

Examen ACCOV

Sur le problème des philosophes

J.F. Peyre

Mercredi 29 Juin 2005

Partie modélisation et analyse

Soit le réseau coloré suivant (figure 1) modélisant une version du problème des philosophes.

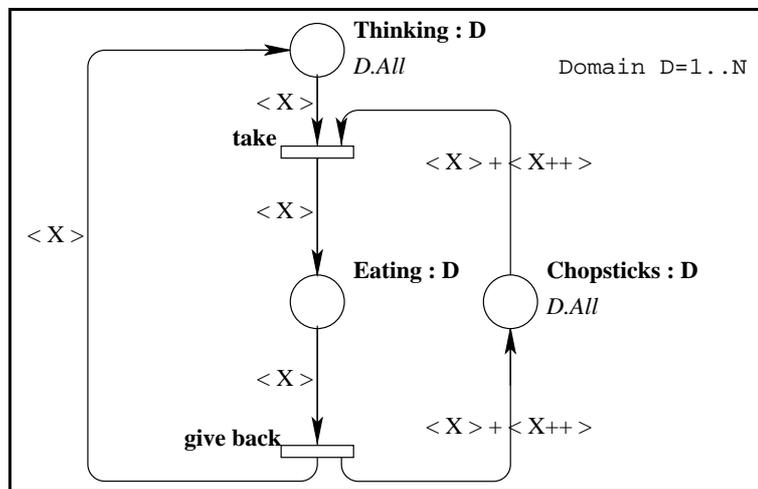


Figure 1: Une version du problème des philosophes

Question 1 : (2 pts) Montrez qu'il ne peut y avoir d'interblocage dans ce modèle.

Question 2 : (2 pts) Montrez que par ailleurs, il y a un problème potentiel de famine.

On souhaite supprimer le problème de famine et on construit le modèle suivant (figure 2). Cette nouvelle version diffère du modèle précédent par l'ajout des deux états intermédiaires d'attente (modélisés par les places `Waiting` et `InQueue`) et par l'ajout d'une ressource supplémentaire (modélisée par la place `NotBlocked`).

Le principe de fonctionnement est le suivant : quand un philosophe x veut manger il va dans l'état `Waiting`. Là, si les fourchettes x et $x + 1$ sont disponibles **et si** la place `NotBlocked` contient le jeton x alors il franchit `Go1` et mange. Sinon, il franchit `requeue` et va en attente dans l'état `InQueue` après avoir pris le jeton $x + 1$ dans la place `NotBlocked` (bloquant ainsi une future demande du philosophe $x + 1$).

On décide que la transition `Go1` est prioritaire sur la transition `requeue` (mais pas sur les autres transitions) ce qui signifie que lorsqu'on peut franchir à la fois `Go1` et `requeue` on franchit `Go1`.

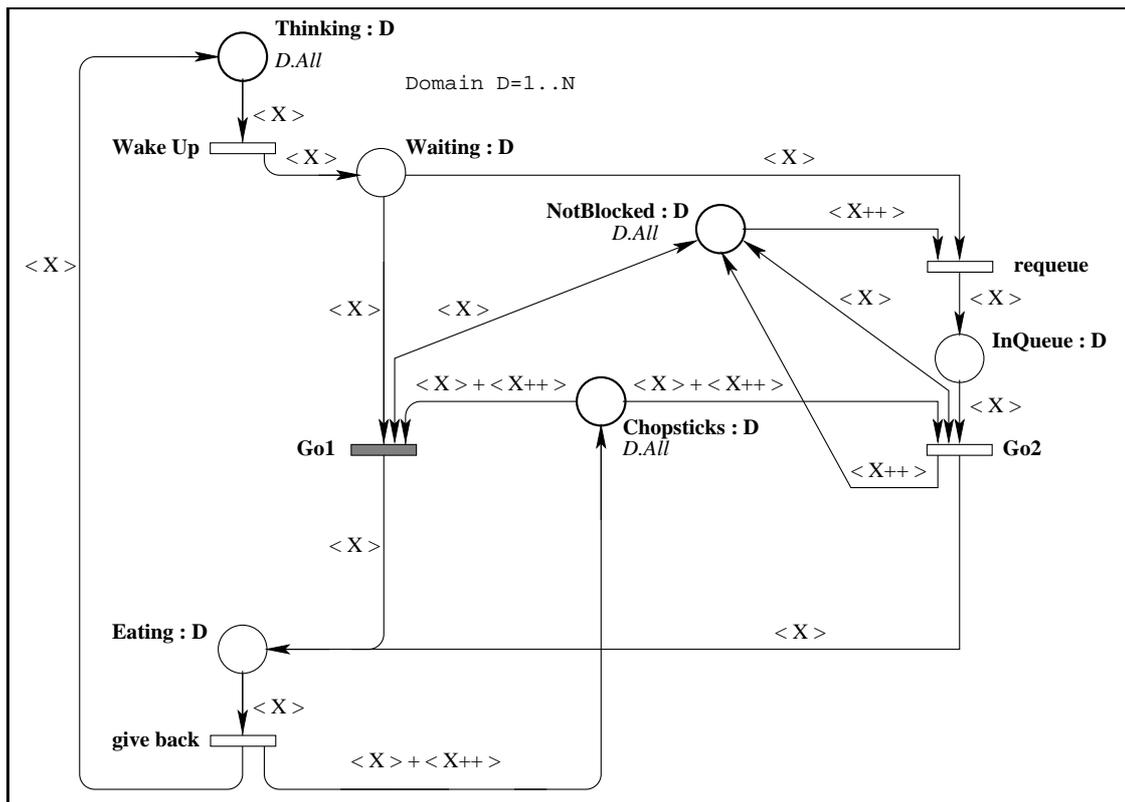


Figure 2: Une version du problème des philosophes

Question 3 : (2 pts) Montrez qu'il existe maintenant une possibilité d'interblocage.

On modifie alors la priorité de la transition `Go2` de telle sorte qu'elle soit plus prioritaire que `Wake Up` (mais pas sur les autres transitions). Cela signifie que lorsqu'on peut franchir à la fois `Go2` et `Wake Up` on franchit `Go2`.

Question 4 : (3 pts) Montrez que maintenant la place la place `NotBlocked` ne peut devenir vide ; vous pourrez pour cela supposer qu'elle ne contient plus qu'une seule marque et étudier les franchissements possibles à partir de ce marquage. Que pouvez en déduire sur la possibilité d'interblocage ? Justifiez brièvement votre réponse.

Question 5 : (2 pts) Y a-t-il un problème de famine dans ce modèle ? Justifiez brièvement votre réponse.

Partie implémentation

On décide d'implémenter cette dernière solution en Ada avec un gérant des fourchettes qui est un objet protégé. Il y a une entrée publique qui correspond à la transition `Wake Up` et une entrée privée qui correspond à la transition `requeue`.

Question 6 : (1 pts) Quelle propriété associée aux objets protégés permet d'assurer que le traitement des tâches en attente sur l'entrée privée est prioritaire sur le traitement des tâches appelant l'entrée publique (et que donc il n'y aura pas d'interblocage) ?

Question 7 : (6 pts) Implémenter cette solution en Ada selon le cadre qui vient d'être défini.

On essaye maintenant d'implémenter une solution en Promela en adoptant la structure suivante: les fourchettes sont gérées par un processus. Les demandes viennent sur une file de messages et si les fourchettes ne sont pas disponibles, un message est déposé sur une seconde file.

Lorsque les fourchettes sont attribuées un message est envoyé en retour au philosophe.

Le relâchement des fourchettes se fait par un envoi sur une troisième file.

Le serveur (de comportement cyclique) examine les trois files et, par un mécanisme à proposer, traite en priorité les messages de la troisième file (relâchement des fourchettes) puis ceux de la seconde file (attente interne) et enfin ceux de la première.

Question 8 : (1+3 pts) Proposer un schéma décrivant cette architecture, puis proposer une solution en Promela.