Examen de « Parallélisme, systèmes distribués et grille »

Master ILC – M2 – 2009-2010

Tous documents autorisés

Durée : 3h00

Attention : faire les deux parties de l'examen sur des feuilles séparées.

I – Partie de S. Vialle (10 points)

Q1 : 4 pt : Des équipes d'ingénieurs différentes doivent faire exécuter un grand nombre de tâches de calculs tout au long de chaque journée de travail. Il est décidé d'acheter un ensemble de PCs pour répondre à ce besoin.

Vous pouvez acheter :

- Des PC bi-cœurs avec 4Go de RAM, à 500 Euros pièce
- Des PC quadri-cœurs avec 4Go de RAM, à 750 Euros pièce
- Un réseau Gigabit Ethernet (switch(s), câbles et cartes) pour une moyenne de 100 Euros supplémentaires par PC
- Un réseau Infiniband (switch(s), câbles et cartes) plus performant pour une moyenne de 500 Euros supplémentaires par PC

Vous disposez d'un budget limité (par exemple : 50 000 Euros). Bien sur on peut acheter un cluster de PC « standards » avec un réseau d'interconnexion « standard » pour exécuter tout type de tâches. Mais si on connait certaines caractéristiques des tâches à exécuter on peut optimiser l'achat du matériel.

Plusieurs solutions sont justifiables pour les quatre cas d'utilisation décrits ci-dessous. Pour chacune d'elle proposez une solution et présentez votre raisonnement.

a) 1 pt : Les tâches à exécuter sont toutes des tâches séquentielles indépendantes très gourmandes en mémoire.

Qu'achetez-vous ? et pourquoi ?

b) 1 pt : Les tâches à exécuter sont des tâches séquentielles indépendantes et consommant chacune peu de mémoire.

Qu'achetez-vous ? et pourquoi ?

c) 1 pt : Chaque tâche est implantée sous forme d'un programme parallèle créant et exécutant plusieurs threads : un nombre de threads correspondant automatiquement au nombre de cœurs de la machine où est exécutée la tâche.

Qu'achetez-vous ? et pourquoi ?

d) 1 pt : Chaque tâche est un programme distribué utilisant de 2 à 8 nœuds de calcul, et échangeant de nombreuses données et résultats intermédiaires entre ces nœuds. Sur chaque nœud le code exécuté est en revanche séquentiel.

Qu'achetez-vous ? et pourquoi ?

Q2 : 6 points : On considère un programme parallèle MPI où P processus sont numérotées de 0 à P-1, et répartis sur P machines. Ces P machines sont virtuellement considérées sur un anneau bidirectionnel, mais sont en réalité reliées à un switch Gigabit Ethernet. Chaque machine peut donc communiquer directement avec n'importe quelle autre machine.

Chaque processeur possède 4 tables de N 'doubles' à envoyer : TabSendG1, TabSendG2, TabSendD1, TabSendD2, et 4 tables de 'N doubles' à recevoir : TabRecvG1, TabRecvG2, TabRecvD1 et TabRecvD2.

Chaque processus MPI stocke son numéro dans une variable 'Me'. Arrivé à un certain point du programme, chaque processus MPI :

- envoie sa table TabSendG2 au processus '2 crans à gauche' c'est-à-dire à celui de numéro (Me-2+P)%P,
- envoie sa table TabSendG1 au processus '1 cran à gauche' de numéro (M-1+P)%P,
- envoie sa table TabSendD1 au processus '1 cran à droite' de numéro (M+1)%P,
- envoie sa table TabSendD2 au processus '2 crans à droite' de numéro (M+2)%P.

Rappel : *le module (%) ne fonctionne que sur les nombres positifs, donc pour calculer le numéro d'un processus 'x crans à gauche' on calcule (Me-x+P)%P et pas (Me-x)%P, pour éviter les erreurs quand (Me-x) est négatif. Ce mécanisme permet de considérer que le voisin de gauche du processus 0 est en fait le processus P-1 situé à l'autre extrémité de l'anneau (comme vu en TD).*

Symétriquement, chaque processus reçoit une table de ses deux voisins de gauche et de ses deux voisins de droite, qu'il stocke dans TabRecvG1, TabRecvG2, TabRecvD1 et TabRecv.

a) 3 pts : On va implanter cette étape de communication avec des envois de messages non bloquants et *bufferisés* en utilisant MPI_Bsend(…), et avec des réceptions de messages bloquants en utilisant MPI_Recv(…). On suppose qu'il y a plus de 5 processus. Complétez le code MPI suivant :

```
//-----------------------------------------------------
// Déclaration des variables
double TabSendG1[N], TabSendG2[N], TabSendD1[N], TabSendD2[N] ;
double TabRecvG1[N], TabRecvG2[N], TabRecvD1[N], TabRecvD2[N] ;

…… // Partie précédente de l'application
```

```
//-----------------------------------------------------
Int BuffSize = 4*(sizeof(double)*N+MPI_BSEND_OVERHEAD);
double *Buff = malloc(BuffSize);
if (Buff == NULL) {
  printf("Erreur: pas assez de mémoire disponible!\n ») ;
  exit(EXIT_FAILURE) ;
} else {
  MPI_Buffer_attach(Buff,BuffSize) ;
}
// ****INSEREZ ici vos opérations de communications ****
……
MPI_Buffer_dettach(&Buff,&BuffSize) ;
free(Buff);
//-----------------------------------------------------
…… // Partie suivante de l'application
//-----------------------------------------------------
```

b) 3 pts : On souhaite maintenant utiliser des communications fortement synchronisées : les envois et les réceptions sont bloquants, avec MPI_Ssend(…) et MPI_Recv(…).

Cette fois-ci il n'y a plus lieu d'allouer et d'utiliser un buffer, mais il faut faire plus attention à éviter les *interblocages*. On suppose qu'il y a au moins 6 processus et que le nombre de processus est pair : P = 2p. Complétez le code suivant :

```
//-----------------------------------------------------
// Déclaration des variables
double TabSendG1[N], TabSendG2[N], TabSendD1[N], TabSendD2[N] ;
double TabRecvG1[N], TabRecvG2[N], TabRecvD1[N], TabRecvD2[N] ;

…… // Partie précédente de l'application
//-----------------------------------------------------
// ****INSEREZ ici vos opérations de communications ****
……
//-----------------------------------------------------
…… // Partie suivante de l'application
//-----------------------------------------------------
```

II – Partie de S. Genaud (10 points)

Q3 : 10 pt : Une entreprise acquiert régulièrement un grand nombre de films sous forme de fichiers au format DV, et elle désire aujourd'hui les convertir au fur et à mesure vers des fichiers MPEG-4. Elle compte utiliser le logiciel libre `ffmpeg` pour la conversion. Les fichiers sont importés à partir de caméscopes par les différents utilisateurs à partir de leurs PCs respectifs. On estime qu'il y a en permanence dans le système de l'ordre de 1000 fichiers DV à traiter, chaque fichier ayant une taille moyenne de 10 Go. Une fois converti, le fichier DV peut être supprimé mais le MPEG-4 devrait faire l'objet d'une sauvegarde.

L'entreprise n'a pas de ressources de calcul dédiées, mais possède un parc de 50 PC assez récents. Ce parc représente une puissance de calcul suffisante pour les besoins à condition d'utiliser les machines quand leurs utilisateurs habituels les laissent inactives.

Par ailleurs, l'entreprise s'interroge sur l'opportunité d'acheter un cluster de 12 noeuds quadcore, dont le coût est 20000 Euros.

On souhaite que différents utilisateurs puissent avoir une interface permettant de lister les fichiers DV non encore convertis, et de demander à ce que le système prenne en charge de manière transparente leur conversion. Le fichier MPEG-4 produit devra être transféré sur le disque d'un serveur web. Les utilisateurs doivent pouvoir commander les conversions d'où qu'ils se connectent, que ce soit à l'intérieur ou de l'extérieur de l'entreprise[1]. Un avantage sera donné à une solution permettant de déclencher des conversions par simple envoi de mail.

a) Donnez une liste des problèmes qui se présentent dans ce système utilisé simultanément par plusieurs utilisateurs.

b) Proposez une solution argumentée pour ce besoin, dans le cas où l'entreprise n'utilise que ses ressources existantes, et dans le cas où elle achète un cluster. Dans chaque cas :

- Schématisez l'architecture logicielle proposée en dessinant les clients et serveurs, les logiciels gestionnaires par rapport à l'architecture physique. Notez quels protocoles applicatifs sont utilisés pour transporter les messages sur le réseau.
- Proposez une solution pour gérer les données.
- Dites quels seront les développements spécifiques nécessaires ou quels types de logiciels existants peuvent être utilisés.
- Dites quels sont les avantages et les inconvénients de votre solution. Quels sont les problèmes prévisibles et dire s'il existe des mesures préventives adaptées.

[1] On suppose que la sécurité est assurée par un VPN et une identification par adresse MAC sur connexion sécurisée. Il n'y a pas besoin de login/mot de passe.