

Modélisation et analyse des SED modélisés par des réseaux de Petri colorés (RdPC)¹

Objectif : L'objectif de ce TP concerne le développement d'un modèle RdPC d'un système à événements discrets dans le but d'analyser son comportement en vérifiant ses propriétés.

Mots clés :

- RdP avec jetons non colorés et colorés
- Simulation pas à pas
- Graphe de marquages, Vérification des propriétés
- Hiérarchie

Ouvrons une connexion !

Considérons deux processus (applications) qui sont exécutés sur deux ordinateurs distants connectés par un réseau. Les deux processus ont besoin de communiquer, mais pour que cette communication devienne possible sans collisions sur le réseau (médium de communication partagé), une étape préalable d'ouverture de connexion est nécessaire. L'établissement d'une connexion entre les deux processus permet également de réserver les ressources nécessaires à l'échange des messages contenant de l'information utile pour les deux processus.

Considérons le protocole suivant de gestion de connexion : un premier processus (P_1 ou P_2) peut initialiser l'ouverture de la connexion en envoyant un message de demande d'ouverture vers le deuxième processus et, ensuite ce premier processus ouvre la connexion de son côté. A la réception du message de demande d'ouverture, le deuxième processus ouvre la connexion de son côté et, ensuite, l'échange des messages utiles entre les deux processus peut démarrer. Lorsqu'il n'y a plus des messages utiles à échanger, la connexion doit être fermée, afin de pouvoir libérer les ressources utilisées pour la communication. Dans ce but, le processus P_1 envoie une demande de fermeture de connexion au processus P_2 et il ferme la connexion de son côté. A la réception du message de demande de fermeture, le processus P_2 ferme également la connexion.

1. Modéliser ce protocole de gestion de connexion par un réseau de Petri (l'échange des messages contenant de l'information utile ne sera pas modélisé, mais seulement l'ouverture et la fermeture de connexion). Simuler le réseau de Petri. Que constatez-vous ? Est-ce que ce protocole de gestion de connexion peut être satisfaisant ?

Une autre solution possible de gestion de la connexion consiste à mettre en place des acquittements pour les messages d'ouverture et de fermeture de connexion, acquittements qui permettront au processus qui est à l'initiative d'une demande d'être informé de la réception de sa demande par son correspondant et seulement ensuite la connexion sera ouverte. Ce protocole fonctionne de la manière suivante : un premier processus (P_1 ou P_2) peut initialiser l'ouverture de la connexion en envoyant un message de demande d'ouverture vers le deuxième processus. A la réception de ce message le deuxième processus ouvre la connexion tout en envoyant un acquittement vers le premier processus. A la réception de l'acquittement le premier processus ouvre à son tour la connexion. La connexion reste ouverte tant qu'une demande de fermeture n'est pas initialisée. Pour simplifier, dans un premier temps, considérons que seulement le premier processus possède le droit de fermer la connexion ouverte en envoyant un message de demande de fermeture. A la réception de ce message le processus P_2 ferme la connexion pour sa part, tout en envoyant un acquittement vers P_1 . Lorsqu'il reçoit l'acquittement, P_1 ferme également la connexion de sa part.

Ce protocole sera considéré pour la suite de cet exercice.

2. Modéliser ce protocole de gestion de connexion par un réseau de Petri. Vérifier les propriétés du modèle et par conséquent le fonctionnement du protocole.
3. Afin de simplifier ce modèle et de le rendre générique, proposer un modèle réseau de Petri hiérarchique. On considère maintenant que tous les deux processus peuvent initialiser la fermeture de la connexion. Vérifier les propriétés du modèle et par conséquent le fonctionnement du protocole.

¹ Vos remarques et commentaires à propos de ce sujet sont les bienvenus.

Contact : Nicolae.Brinzei@univ-lorraine.fr ; Jean-Francois.Petin@univ-lorraine.fr, Gael.Hequet@univ-lorraine.fr.

4. Proposer une amélioration du protocole afin d'obtenir un fonctionnement correct. Modifier le réseau de Petri pour refléter cette amélioration. Vérifier à nouveau les propriétés du modèle et par conséquent le fonctionnement du protocole modifié.

Compteurs électriques intelligents ! (version simplifiée)

Cet exercice propose une extension de l'exercice précédent. On ajoute une troisième entité de type processus et le protocole de communication est du type maître-esclave. Ainsi, le processus P_1 est celui qui initialise toujours la connexion avec les processus P_2 et P_3 : il communique dans un premier temps avec P_2 et ultérieurement avec P_3 et ainsi de suite.

1. Modifiez le réseau de Petri de l'exercice précédente pour intégrer un troisième processus. On considère à nouveau juste les phases d'ouverture et fermeture de connexion, sans les échanges des messages utiles.
 - I. Valider le bon comportement de ce système en vérifiant ses propriétés. Donner la signification des propriétés vérifiées et justifier votre choix pour ces propriétés.
 - II. Si le comportement du système n'est pas celui que vous attendez, apporter les modifications nécessaires pour corriger le(s) problème(s) détecté(s).
2. *Question bonus : Une fois la connexion ouverte, les processus P_2 et, respectivement, P_3 envoient un message au processus P_1 . Une fois que P_1 reçoit le message, il initialise la fermeture de la connexion. Refaire les points I et II ci-dessus.*

Compteurs électriques intelligents !

Dans le cadre de la « ville intelligente »², on considère un système de relevé de la consommation d'énergie électrique à distance³. Les compteurs électriques intelligents sont connectés à un serveur du gestionnaire du réseau de distribution d'électricité. Ce serveur exécute deux tâches (exécutés par deux applications différentes)⁴ :

1. relevé de la consommation d'électricité ;
2. surveillance de l'état de bon fonctionnement des compteurs distants.

Toutes les communications entre le serveur et les compteurs et vice-versa s'effectuent après une ouverture préalable d'une connexion.

1. Le relevé de la consommation d'électricité s'effectue de la manière suivante :
 - le serveur interroge un premier compteur
 - le serveur ouvre une connexion avec un compteur
 - une fois la connexion ouverte, le serveur envoie une requête au compteur pour lui demander les mesures suivantes : la consommation courante, ainsi que la consommation depuis le début de la journée
 - le compteur répond en transmettant ses mesures
 - le serveur ferme la connexion avec le compteur
 - le serveur interroge ensuite, de manière itérative, les autres compteurs connectés au réseau en suivant le même protocole de fonctionnement décrit précédemment.
2. Pour informer sur son état de fonctionnement, un compteur :
 - ouvre d'abord une connexion avec le serveur
 - transmet au serveur un message en l'informant de son état ; le serveur répond par un acquittement signifiant la bonne réception du message ; cet acquittement vaut également une demande de fermeture de la connexion

² Smart city.

³ Par exemple les compteurs Linky qui représentent la nouvelle génération des compteurs communicants d'Enedis. Enedis est le gestionnaire du réseau public de distribution d'électricité.

⁴ On considère ici un fonctionnement simplifié du système.

- ferme la connexion.

Afin de gérer le bon fonctionnement de ces échanges, c'est un premier compteur qui détient initialement le droit d'ouverture d'une connexion afin que celui-ci puisse informer le serveur sur son état. Ensuite, ce premier compteur cède à son tour ce droit au compteur suivant et ainsi de suite. Le dernier compteur transmet ce droit au serveur qui, connaissant les compteurs en état de fonctionnement, commencera à interroger les compteurs un à un afin de réaliser le relevé de la consommation d'électricité. Une fois que le relevé de tous les compteurs est réalisé, le processus peut recommencer.

On considère que le système de relève automatique est constitué d'un serveur⁵ et de deux compteurs.

Modéliser ce système par RdPC.

- III. Valider le bon comportement de ce système en vérifiant ses propriétés. Donner la signification des propriétés vérifiées et justifier votre choix pour ces propriétés.
- IV. Si le comportement du système n'est pas celui que vous attendez, apporter les modifications nécessaires pour corriger le(s) problème(s) détecté(s).

⁵ Dans un système réel, plusieurs serveurs peuvent exister : soit pour assurer la redondance (si un premier serveur tombe en panne, un autre serveur peut continuer à assurer le bon fonctionnement du système), soit pour se partager l'ensemble des compteurs connectés, chaque serveur prenant en charge un sous-ensemble de compteurs connectés.