

### TD 4 Répartition de charge

#### 4.1 Scheduling statique et dynamique

Le problème considéré est celui de l'allocation de  $N$  tâches comparables, mais pas nécessairement identiques, sur  $P$  processeurs. La difficulté est de trouver un compromis entre allocation complètement dynamique, qui a un surcoût de gestion trop élevé, et allocation complètement statique. La classe de méthodes de *chunking* alloue des ensembles de taille décroissante. Par exemple, GSS (guided self-scheduling) alloue à la requête  $i$  un nombre de tâche égal à  $\lceil R_i/P \rceil$ , où  $R_i$  est ce qui reste.

1. Pour  $N = 256$  et  $P = 4$ , avec toutes les tâches identiques de durée d'exécution 1, donner le détail de l'ordonnancement sur chaque processeur.
2. On considère maintenant un ensemble formé de 32 tâches de durée 1, 32 tâches de durée 4, et 48 tâches de durée 2. Décrire le meilleur et le pire cas pour l'allocation complètement statique et pour GSS.

#### 4.2 Méthode du pivot de Gauss

Une version très simplifiée de la méthode du pivot de Gauss est :

```
do i = 1,n
do k = i+1,n
do j = i+1,n
a(k,j) = a(k,j) -a(k,i)*a(i,j)/a(i,i)
end do
end do
```

1. Quelles sont les tâches parallélisables? Dans le cas du parallélisme maximal, quelles sont les opérations parallèles mises en jeu?
2. Etudier les avantages et inconvénients d'une distribution des colonnes par blocs (par exemple sur une matrice 16x16 sur 4 processeurs).
3. Etudier les avantages et les inconvénients d'une distribution cyclique des colonnes.