

TD 1

1. Répartition de processus parallèles dans un système biprocesseur

Plusieurs tâches sont soumises au cours du temps à un système multiprocesseurs comprenant deux processeurs, P_1 et P_2 , travaillant en parallèle.

Si le processeur choisi pour affecter une nouvelle tâche est occupé, on place la tâche dans une file d'attente associée à ce processeur (la file d'attente des processus prêts). Les files d'attente sont ordonnancées selon le mécanisme FIFO.

Les processeurs sont spécialisés et différents ; le temps d'exécution des tâches affectées à P_1 s'avère alterner entre 3 et 1, en commençant par 3, tandis que celui des tâches affectées à P_2 est toujours égal à 2.

Soit $x_1(t)$ et $x_2(t)$ les longueurs des files d'attente pour les processeurs P_1 et P_2 , respectivement (incluant une tâche en cours d'exécution¹), et $N(t)$ le nombre total de clients ayant quitté le système au temps t . Si un ou plusieurs événements se produisent à l'instant t , les valeurs de ces trois variables sont celles obtenues *juste après* que se soit produit l'ensemble de ces événements.

Le choix du processeur auquel affecter une nouvelle tâche est effectué selon la **Règle 1** : Envoyer la tâche sur P_1 si la longueur de la file d'attente pour P_1 demeure inférieure ou égale à 2, sinon envoyer la tâche sur P_2 . Les files d'attente sont vides à $t = 0$.

a. Les tâches, supposées indépendantes, sont soumises aux instants $t = 0, 1, 2, 4, 6, 7, 10$. Dessiner le chronogramme de $x_1(t)$, $x_2(t)$ et $N(t)$ pour t dans l'intervalle $[0, 10]$, montrant les arrivées et les départs des tâches (supposer que juste avant l'instant $t = 0$, $x_1(0_-) = x_2(0_-) = N(0_-) = 0$). Si une fin d'exécution se produit au même instant qu'une soumission, supposer qu'elle est prise en compte avant la soumission.

b. On suppose maintenant que les tâches arrivent aux instants 0,1, 0,7, 2,2, 3,2 et 3,9. Le temps d'exécution alterne maintenant entre 3,2 et 1,1 sur le processeur P_1 tandis qu'il est constant, égal à 2,0 unités, sur P_2 . On considère un modèle actionné par les événements, avec pour ensemble d'événements $E = \{a, d_1, d_2\}$, où a = arrivée et d_i = départ du processeur P_i , $i = 1, 2$. Dessiner le chronogramme de $x_1(t)$, $x_2(t)$ et $N(t)$ pour t dans l'intervalle $[0, 10]$. Si deux événements se produisent au même instant, supposer qu'un départ se produit toujours avant une arrivée.

2. Variante pour la répartition de processus parallèles

Répéter le problème 1 avec l'algorithme d'ordonnancement suivant - où les files sont toujours gérées en FIFO - qui diffère sur la façon de choisir le processeur à allouer à chaque nouvelle tâche (si deux événements se produisent au même instant, on suppose toujours qu'un départ se produit avant une arrivée) :

Règle 2 : envoyer la tâche sur le processeur dont la file d'attente est la plus courte. En cas d'égalité, l'envoyer sur P_2 .

3. Modélisation par automates d'état

Modéliser les deux systèmes biprocesseurs (règle 1 et règle 2) à l'aide d'automates dont le nombre d'états est infini dénombrable.

1. Dans tout le TD, les longueurs prennent en compte le processeur, cf. Chapitre 1, page 22.