

Les réseaux de Petri : Corrigé de TD1

(Source de TD1 et ses corrigés : M.-C. Portmann)

Exercice 1 :

Le réseau de Petri ci-dessous représente l'activité d'un barman.

C'est un graphe d'événements.

Pour passer à deux barmans qui ont le même temps de service, il suffit d'avoir mis au départ deux jetons dans le deuxième circuit qui correspond à l'activité ou à la non activité des barmans. On ne sait pas quel est le barman qui s'active lorsque les deux sont inactifs et cela n'a pas d'importance.

Bien sûr, on peut généraliser à p barmans ayant le même temps de service. Et ce sera toujours un graphe d'événements.

Si les deux barmans n'ont pas le même temps de service, on peut créer un circuit pour chacun des barmans et dans ce cas, on sait quel est le barman qui est inactif lorsque un seul l'est.

Si on cherchait un fonctionnement au plus tôt de ce réseau de Petri, il faudrait décider par exemple que lorsque l'un des deux barmans sont disponibles, l'un est toujours plus rapide que l'autre pour s'adresser au client et prendre sa commande, car ici, il faut lever le conflit, le graphe n'est plus un graphe d'événement.

On peut bien sûr généraliser au cas de p barmans ayant des temps de service différents.

Exercice 2 :

1. La figure de la page suivante montre le graphe d'événements correspondant.

Chaque produit y est représenté par autant de circuits élémentaires que sa proportion dans l'équilibrage des produits, donc ici 2 circuits élémentaires pour le produit P1 et un seul circuit élémentaire pour les deux autres produits. Nous appelons ces circuits les ***circuits "produits"*** ou ***circuits de process***.

Chaque fois que l'on termine un produit fini P_i , une pièce brute de produit P_i est candidate à une entrée dans l'atelier, grâce à la place qui suit la transition correspondant à la dernière opération de la gamme.

Si on n'avait pas mis la première transition du circuit produit, mais si l'on avait mis comme arrivée de l'arc qui sort de la place "une pièce brute P_i veut entrer" directement la transition correspondant à la première opération de la gamme, alors, la nouvelle pièce serait traitée tout de suite et, la progression des différents produits étant indépendante (sous l'hypothèse d'avoir suffisamment de machines identiques de chaque type), les produits dont la somme des durées opératoires est la plus courte auraient un plus court cycle de fabrication et seraient donc fabriqués en plus grande quantité que les autres à égalité du nombre de circuits élémentaires les concernant.

Le circuit le plus à gauche qui passe par la première transition de chacun des autres circuits est un circuit régulateur qui permet d'assurer que les entrées (et par conséquent les sorties) respectent les proportions demandées. Si ce circuit ne contient qu'un seul jeton, il impose ici de rentrer d'abord une pièce brute P1, puis une autre pièce brute P1, puis une pièce brute P2, puis une pièce brute P3, puis de recommencer ... Il est évident que si on échange la position des circuits, on pourrait imposer d'entrer une pièce P1, puis une pièce P2, puis une pièce P1, puis une pièce P3 avant de recommencer ...

On fige donc la succession des entrées répétitives des pièces brutes dans l'atelier en construisant ce ***circuit de contrôle des entrées*** (dans une phase d'optimisation combinatoire, on pourra donc envisager les différentes façons de construire ce circuit, pour le moment, on suppose que l'on a choisi le circuit de ce schéma de manière arbitraire).

2. On ajoute autant de ***circuits de commande*** que de machines qui vont obliger que les produits se succèdent (strictement ou non) sur les machines sans recouvrement.

Comme pour le cas du circuit de contrôle des entrées, on "brise la combinatoire du problème" en choisissant l'ordre de succession des produits sur chacune des machines (on pourra toujours a posteriori inclure ce travail dans un module exploratoire qui testera si changer cet ordre peut améliorer le critère considéré).

Nous avons choisi ici de prendre l'ordre P1 (premier exemplaire), P1 (deuxième exemplaire) et enfin P2 sur la machine 1 ; l'ordre P1 (premier exemplaire), P1 (deuxième exemplaire) puis P2 et enfin P3 sur la machine 2 ; l'ordre P1 (premier exemplaire), P1 (deuxième exemplaire) puis P3 sur la machine 3. Ce qui nous donne le schéma complété ci-dessus.

Alors que l'on pourrait envisager de mettre plusieurs jetons dans le circuit de contrôle des entrées (ce qui signifierait que l'on entre en parallèle plusieurs fois un sous-ensemble $P1, P1, P2, P3$), par contre, il faut mettre dans chaque circuit de commande machine un jeton et un seul, c'est lui qui assure que la machine ne travaille pas sur plusieurs produits à la fois ; et si l'entrée est, par exemple, une phase de palétisation de la pièce brute sur une machine unique, il faut un seul jeton dans le circuit de contrôle des entrées, ce que nous retiendrons ici.

Ayant choisi les circuits de commande et de contrôle et sachant que l'on va déclencher toutes les transitions tirables au plus tôt, le seul degré de liberté restant est le marquage initial.

Si l'on met des jetons dans la gamme entre deux machines, cela signifie que l'on ne démarre pas à vide et qu'il existait des encours dans l'atelier. Si l'on démarre à vide, tous les encours sont des pièces brutes et on en place autant qu'on le souhaite, mais au moins un, dans la place "une pièce brute est candidate à une entrée", les autres places des circuits pièces restant vides.

On place le jeton du circuit de contrôle des entrées où bon nous semble et le jeton de contrôle des machines de préférence devant une première opération d'une gamme (car il faut être sûr que le processus démarre et que toutes les transitions seront accessibles à partir du marquage initial).

Après ces derniers choix, nous avons parfaitement défini un ordonnancement répétitif au plus tôt.

Exercice 2 :

1. On considère tout d'abord l'atelier mono-produit qui fabrique le seul produit P et le schéma proposé dans l'énoncé.

Remarques sur le schéma :

Les boucles sur les places "A1 inactif" et "A2 inactif" permettent de ne lancer la fabrication que d'un seul container à la fois pour chaque chaîne, alors qu'il n'est pas nécessaire d'avoir une telle boucle pour les fournisseurs, puisque plusieurs containers peuvent être acheminés en parallèle. La boucle du magasin permet de représenter le passage des containers chez les clients. Ils partent pleins et reviennent vides en aval des chaînes (c'est-à-dire sur la place en amont des transitions chaînes qui ne peuvent être tirées que s'il y a des containers vides et s'il y a des composants en quantité suffisante). La valeur μ permet de modéliser une rotation plus ou moins rapide des containers chez les clients.

2. Ce graphe n'est pas un graphe d'événement, car les places correspondants aux containers de composants pleins et aux containers de produits vides ont deux transitions en sortie (les deux chaînes d'assemblage), cela représente le fait que le problème n'est pas déterministe, et si les deux chaînes sont arrêtées et peuvent démarrer, il faudra de manière arbitraire traiter l'une avant l'autre (par exemple en donnant toujours la priorité à la chaîne la plus rapide).
3. Le passage à deux produits a nécessité ici une multiplication des transitions correspondant aux chaînes. Une transition correspondant maintenant à l'exécution d'un produit sur une chaîne. Cette duplication a été rendu nécessaire de manière à distinguer les containers vides qui seront remplis et les quantités différentes de composants qui seront utilisées

selon les produits (on aurait pu ajouter un composant C3 pour le produit Q avec un des fournisseurs existants ou un nouveau fournisseur, ne pas utiliser le composant C1 ou le composant C2 pour le produit Q, introduire autant de produits et de composants que l'on souhaite, cela ne complique pas la modélisation, le modèle à deux produits, deux chaînes et deux fournisseurs est suffisant pour mettre en évidence toutes les difficultés, seule la lisibilité du dessin souffre de la complexité croissante de l'exemple.