Communication dans les réseaux

Résumé: Dans ce TD, nous allons écrire des algorithmes parallèles adaptés à des réseaux aux topologies plus ou moins exotiques.

1 Cubes connectés en cycles

Définition 1. Un réseau $CCC(m)$ est obtenue en remplaçant chaque processeur d'un hypercube de dimension m par un anneau de m processeurs et en connectant chaque processeur de l'anneau dans une dimension de l'hypercube (voir Figure 1).

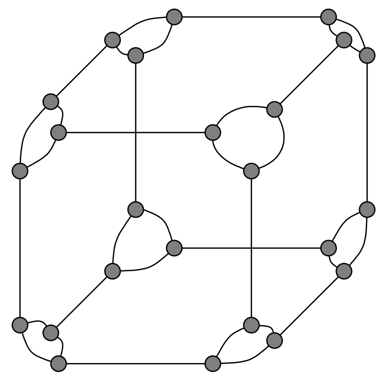

FIG. 1 – Cube connecté en cycle de taille 3.

▷ **Question 1.** Quelle est le nombre de processeurs de $CCC(m)$? Quel est le diamètre de $CCC(m)$?

Réponse. Le nombre de processeurs est clairement égal à $m \cdot 2^m$ pour ce qui est du diamètre, cela demande un peu plus d'explications.

Un sommet peut se représenter par le couple $< w, i >$ où $i \in [\![1, m]\!]$ et w est une suite de m bits. $< w, i >$ et $< w', i' >$ sont liés si et seulement si

- $w = w'$ et $i - i' \equiv \pm 1[m]$ (arête de cycle) ou
- $i = i'$ et w et w' diffèrent sur le $i^{\text{ème}}$ bit (arête d'hypercube).

Il n'est pas bien difficile de voir que $< s, i >$ et $< \bar{s}, i + \lfloor m/2 \rfloor [m] >$ sont les points les plus éloignés. L'algo de routage suivant montrera qu'ils sont à distance $2m - 1 + \max\left(1, \lceil \frac{m-3}{2} \rceil\right)$. □

▷ **Question 2.** Proposer un algorithme de routage d'un message d'un processeur à un autre dans un $CCC(m)$ et évaluer son coût.

Réponse. La représentation des sommets permet de faire le routage simplement. Pour aller de $< w, i >$ à $< w', i' >$:

- si $w =< \dots \alpha \dots, i >$ et w' diffèrent sur le $i^{\text{ème}}$ bit alors se déplacer vers $< \dots \bar{\alpha} \dots, i >$ puis vers $< \dots \bar{\alpha} \dots, i - 1 >$ et recommencer,
- sinon on se déplace vers $< \dots \alpha \dots, i - 1 >$ et on recomence.

Donc au bout de m étapes de ce type (donc d'au plus $2m$ sauts) on s'est déplacé jusqu'à $< w', i >$. Ensuite, on peut se déplacer en au plus $\lfloor \frac{m}{2} \rfloor$ sauts jusqu'à $< w', i' >$ en se déplaçant sur l'anneau.

Seulement pourquoi se déplacer vers $< \dots, i - 1 >$ plutôt que vers $< \dots, i + 1 >$. En fait, un choix judicieux en fonction de la posicieux entre i et i' permet de décider dans quel sens il vaut mieux se déplacer et d'économiser quelques sauts dans les derniers déplacements sur l'anneau. □

▷ **Question 3.** Proposer un algorithme de diffusion d'un processeur à tous les autres dans un $CCC(m)$ et évaluer son coût.

Réponse. La diffusion peut se faire en $3m + 1$ étape. Dans un $CCC(m)$, un sommet peut toujours être considéré comme un sommet passerelle entre deux $CCC(m - 1)$. On envoie d'abord vers l'autre hypercube puis on innonde sur son propre hypercube et l'autre hypercube en parallèle. Comme chacun des cycles a été augmenté d'un élément la diffusion dans les $CCC(m-1)$ peut prendre une étape de plus et donc $d(m) = 2 + d(m-1) + 1 = 3m - 2$. Seulement voilà la fameuse étape obligatoire n'est pas toujours necessaire et dépend de la parité de m (1 si m est pair et 0 sinon). Donc le vrai temps est plutôt $d(m) = \lfloor \frac{5m}{2} \rfloor - 1 \ldots$ □

2 Transposition d'une matrice

On veut concevoir une algorithme parallèle pour la transposition d'une matrice $n \times n$. On suppose la matrice stockée de manière distribuée dans les processeurs. On supposera que les différents liens de communication sont bi-directionnels et peuvent être utilisés de manière simultanée.

▷ **Question 4.** Proposer une solution sur un anneau de p processeurs et donner sa complexité (on suppose que la distribution est mono-dimensionnelle).

Réponse. Regardons d'abord quelles communications sont nécessaires à la transposition de la matrice sur une ligne de 4 processeurs (voir Figure 2).

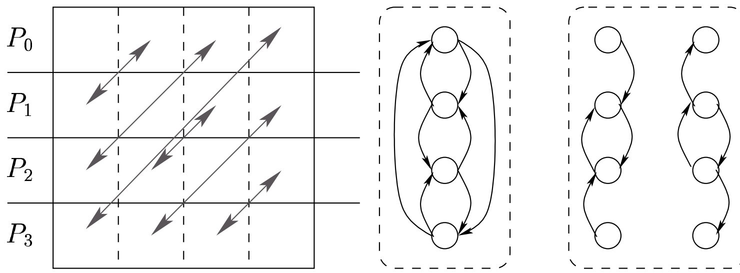

FIG. 2 – Mouvements de données sur une ligne de 4 processeurs.

Chaque processeur doit communiquer une donnée particulière à tous les autres processeurs. C'est un échange total personnalisé. En regroupant les communications en fonction de la distance à la quelle se trouvent les processeurs impliqués, on obtient les phases de la Figure 3.

On notera que dans chacune de ces phases (à part peut-être la dernière) chacun des processeurs échange des données avec deux autres processeurs et qu'ainsi, comme la distances entre chaque couple de processeurs est constante, en "pipelinant" les communications, chaque lien de communication est utilisé en permanence et chaque message emprunte le chemin le plus court possible.

La quantité de communications nécessaires est donc égale à

$$\sum_{k=1}^{\lfloor \frac{p}{2} \rfloor} k(\beta + \tau \frac{n^2}{p^2}) = \frac{1}{2} \left(\left\lfloor \frac{p}{2} \right\rfloor \left(\left\lfloor \frac{p}{2} \right\rfloor + 1 \right) \right) \left(\tau \frac{n^2}{p^2} + \beta \right) \sim \tau \frac{n^2}{8} + \beta \frac{p^2}{8}$$

□

▷ **Question 5.** Proposer une solution sur une grille torique $p = q \times q$ processeurs et donner sa complexité (on suppose que la distribution est bi-dimensionnelle).

Réponse. De même que précédemment, regroupons les communication en fonction de la distance séparant les processeurs (voir Figure 4).

En décomposant les communications le long des arcs, on peut s'apercevoir qu'à une étape donnée, un processeur participe à une seule des communications. Du coup, le temps des communications nécessaires à la mise en œuvre d'une transposition est égal à

$$\left\lfloor \frac{q}{2} \right\rfloor \cdot \left(\tau \frac{n^2}{q^2} + \beta \right) \sim \tau \frac{n^2}{2\sqrt{p}} + \beta \frac{\sqrt{p}}{2}$$

Cet algorithme est clairement optimal puisque son temps d'exécution est aussi égal au temps de communication d'une portion de matrice entre les deux processeurs les plus éloignés. □

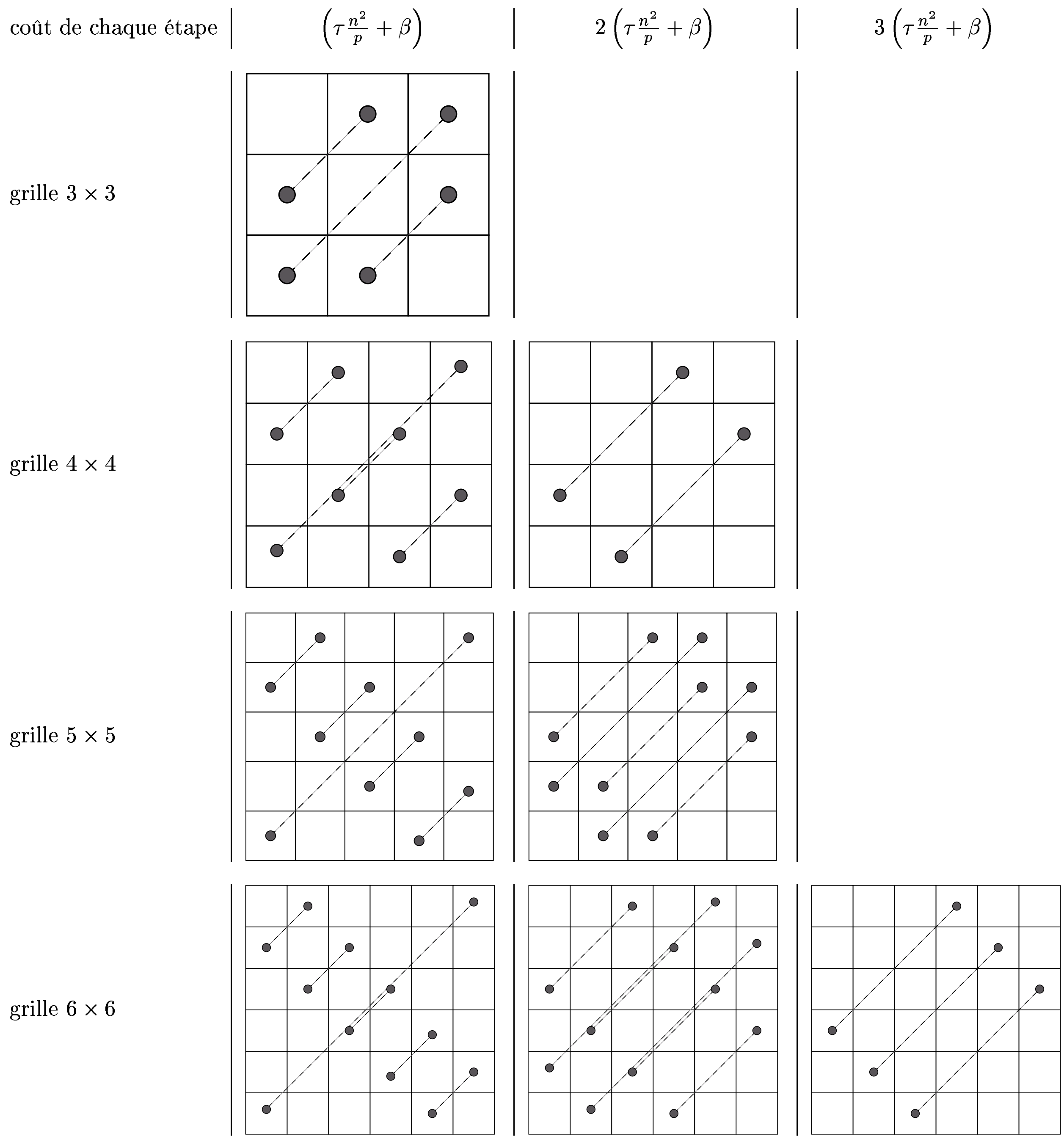

FIG. 3 – Organisation des communications sur des lignes de processeurs.

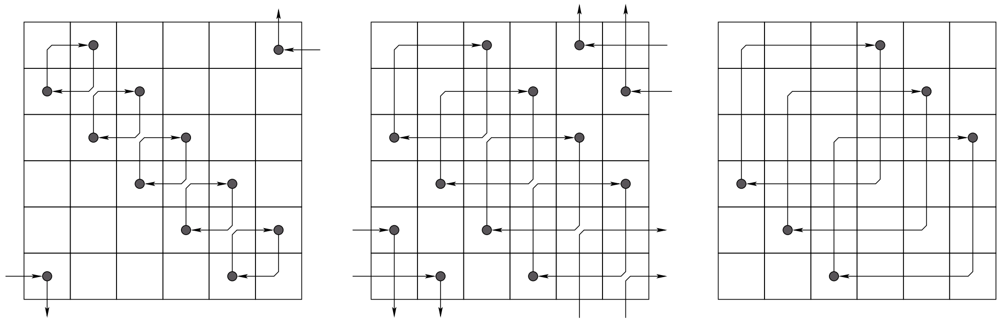

FIG. 4 – Mouvements de données sur une grille de 6 × 6processeurs.

▷ **Question 6.** Proposer une solution sur un hypercube de $p = 2^m$ processeurs et donner sa complexité (on suppose que la distribution est bi-dimensionnelle).

Réponse. La structure récursive des hypercubes permet de mettre facilement en œuvre la transposition récursive illustrée en Figure 5. Du coup, le temps des communications nécessaires à la mise en œuvre d'une

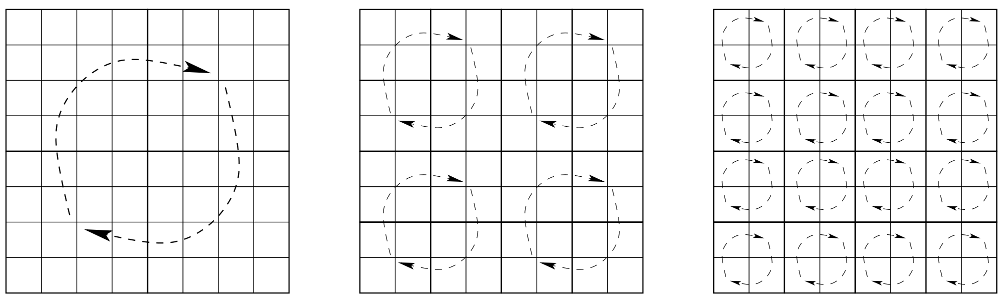

FIG. 5 – Mouvements de données sur un hypercube de dimension 4.

transposition est égal à

$$(m-1) \cdot \left(\tau \frac{n^2}{p} + \beta\right) \sim \tau \frac{(\log p) n^2}{p} + \beta \log p$$

□