TD 1 de Model Checking

16 avril 2010

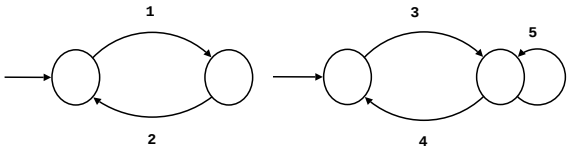
Modélisation des systèmes réactifs

Exercice 1 (Exemple de l'ascenceur.)**.** *Le système de contrôle d'un ascenceur (pour 3 étages) est défini par :*

- *le contrôleur garde en mémoire l'étage courant et l'étage cible.*
- *en mode actif, quand l'étage cible est atteint, les portes s'ouvrent et le contrôleur passe en mode attente.*
- *en mode actif, quand l'étage cible est plus élevé que l'étage courant, le contrôleur fait s'élever l'ascenceur.*
- *en mode actif, quand l'étage cible est moins élevé que l'étage courant, le contrôleur fait descendre l'ascenceur.*
- *en mode attente, il se peut que quelqu'un entre dans l'ascenceur et choisisse un nouvel étage cible. L'ascenceur ferme alors les portes et redevient actif.*
- *initialement, l'ascenceur est à l'étage 0 et en mode attente.*

Questions : 1. Proposez une machine à états modélisant le contrôle de l'ascenceur (définition formelle et dessin). 2. Définissez et dessinez le système de transitions correspondant (en vous limitant aux configurations accessibles depuis l'état initial). 3. Est-ce que les portes peuvent s'ouvrir quand l'ascenceur est actif ?

Exercice 2 (Concurrence)**.** *Soit les machines concurrentes suivantes.*



Quelles sont les transitions du système concurrent dans les cas suivants :

1. *sémantique synchrone + synchronisation entre 1 et 3 ;*
2. *sémantique asynchrone + synchronisation entre 1 et 3 ;*

Quel est le lien entre vecteur de synchronisations et rendez-vous ?

Exercice 3 (Sûreté)**.** *On se donne un système de transitions S dont certaines transitions sont distinguées et correspondent à des opérations d'aquisition de verrou (`lock`), de rendu de verrou (`unlock`), de lecture (`read`) et d'écriture (`write`). On se donne la propriété φ suivante :*

Si on ne regarde que les `lock` et `unlock` : `unlock` est toujours précédé directement de `lock`, ET une suite arbitraire de `read`,`write` est toujours précédée directement d'un `lock` .

Questions :

1. *Est-ce que φ est une propriété de sûreté ? Sinon modifiez là en conséquence.*
2. *Écrivez un automate observeur pour vérifer φ (ou sa modification) et expliquez la nouvelle propriété à vérifier.*

3. *On considère maintenant la spécification :*

   Si on ne regarde que les `lock` et `unlock` : `unlock` est toujours précédé directement de `lock`, ET une suite arbitraire de `read`,`write` est toujours précédée directement d'un `lock` et fermée directement par un `unlock`.

   Est-ce toujours une propriété de sûreté ?

Exercice 4 (Co-accessibilité)**.** *Nous avons vu comment vérifier l'invariance et l'accessibilité à partir du calcul des états accessibles ("calcul en avant"). On peut aussi vérifier ces propriétés en calculant "en arrière". On définit informellement $pre(q)$ comme les configurations à partir desquelles on peut atteindre q_0.*

- *définir formellement la relation pre (s'inspirer de post)*
- *On appelle ensemble de co-accessibilité de q l'ensemble $pre^*(q)$. Que représente-t-il intuitivement ?*
- *Donner un algorithme pour calculer $pre^*(q)$. Justifiez la terminaison.*
- *Comment vérifier l'invariance et l'accessibilité à partir de la co-accessibilité ?*

Exercice 5. *Pourriez-vous modéliser sous forme de machine à états : les automates finis, les automates à pile, les machines de Turing, un programme impératif écrit en C ? Dans le cas des automates finis, qu'y a-t-il de remarquable entre la machine à état et le système de transitions ?*

Exercice 6 (**)**.** *Comment calculer $post^*_F$ et pre^*_F dans le cas d'un système de transitions avec hypothèses d'équité sur le modèle ?* (voir poly pour les définitions).

Exercice 7. *Soit un système de transitions $S = \langle Q, T, \rightarrow \rangle$ et $q_0 \in Q$ une configuration initiale.*
Montrez que :

1. *l'ensemble d'accessibilité $post^*(q_0)$ est le plus petit invariant de S contenant q_0 ;*
2. *il existe $k \in \mathbb{N}$ tel que $post^*(q_0) = \bigcup_0^k post^i(q_0)$.*