Université Paris 7 — DESS Logiciels Fondamentaux

U.F.R d'informatique — Année 2002-2003

TD 9 – Model Checking de CTL

1 Exemples d'applications

1. Sans utiliser la feuille de rappel, proposez directement un algorithme de graphe pour calculer $EF\phi$ connaissant $[|\phi|]$ **correction** : ce sont les predecesseurs
2. meme question pour EGy, en restant lineaire

   correction : Il faut chercher plus : les cycles sur y, puis les predecesseurs de ces etats. Bref un calcul des composantes fortements connexe (Tarjan), du sous graphe etiquete par y, puis les predecesseurs des composantes non triviales.
3. Effectuer le model checking de l'automate polycopié pour les formules $EF((x = z) \wedge (x \neq y))$ et EGy

   correction : On peut reecrire $EFp = EtrueUp$, $AFp = AtrueUp$, $EGp = \neg AF\neg p$, et $AGp = \neg EF\neg p$.
4. Sur l'exemple du micro-onde verifier la formule $AG(start \Rightarrow AFheat)$

2 Ecrire en CTL

correction : Ceci est un exercice de formalisation, les phrases proposees sont parfois ambigues. C'est par exemple lors de cette phase qu'on discute avec un client de ce qu'il veut. Bref les solutions ne sont pas forcements uniques et equivalentes.

- quoique je fasse maintenant, je garde la possibilite de faire a dans le futur.

  correction : $AXEFa$, ou avec une interpretation plus forte : $AX.AF.EF.a$
- Il existe à présent un choix après lequel je pourrais toujours choisir b

  correction : $EXEFb$
- Quelque soit l'execution,, tout p sera inévitablement suivit d'un q.

  correction : $AG(p \rightarrow AFq)$
- Quelque soit l'execution, tout p sera inévitablement suivit d'un q dans un futur strict.

  correction : $AG(p \rightarrow AXAFq)$
- chaque q impose que p devienne vrai avant une eventuelle occurrence de r

  correction : $AG(q \rightarrow A(\neg R\mathcal{U}p))$
- si un train passe la bariere est fermee

  correction : $AGpasse \rightarrow bar_fermee$
- je ne rentre que si le passage est vide, et je fermerai la porte si je ne reste pas dans le passage

  correction : $(\neg EF(r \wedge \neg v)) \wedge (AG(r \Rightarrow A(Greste \vee F\neg reste \wedge Ffermer)))$

- si les deux process sont en attente infiniment souvent, ils seront servit l'un apres l'autre.

  correction : $A[GFatt_1 \wedge GFatt_2 \Rightarrow \neg F(s_1 \wedge (\neg s_2)\mathcal{U}s_1 \vee s_2 \wedge (\neg s_1)\mathcal{U}s_2)]$
- si je l'allume au pont il est possible de garder la torche allumee jusqu'au bout du chemin

  correction : $A[(\neg allume\mathcal{U}pont) \Rightarrow F(allume \rightarrow allume\mathcal{U}bout]$
- Si nos chemins se separent sans qu'on l'ai decide ensemble, on pourra toujours se retrouver des lors qu'on le decide chacun de son cote.

  correction : Rq que ca exprime qq chose qui est de l'ordre de l'agenre de processus. $A(\neg(d_t \wedge d_m)\mathcal{U}\neg(moi \wedge toi) \Rightarrow F(\neg(toi \wedge moi \wedge EF(d1 \wedge F(d2 \wedge Ftoi \wedge moi)) \vee EF(d_2 \wedge F(d1 \wedge Ftoi \wedge moi))$
- la torche passe de pays en pays en n'etant portee que par un seul individu par pays, une et une seule fois

  correction : $AG(i_1 \wedge p_1\mathcal{U}X\neg p_1 \vee ...) \wedge \neg EF(p_1 \wedge p_2 \vee \neg p_1 \wedge \neg p_2)$
- si je ne peut obtenir un resultat par la force, je pourrais toujours le faire par la ruse

  correction : $A((Gforce \wedge \neg Fresult) \Rightarrow FEruse\mathcal{U}result)$
- jusqu'ici, quand je le desirais, j'arrivais à tourner d'une main en tapant de l'autre, mais depuis que j'ai 30 ans j'y arrive une fois sur deux, à moins qu'il ne fasse nuit.