TD 2

1 Propriétés qualitatives

Montrer que les 3 points suivants (discutés en cours) sont équivalents. NB : B est un sous-ensemble d'états d'une chaîne de Markov $\mathcal{M} = (S, P, s_0, \ell)$ finie. On rappelle que les formules de CTL s'interprètent sur la structure de Kripke induite par $\mathcal{M}$.

1. $Pr(s \models \mathsf{G}\,\mathsf{F}\ B) = 1$
2. $\mathcal{C} \cap B \neq \emptyset$ pour toute composante fortement connexe terminale $\mathcal{C}$ atteignable depuis s
3. $s \models \mathsf{AG}\ \mathsf{EF}\ B$

2 Modéliser et analyser

On considère l'algorithme probabiliste suivant. Un processus P choisit aléatoirement une fréquence de communication parmi n fréquences possibles (chacune est équiprobable). Ensuite P souhaite vérifier que celle-ci n'est pas encore utilisée par un autre processus (il y a p processus connectés qui utilisent déjà chacun une fréquence distincte, avec $p \leq n$) : pour cela, P va émettre un message sur cette fréquence et attendre un éventuel signal d'un autre processus : si il ne reçoit rien, il va réémettre le message car il est possible que le premier ait été perdu (probabilité de 1/10), et là encore si il ne reçoit rien, il va émettre le message une dernière fois, avant de considérer que la fréquence est disponible et qu'il peut l'utiliser normalement. Et à chaque étape, si il reçoit une réponse d'un autre processus, il retourne au début de son algorithme et choisit aléatoirement une fréquence etc.

On veut calculer la probabilité de trouver une fréquence non encore utilisée en fonction de n et p. Modéliser ce problème sous la forme d'une chaîne de Markov (votre chaîne devrait avoir moins de 10 états). Appliquer la résolution par équations. Pour $n = 100$ et $p = 20$, appliquer l'algorithme itératif d'approximation (quelques itérations suffisent...). Résoudre le problème avec PRISM (pour $n = 100$ et $p = 20$).

3 Equivalence de formules PCTL

Pour chaque cas ci-dessous, indiquer si c'est correct (à justifier !) ou erroné (donner un contre-exemple) :

- $\mathsf{P}_{=1}\mathsf{X}\,(\mathsf{P}_{=1}\mathsf{G}\ a) \stackrel{?}{\equiv} \mathsf{P}_{=1}\mathsf{G}\,(\mathsf{P}_{=1}\mathsf{X}\ a)$
- $\mathsf{P}_{>.5}\mathsf{X}\,(\mathsf{P}_{>.5}\mathsf{F}\ a) \stackrel{?}{\equiv} \mathsf{P}_{>.5}\mathsf{F}\,(\mathsf{P}_{>.5}\mathsf{X}\ a)$
- $\mathsf{P}_{=1}\mathsf{X}\,(\mathsf{P}_{=1}\mathsf{F}\ a) \stackrel{?}{\equiv} \mathsf{P}_{=1}\mathsf{F}\,(\mathsf{P}_{=1}\mathsf{X}\ a)$

4 Programmation

A faire en binôme. Programmer dans le langage de votre choix. la résolution du problème suivant :

Input : une chaine de Markov composée de n états $s_1, \ldots, s_n$ donnée sous la forme de sa matrice P et un état s_m avec $1 \leq m \leq n$.
Output : un vecteur $(x_i)_{1 \leq i \leq n}$ où x_i est une approximation (via le calcul itératif vu en cours) de la probabilité d'atteindre s_m depuis l'état s_i.

Appliquer votre programme à l'exemple de l'exercice 2.