Maîtrise d'informatique
Module de Programmation parallèle

Examen

Philippe Marquet

Juin 1998

Tous documents autorisés
Durée : 3h

1 Shearsort en MPL sur la MasPar

Dans le principe, le *shearsort* étend le tri pair-impair sur une grille 2-D.

Soient $n \times n$ valeurs à trier. On considère une grille de $n \times n$ processeurs virtuels (n est une valeur entière paire). On trie alternativement les lignes et les colonnes : les lignes sont triées aux itérations impaires et les colonnes sont triées aux itérations paires. Les lignes sont triées :

- dans l'ordre croissant pour les lignes paires (de 0 à $n-2$) ;
- dans l'ordre décroissant pour les lignes impaires (de 1 à $n-1$).

Les colonnes sont triées dans l'ordre croissant de haut en bas. On obtient le résultat ($n \times n$ valeurs triées) « en serpent » au bout de $2 \log n + 1$ itérations. La figure 1 donne un exemple d'exécution du tri de 16 valeurs.

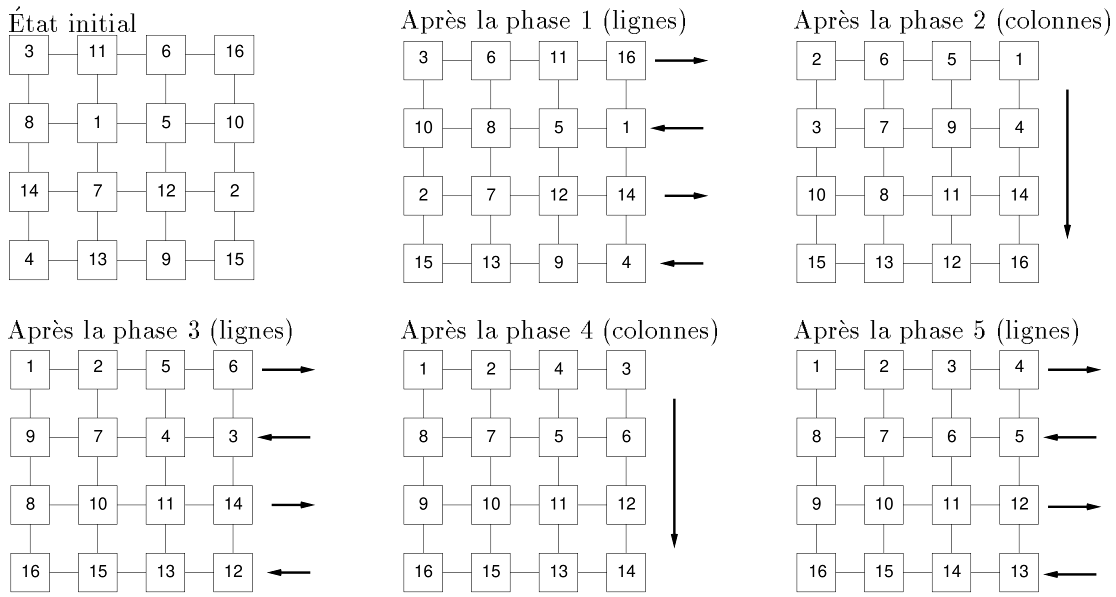


Fig. 1 – *Exemple d'exécution du shearsort pour $n = 4$*

Soit la MasPar du LIFL (128×128 processeurs). On applique l'algorithme du shearsort pour trier les 128×128 valeurs contenues dans une variable `plural int val`.

Exercice 1
Donnez l'algorithme d'une phase de tri des 128 lignes (phase impaire du shearsort). Explicitez en

particulier en quoi les traitements des lignes paires et impaires peuvent ou ne peuvent pas être réalisés simultanément.

Exercice 2
Donnez le code MPL implantant l'algorithme de l'exercice précédent.

Exercice 3
Discutez de l'implantation de l'algorithme shearsort pour trier les $128p \times 128p$ valeurs contenues dans une variable `plural int val [p][p]`.

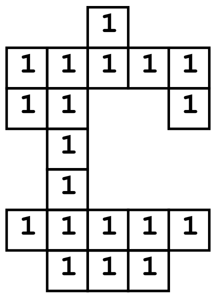
2 Mise en relief d'une image en C/PVM

L'objet de cet exercice est la construction d'une image en relief à partir d'un squelette d'image.

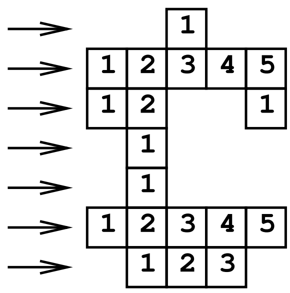
Image des distances Soit M_s une matrice de taille $n \times n$ à valeurs 0 ou 1. On appelle image des distances suivant la direction k de M_s la matrice M_{d_k} de taille $n \times n$ et à valeurs entières non-négatives dans laquelle les valeurs sont données par les conditions suivantes ; on suppose que les points du bord ne contiennent que des zéros :

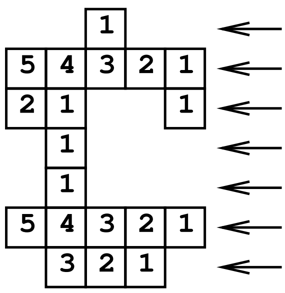
1. Si $M_s[i,j] = 0$, alors $M_{d_k}[i,j] = 0$.
2. Si $M_s[i,j] = 1$, alors $M_{d_k}[i,j]$ contient la plus petite distance dans la direction k d'un point (i',j') au point (i,j) tel que $M_s[i',j'] = 0$. La distance correspond au nombre de déplacements élémentaires, dans la direction k qu'il faut faire. Seules les quatre directions cardinales sont considérées ($1 \le k \le 4$).

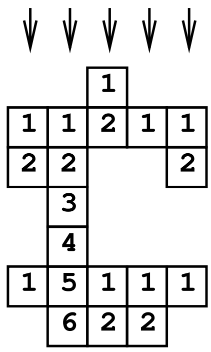
Exemple Soit la matrice M_s suivante (les valeurs nulles ne sont pas représentées) :

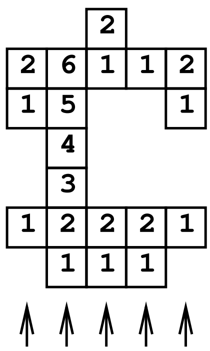


Les images des distances suivant quatre directions (dans l'ordre est, ouest, sud, nord) sont les suivantes :









Calcul de l'image des distances Le calcul de l'image des distances suivant une direction k est réalisé par une opération de préfixe parallèle (ou scan) par segments. Cette opération fonctionne sur le même principe que le préfixe parallèle, mais, au lieu de calculer le préfixe sur tout le vecteur, on le calcul « par morceaux » ; les morceaux étant définis comme les segments du vecteur ne contenant pas de valeurs nulles.

Somme des images des distances La somme des images des distances pour les quatre directions cardinales résulte en une matrice M_Σ. Cette matrice est la matrice relief de la matrice initiale M_s. La valeur $M_\Sigma(i,j)$ est l'altitude du point (i,j).

Implémentation au dessus de PVM On travaille sur une *Parallel Virtual Machine* (PVM) de N tâches. On désire implanter le calcul d'une matrice relief. On choisit de distribuer les structures de données associées aux matrices en attribuant p colonnes à chacune des N tâches ($n = p \times N$).

Exercice 4

- Argumentez la distribution des données choisie.
- Explicitez les structures de données devant être allouées sur une tâche donnée en distinguant les structures nécessaires au rangement des données provenant des matrice de celles nécessaires à la gestion de la PVM.

Exercice 5
Quelle(s) information(s) doit recevoir chaque tâche pour compléter son calcul local de M_Σ ?

Exercice 6

1. Donnez l'algorithme d'une tâche pour le calcul d'une matrice M_Σ.
2. Expliquez particulièrement en quoi il est possible de procéder à un recouvrement des communications par des calculs locaux.

Exercice 7
Donnez le code C/PVM d'une tâche du programme implantant cet algorithme.

Exercice 8
Imaginez-vous une structure que proposerait une bibliothèque de passage de message qui faciliterait une programmation efficace du présent algorithme?