Petri RP_1

On considère le réseau de Petri RP_1 dessiné à la Figure 1.

1. Dessinez le graphe de marquages accessibles de ce réseau de Petri (que nous avons aussi appelé $ST(RP_1)$ en cours).
2. Modifiez en ajoutant une place (connectée aux transitions existantes) ce réseau de Petri pour que la transition mettant des jetons dans la place P_3 ne puisse être franchie que si la transition mettant des jetons dans la place P_2 a été franchie juste avant.
3. Le réseau de Petri donné initialement vérifie-t-il la formule LTL $GF(\#P_1 = 2)$? Justifiez votre réponse. Et qu'en est-il pour le réseau de Petri obtenu suite à la question 2?

Exercice 2 :

Modélisation par réseaux de Petri [7 points]

Dans cet exercice, le but est de construire différents réseaux de Petri pour modéliser le problème suivant sous différentes conditions. Quatre développeurs Linux et quatre développeurs Microsoft veulent traverser une rivière grâce à un bateau. Au début tout le monde est sur la rive A de la rivière tout comme le bateau. Les développeurs attendent de monter dans le bateau, ils montent dans le bateau un par un. Le bateau dispose de quatre places, si il est plein ou si il n'est pas sur la rive A, les développeurs ne peuvent plus monter dedans. Il n'y a (pour l'instant) pas de règle sur qui peut monter dans le bateau. Quand le bateau est plein, il part sur la rivière puis il arrive sur la rive B. Quand il arrive, les développeurs qui sont montés dessus peuvent descendre un par un et ils sont maintenant sur la rive B (où ils resteront). Quand le bateau se trouvant sur la rive B est vide, il retourne sur la rive A et il attend de nouveau d'être plein pour repartir.

1. Donnez une représentation par réseau de Petri de ce système. Il faudra distinguer les développeurs Linux et Microsoft sur la rive A et sur la rive B (et si vous le souhaitez aussi sur la bateau).
2. Donnez la formule LTL précisant que tous les développeurs Linux vont finir par arriver sur la rive B. NB : Pour Les formules LTL demandées, les propositions atomiques devront avoir la forme $\#p \sim a$ où p est le nom d'une place, a est un entier positif et \sim est un symbole dans $\{<, >, =, \geq, \leq\}$ ($\#p$ caractérisant le nombre de jetons dans la place p).
3. Votre système vérifie-t-il cette formule? Justifiez (brièvement) votre réponse.

On raffine un peu le système, maintenant le bateau ne part que si il est plein mais si il est parti et qu'il y a dessus trois développeurs Linux et un développeur Microsoft, il n'arrive jamais sur la rive B (il coule à cause probablement d'une insurrection des développeurs en majorité).

4. Proposez une nouvelle modélisation de ce système par un réseau de Petri.

5. Ce nouveau réseau vérifie-t-il la formule LTL décrite à la question 2? Justifiez (brièvement) votre réponse.

On raffine encore un peu le système en interdisant aux développeurs Linux sur la rive A de monter sur le bateau si il y a déjà au moins deux développeurs Linux dessus.

6. Proposez une nouvelle modélisation de ce système par un réseau de Petri.

7. Ce nouveau réseau vérifie-t-il la formule LTL décrite à la question 2? Justifiez (brièvement) votre réponse.