

Outils formels de Modélisation

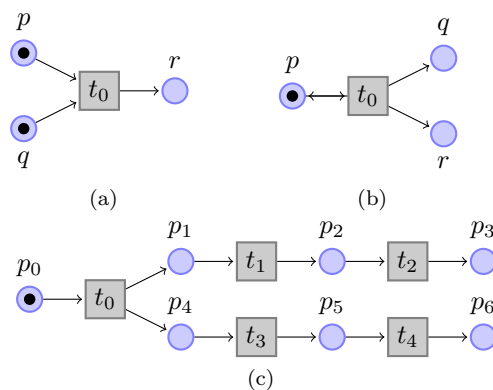
2^{ème} séance d'exercices

Aurélien Coet, Dimitri Racordon

Dans cette séance d'exercices, nous allons aborder de manière intuitive les réseaux de Petri, par le biais de l'observation et de la création de modèles simples.

1 Transformations 101 (★)

Récrivez les réseaux de Petri suivants en automates à états finis, tout en imitant au mieux leurs comportements respectifs.



Est-il possible de réécrire n'importe quel réseau de Petri en automate à états finis? Justifiez votre réponse.

2 Non déterminisme (★★)

Considérez le réseau de Petri de la figure 2.1.

1. De quel type d'algorithme ce réseau représente-t-il le comportement?
2. A partir du marquage initial, existe-t-il un marquage pour lequel à la fois t_1 et t_2 sont tirables?
3. Indiquez un état possible du réseau après 50 tirs de transitions?
4. Après 50 tirs de transitions, combien de fois t_0 aura-t-elle été tirable?

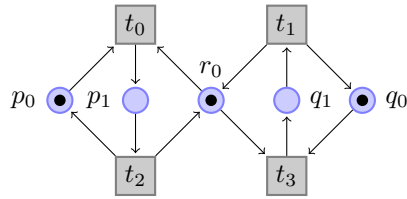


Figure 2.1: Un réseau de Petri non déterministe

3 Circulation [🚗] (★★★)

Le réseau de la figure 3.1 est une ébauche représentant un feu de circulation, dont le comportement est défini par la séquence $(r) \rightarrow (g) \rightarrow (y) \rightarrow (r) \dots$. Complétez l'ébauche afin de relever le nombre de fois que le feu *pass*e au orange.

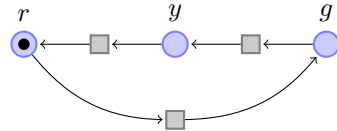


Figure 3.1: Un feu de circulation

Un feu de circulation suisse ne suit pas la séquence décrite ci-dessus, mais plutôt $(r) \rightarrow (r) \wedge (y) \rightarrow (g) \rightarrow (y) \rightarrow (r) \dots$. Modifiez le réseau afin de garantir que seul ce comportement peut être représenté par le réseau.

Vous trouverez dans le dossier **Exercice/** de ce dépôt un projet F# contenant une implémentation du modèle de feu de circulation présenté dans cet exercice. Modifiez le code dans le fichier **Program.fs** pour refléter les transformations apportées au modèle dans les points précédents.