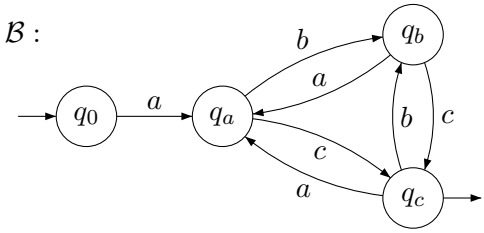
Master 2 SAR - MSR(Ni405) - Année 2012-2013
Examen du 16 novembre 2012
Notes de cours autorisées

Les exercices 3 et 4 sont à rendre sur une deuxième copie

Exercice 1

1. Dessiner un automate fini $\mathcal{A}$ acceptant les mots sur l'alphabet $a, b, c\}$ commençant par a et se terminant par bc.
2. Décrire en français le langage accepté par l'automate $\mathcal{B}$ ci-dessous.

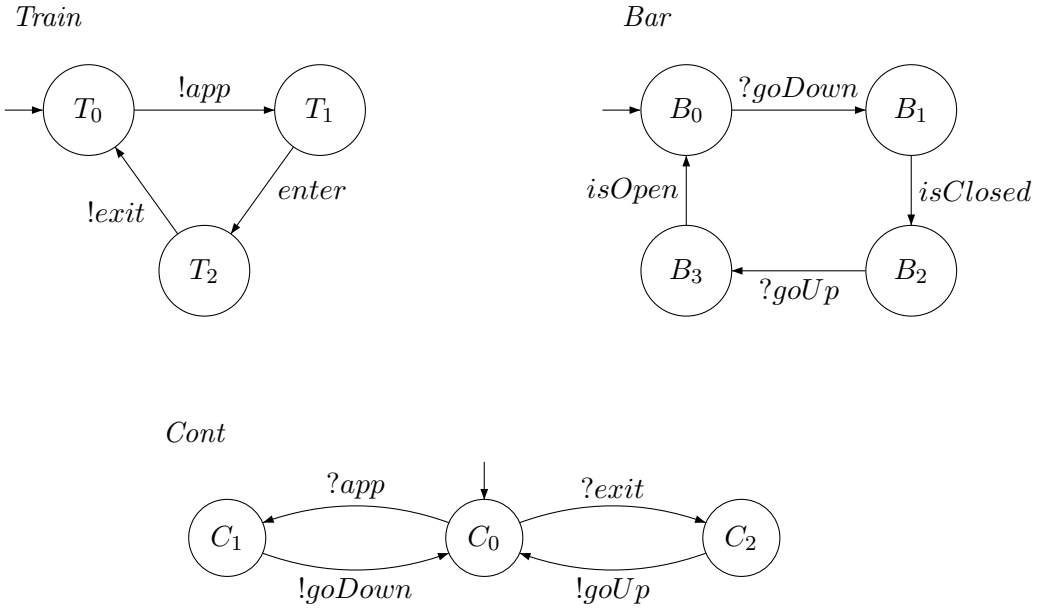


3. Dessiner un automate acceptant l'intersection des langages $\mathcal{L}(\mathcal{A})$ et $\mathcal{L}(\mathcal{B})$.

Exercice 2

Afin de concevoir un modèle (très simplifié) pour un passage à niveau, on considère trois processus : un train, une barrière et un contrôleur. Au départ, le train est loin du passage à niveau. L'action *app* représente l'approche du train, qui entre ensuite dans la zone du passage à niveau (action *enter*). Enfin, le train s'éloigne (action *exit*)... pour revenir plus tard. Le rôle du contrôleur est de fermer la barrière à l'approche d'un train, de façon à interdire aux voitures de traverser la voie ferrée pendant que le train passe. Quand le train est passé, le contrôleur rouvre la barrière.

Le train, le contrôleur et la barrière sont définis par les automates suivants :



Le système global est défini par $\mathcal{S} = (Train \parallel Cont \parallel Bar)_f$, avec la fonction de synchronisation f donnée par la table suivante :

$Train$	$Cont$	Bar	$produit$
$!app$	$?app$	$-$	ε
$enter$	$-$	$-$	$enter$
$!exit$	$?exit$	$-$	ε
$-$	$!goDown$	$?goDown$	ε
$-$	$!goUp$	$?goUp$	ε
$-$	$-$	$isClosed$	$isClosed$
$-$	$-$	$isOpen$	$isOpen$

1. Représenter l'automate fini $\mathcal{S}$ (prévoir de la place).
2. Construire un automate déterministe équivalent (pour les langages acceptés).
3. Le système vérifie-t-il la propriété de sûreté : *lorsque que le train est dans le passage à niveau, la barrière est fermée* ?

Exercice 3 - Modélisation

Le terrain sur lequel est située la gare est étroit, et celle-ci est donc traversée par une section à voie unique, trop courte pour pouvoir contenir plus d'un train. Il y a un aiguillage à chaque extrémité de la section et, en plus du passage à niveau, le contrôleur doit donc gérer ces aiguillages (voir Figure 1).

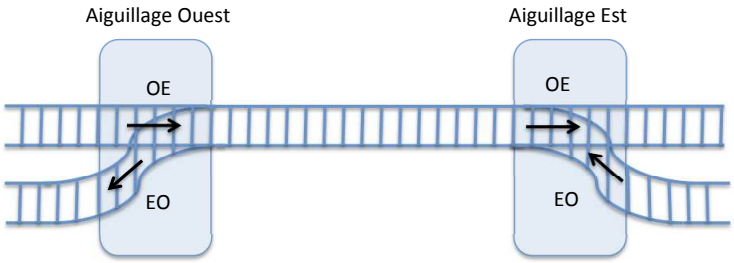


FIGURE 1 – Section de voie contrôlée par des aiguillages

Chaque aiguillage a deux états (OE, EO) et le contrôleur ne peut basculer un aiguillage (modifier son état) que lorsque la section à voie unique est vide.

Question 1 Justifiez cette condition et construisez un réseau de Petri modélisant les états de la voie et les transitions entre ces états (NB : on s'intéresse ici *uniquement* à la voie).

Les états des deux aiguillages doivent être toujours identiques, le contrôleur doit donc les basculer simultanément. Lorsqu'il a effectué cette opération, il sait qu'il a toujours le temps d'aller prendre une bière au Café de la Gare avant de remettre les aiguillages dans leur position précédente.

Question 2 Construisez un réseau de Petri modélisant le comportement du contrôleur et les états des aiguillages qui en découlent (NB : ici, on *ne s'intéresse pas* à la voie).

Question 3 Comment doit-on connecter le modèle de la Question 1 et le modèle de la Question 2 pour garantir que le contrôleur ne bascule les aiguillages que quand la voie est vide ?

Question 4 Proposez une représentation de ce système avec un réseau de Petri coloré, en précisant les ensembles de couleurs que vous utilisez et le marquage initial du réseau.

La direction régionale avertit le contrôleur qu'avec les nouveaux plannings, il passe toujours deux trains dans le sens Est/Ouest, puis trois trains dans le sens Ouest/Est, mais le contrôleur pourra toujours aller boire une bière entre deux opérations sur les aiguillages.

Question 5 Proposez un réseau de Petri modélisant le nouveau comportement du contrôleur (uniquement).

Exercice 4 - Analyse

NB : Chaque réponse doit être justifiée, vous devrez soigner vos justifications.

On considère le réseau de Petri de la Figure 2.

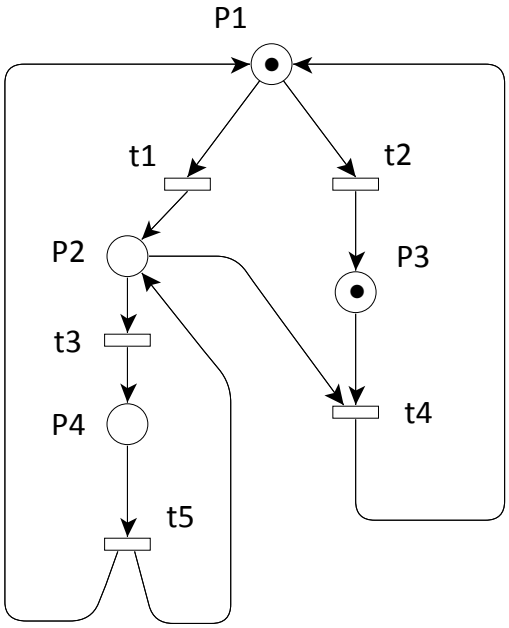


FIGURE 2 – Modèle de réseau de Petri

Question 1 Construisez le graphe de couverture de ce modèle.

Il n'est pas néceesaire d'avoir construit l'intégralité du graphe pour répondre à certaines des questions suivantes.

Question 2 Ce réseau est-il quasi-vivant ?

Question 3 Ce réseau est-il pseudo-vivant ?

Question 4 Ce réseau admet-il un état d'accueil ?