Anneaux de processeurs et correction du partiel

1 Anneaux - Rotations de Givens

Pour triangulariser une matrice A d'ordre n de façon numériquement stable, on peut utiliser les rotations de Givens. L'opération de base $\textsf{Rot}(i,j,k)$ consiste à combiner les deux lignes i et j, qui doivent toutes deux commencer par $k-1$ zéros, pour annuler l'élément en position (j,k) :

$$\begin{pmatrix} 0 & \dots & 0 & \mathbf{a'_{i,k}} & a'_{i,k+1} & \dots & a'_{i,n} \\ 0 & \dots & 0 & \mathbf{0} & a'_{j,k+1} & \dots & a'_{j,n} \end{pmatrix} \leftarrow \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 0 & \dots & 0 & \mathbf{a_{i,k}} & a_{i,k+1} & \dots & a_{i,n} \\ 0 & \dots & 0 & \mathbf{a_{j,k}} & a_{j,k+1} & \dots & a_{j,n} \end{pmatrix}$$

3p Nous laissons au lecteur le soin de déterminer l'angle θ permettant d'effectuer cette opération. :-)

L'algorithme séquentiel peut s'écrire :

```
Givens(A)
  Pour k = 1 to n − 1 :
    Pour i = n downto k + 1 step −1 :
      Rot(i − 1, i, k)
```

On considère qu'une rotation $\textsf{Rot}(i,j,k)$ s'exécute en temps unité, indépendamment de k.

▷ **Question 1** *Mettre en œuvre cet algorithme sur un réseau linéaire de n processeurs.*

▷ **Question 2** *Mettre en œuvre cet algorithme sur un réseau linéaire comportant seulement $\lfloor \frac{n}{2} \rfloor$ processeurs.*

2 Facteur d'accélération

On discute diverses ■ lois ■ (Amdahl, Gustafson) sur le facteur d'accélération et l'efficacité. Ces lois font désormais partie du bagage de tout honnête paralléliseur !

▷ **Question 3** *Soit un problème à résoudre, qui comporte un pourcentage f d'opérations intrinsèquement séquentielles. Montrer que le facteur d'accélération est limité par $1/f$, quel que soit le nombre de processeurs utilisés. Quelle leçon en tirer pour la parallélisation d'un problème de taille fixe ?*

▷ **Question 4** *Pour un problème matriciel de taille $n \times n$, on suppose que :*
- *le nombre d'opérations arithmétiques à exécuter est n^α, où α est une constante ;*
- *le nombre d'éléments à stocker en mémoire est $w_1 n^2$, où w_1 est une constante ;*
- *le nombre d'opérations d'entrées-sorties (purement séquentielles) est $w_2 n^2$, où w_2 est une constante.*

Comment estimer l'accélération obtenue avec p processeurs sur un problème de grande taille ? Quelle leçon en tirer pour la parallélisation d'un problème de taille variable ?

▷ **Question 5** *Donner des exemples de facteur d'accélération superlinéaire, c'est-à-dire d'efficacité strictement supérieure à 1.*

3 Anneaux

3.1 Résolution d'un système triangulaire

On s'intéresse à la résolution du système linéaire $Ax = b$, où A est une matrice triangulaire inférieure d'ordre n et b un vecteur à n composantes. On suppose disposer d'un anneau orienté de p processeurs P_0, P_1, ..., P_{p-1}. On prendra n divisible par p.

▷ **Question 6** *On veut distribuer les colonnes de A aux processeurs. Quelle stratégie adopter ? Donner le schéma d'un algorithme de résolution parallèle adapté à la distribution choisie.*

▷ **Question 7** *On veut maintenant distribuer les lignes de A aux processeurs. Quelle stratégie adopter ? Donner le schéma d'un algorithme de résolution parallèle adapté à la distribution choisie.*

À l'étape i, le processeur responsable de $x[i]$ calcule sa valeur définitive, puis la diffuse aux autres processeurs, qui mettent à jour leurs données. Le lecteur attentif aura remarqué la parfaite similitude entre cet algorithme et celui de la décomposition LU. Toutes les variantes proposées pour cette dernière peuvent être utilisées : recouvrement calcul-communication, pipeline de la diffusion, jusqu'à la technique du *look-ahead* !