TD 3 Architectures de communication

3.1 Tores et hypercube

Le tore de dimension k et de base n, noté $T(n,k)$, possède $N = n^k$ sommets, chacun d'entre eux étant connecté à un cycle dans chacune des k dimensions. $T(n,2)$ est noté $T(n)$. L'hypercube à n dimensions est noté $H(n)$.

1. Dessiner $T(2)$, $T(4)$ et $T(2,4)$. A quels réseaux peut-on comparer : un tore de dimension 2 ; un tore de base 2 ?
2. Calculer le degré, le diamètre, la distance moyenne et la bisection géométrique de $T(n)$, $H(n)$ et $T(n,k)$.
3. On considère $H(n)$ et $T(n)$. Chaque sommet émet sur le réseau avec le taux λ et une distribution uniforme des destinations. Majorer λ (les unités sont telles que le débit bidirectionnel d'un lien est égal à 1).
4. Comparer les latences moyennes de $T(64)$ et $H(12)$ en supposant des liens de largeur W, en routage paquet et en routage wormhole.
5. La *bisection physique* est le nombre de liens de 1 bit qui traversent la médiatrice du réseau. Pour un réseau implanté dans une surface donnée, la bisection physique est contrainte par la technologie disponible. On veut maintenant tenir compte de cette contrainte pour la comparaison des réseaux.
   Quelle hypothèse faut-il modifier par rapport au modèle précédent ? Comparer les latences moyennes de $T(64)$ et $H(12)$ en supposant des liens de largeur 1 bit pour $H(12)$, en routage paquet et en routage wormhole.

3.2 Réseau de Beneš

1. Dessiner les réseaux de Beneš à 4, 8 et 16 entrées.
2. Un algorithme de routage off-line du réseau de Beneš est décrit fig. 1. Appliquer cet algorithme au réseau de Beneš à 8 entrées, et à la permutation (5 3 4 7 0 1 2 6).
3. L'algorithme précédent peut-il être utilisé pour le routage on-line ?

3.3 Réseau butterfly

$\mathcal{B}_n$ est le réseau butterfly à $N = 2^n$ entrées.

1. Combien existe-t-il de chemins dans $\mathcal{B}_n$ entre une entrée et une sortie donnée ? En déduire un algorithme de routage du butterfly (routage *glouton*). Quelle est la complexité matérielle du routage glouton ?

On considère le réseau de Beneš comme un réseau de Clos, donc avec deux commutateurs intermédiaires. Soit s une entrée quelconque et d sa destination. Router s par le commutateur intermédiaire supérieur. Router la sortie placée sur le même commutateur du denier étage que d par le commutateur intermédiaire inférieur. Satisfaire la contrainte créee sur le premier étage en routant par le commutateur intermédiaire supérieur. Lorsque ce processus aboutit à un commutateur du premier étage dont l'autre entrée est déjà routée, recommencer en choisissant une entrée non routée. Le routage des commutateurs intermédiaires utilise le même algorithme.

Fig. 1 – Routage du réseau de Beneš

3.4 Performances

Myrinet

Etude des documents joints.

Modélisation

Un algorithme effectue en chaque point d'un tableau $N \times N$ un calcul qui fait intervenir ses 4 voisins (ex. moyenne -*stencil* 5 points). On considère une machine MIMD, mémoire distribuée à 1024 processeurs, organisée en tore 2D. Chaque processeur opère à 2 GFlop/s. La latence L pour les messages courts est 10μ s, le débit asymptotique r est 1 Gb/s ; le temps de communication est $s + mr^{-1}$, où m est la longueur du message.

1. Comparer les temps de communication pour les distributions bloc et bande.
2. Les données sont partitionnées par blocs, et les blocs sont distribués sur les processeurs, soit "naturellement", soit de façon aléatoire uniforme. Déterminer la valeur de N qui permet le recouvrement communication-calcul.