

**Examen du cours I3**  
**Programmation hybride et multicœurs :**  
**Programmation hybride MPI-OpenMP**

**Vendredi 10 février 2017 - durée 1 heure 30**  
**Supports de cours autorisés**

Le but de cet examen est d'étudier l'implémentation du produit  $Y = L X$  où  $X, Y$  sont des vecteurs de taille  $n$  et  $L$  est une matrice carrée pleine  $n \times n$ , mais où les coefficients non nuls sont en-dessous de la diagonale :

$L(i,j)$  est nul pour  $j < i$

$L(i,i)$  est non nul

$L(i,j)$  est quelconque pour  $j > i$

Pour ce type de matrice, le produit  $Y = L X$  peut s'écrire en pseudo-code :

```
pour i=1 à n
    Y(i) = 0
    pour j=1 à i
        Y(i) = Y(i) + L(i, j) * X(j)
    fin pour
fin pour
```

Remarque :

On suppose disposer de types de donnée « vecteur » et « matrice » qui proposent un accès à la  $i^{\text{ème}}$  coordonnée du vecteur et au coefficient d'indice  $(i,j)$  de la matrice.

On suppose que l'architecture de la machine cible est un cluster de SMP. La machine dispose de  $N$  nœuds, chaque nœud comportant un processeur contenant  $M$  cœurs.

**Question de cours 1 :**

Rappeler les différences principales entre la programmation OpenMP « grain fin » et « gros grain ».

**Question de cours 2 :**

Dans le modèle de programmation hybride MPI-OpenMP, décrivez les principaux avantages espérés par rapport à une programmation purement MPI et une programmation purement OpenMP.

**Exercice 1 :**

Écrire le pseudo-code d'une version OpenMP « grain fin » du produit matrice-vecteur.

On rappelle que dans cet exercice et les suivants, la syntaxe OpenMP et/ou MPI ne fera pas partie de l'évaluation

**Exercice 2 :**

Écrire le pseudo-code d'une version OpenMP « gros grain » du produit matrice-vecteur. On fera attention en particulier à la répartition de la charge de calcul entre les threads (équilibre de charge).

Comparer avec la version « grain fin ». Discuter des performances théoriques attendues.

**Exercice 3 :**

En partant de la version OpenMP « gros grain » de la question 1, écrire une version hybride MPI-OpenMP.