Ecole Polytechnique

Cours : “Parallélisme”
Travaux dirigés
E. Goubault & M. Martel

TD 8

5 mars 2002

On commencera avant tout par finir le TD précédent (au moins le routage dans l'hypercube et sa simulation en JAVA).

1 Méthode aux différences finies

On considère le code suivant (très classique dans des méthodes aux différences finies):

```
for (i=1;i<=N;i++)
  for (j=1;j<=N;j++)
    a[i][j]=1/4*(a[i-1][j]+a[i+1][j]+a[i][j-1]+a[i][j+1]);
```

- Donner le GDE pour N=5
- Donner le GDR avec le type de dépendances (flot, anti, sortie)
- Imaginer une torsion de l'espace d'itération qui permettre d'extraire une boucle parallèle

On pourra simuler la parallélisation de ce code avec les threads JAVA. Indiquer d'autres parallélisations possibles.

2 Elimination de dépendances

On considère les deux codes suivants, qui font la même chose:

```
for (i=1;i<=N;i++) {
  S1: a[i]=b[i]+c[i];
  S2: a[i+1]=a[i]+2*d[i]; }
```

```
for (i=1;i<=N;i++) {
  S1': temp[i]=b[i]+c[i];
  S1: a[i]=temp[i];
  S2: a[i+1]=temp[i]+2*d[i]; }
```

- Donner le GDR du premier code.
- Donner le GDR du deuxième code.
- Conclusion?

3 Algorithme d'Allen et Kennedy

On considère le code suivant:

```
for (i=1;i<=N;i++)
  for (j=1;j<=N;j++) {
    S1: a[i+1][j+1]=a[i+1][j]+b[i][j+2];
    S2: b[i=1][j]=a[i+1][j-1]+b[i][j-1];
    S3: a[i][j+2]=b[i+1][j+1]-1; }
```

- Donner le GDR avec le type de dépendances (flot, anti, sortie), leur expression (vecteur de dépendance) et leur niveau.
- Appliquer l'algorithme de Allen et Kennedy pour restructurer le nid de boucles