

Outils formels de Modélisation

4^{ème} séance d'exercices

Dimitri Racordon
10.10.14

Dans cette séance d'exercices, nous allons consolider notre étude des propriétés inhérentes aux réseaux de Petri. Notamment, nous étudierons les définitions formelles de ces propriétés afin de nous familiariser avec leur notation.

1 Vous avez dit formelles ... (★★★)

Soit un réseau de Petri défini par $N = \langle P, T, *, \Delta, \Delta^*, M_0 \rangle$. Pour chacune des propriétés exprimées formellement ci-dessous, dessinez un exemple de réseau:

1. $|P| = 3 \wedge \exists t \in T \text{ tq. } \forall p \in P : \Delta^*(t, p) = 1$
2. $1 < |P| < |T| \wedge \forall p \in P : \forall t \in T : M \rightarrow^t M' \implies M'(p) > M(p)$
3. $1 < |P| < |T| \wedge \exists s \in T^* \text{ tq. } M \rightarrow^s M' \implies M' \rightarrow^s M$
4. $\forall s \in T^* : M \rightarrow^s M' \implies M \notin R(M') \wedge \nexists M' \in R(M_0) \text{ tq. } \forall p \in P : M'(p) = 0$

2 Graphe de marquage (★)

Considérez le graphe de marquage figure 2.1. Le réseau étant borné à 1, chaque marquage est représenté uniquement par les places pour lesquelles un jeton est présent. Répondez aux questions suivantes:

1. Ce réseau est-il vivant?
2. Ce réseau peut-il être réinitialisé?
3. Combien y a-t-il d'états dans ce réseau?

En admettant que la classe Swift suivante représente un noeud d'un graphe de marquage, représentez le graphe de marquage de la figure 2.1 en Swift:

```
class MarkingGraph { = un noeud du graphe de marquage
    let marking : PTMarking
    let successors : [MarkingGraph]

    init(marking: PTMarking, successors: [MarkingGraph]) {
        self.marking = marking
        self.successors = successors
    }
}
```

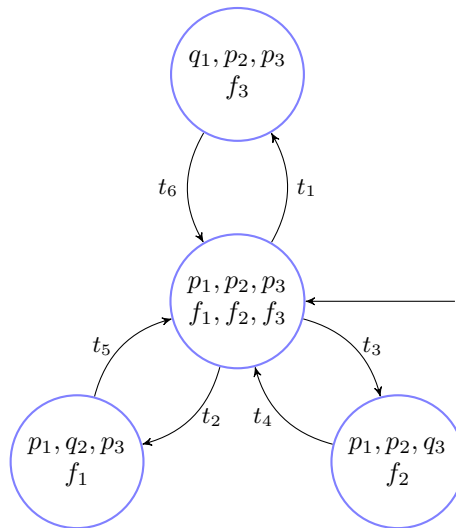


Figure 2.1: Une soirée chez Platon