

CORRECTION DU TD N°1

Métriques de performance

Exercice 1 : Etude du coût de ressources en *size up*

1- Trouver le nombre p de noeuds à utiliser pour traiter un problème de taille n aussi vite qu'un problème de taille n_0 sur 1 noeud.

On définit naturellement $T^{ideal}(n; p)$ par : $T^{Ideal}(n; p) = \frac{q(n)}{p.V}$

On cherche alors le nombre p de noeuds à utiliser pour traiter un problème de taille n aussi vite qu'un problème de taille n_0 sur 1 noeud :

$$T^{Ideal}(n; p) = T^{Ideal}(n_0; 1)$$

$$\frac{q(n)}{p.V} = \frac{q(n_0)}{1.V}$$

D'où

$$p = \frac{q(n)}{q(n_0)}$$

2- Soit un problème de complexité $O(n^2)$. Déterminer le nombre de ressources p nécessaires pour un problème de taille $n=2n_0$, ensuite pour un problème de taille $n=kn_0$

Indication : Pour un problème de complexité $O(n^2)$, on prend $q(n_0) \approx \alpha.n_0^2$

Si on considère un problème de complexité $O(n^2)$. Un problème de taille n_0 engendre alors $q(n_0) \approx \alpha.n_0^2$ opérations, et un problème de taille $2.n_0$ engendre $q(2.n_0) \approx \alpha.(2.n_0)^2$ opérations. On obtient donc :

$$p = \frac{\alpha.(2.n_0)^2}{\alpha.n_0^2} = 4$$

il nous faut utiliser $p = 4$ noeuds de calcul pour maintenir constant le temps d'exécution :

$$T^{Ideal}(2.n_0; 4) = T^{Ideal}(n_0; 1)$$

Un problème de taille $k.n_0$, engendre $q(k.n_0) \approx \alpha.(k.n_0)^2$ opérations. On obtient donc :

$$p = \frac{\alpha \cdot (k \cdot n_0)^2}{\alpha \cdot n_0^2} = k^2$$

il nous faut utiliser $p = k^2$ noeuds de calcul pour maintenir constant le temps d'exécution :

$$T^{Ideal}(k \cdot n_0; k^2) = T^{Ideal}(n_0; 1)$$

3- La courbe bleue de la figure illustre le cas idéal où

$$T(100; 1) = T(200; 4) = T(400; 16) = T(k \cdot 100, k^2).$$

Les temps d'exécutions sont constants et sont obtenus pour les nombres de ressources prévus (k^2).

– La courbe verte montre un cas pratique où l'on arrive à maintenir constant le temps d'exécution mais où les nombres de ressources à utiliser sont un peu supérieurs aux prévisions.

– Enfin la courbe orange montre un cas où l'on arrive à maintenir constant le temps d'exécution, mais où le nombre de ressources à utiliser augmente beaucoup plus vite que celui prévu théoriquement.

La courbe orange illustre donc une distribution qui fonctionne, et qui permet même de traiter des problèmes plus gros sur un plus grand nombre de machines, mais qui est très imparfaite. Les raisons les plus fréquentes sont les communications et opérations de synchronisation qui croissent plus vite que les calculs quand la taille du problème et le nombre de noeuds augmentent. Dès lors, réaliser un size up à temps d'exécution constant peut vite devenir très coûteux, voire insupportable, et l'on doit se contenter d'un size up à temps d'exécution croissant.

Dr. Tahar Mehenni