Benchmarking | Travaux Pratiques

ANGD, Autrans 2009

LIBERATE IT

Ressources à votre disposition - Accès

- Accès via la machine pont ciment
  - ssh acces-ciment.imag.fr -l <*login*>
- **healthphy**
  - Accès : ssh healthphy.ujf-grenoble.fr
  - SMP SGI – 18 Noeuds bi-sockets bi-coeurs Itanium Montecito 1.4 GHz, interconnect NumaLink
  - Batch Manager : PBSPro, queue « it »
- **tao**
  - Accès : ssh -p 9922 telma.frec.bull.fr -l <*login*>
  - 36 Noeuds BULL R422-E2 bi-sockets quadri-coeurs Xeon Nehalem 2.93 GHz, interconnect IB-QDR
  - Batch Manager : PBSPro, queue par défaut

Soumettre des jobs

- **Healthphy**
  - Un job interactif sur 1 nœud

    qsub -I -q it -lselect=1:ncpus=4
  - Identifier le noeud sur lequel je tourne :

    cat /dev/cpuset/PBSPro/$PBS_JOBID/mems

- **Tao**
  - Un job interactif sur 1 nœud, 8 processeurs, 8 taches MPI

    qsub -I -lselect=1:ncpus=8:mpiprocs=8 -lplace=scatter -lpanasas=0 -lwalltime=30:00

Attention : pour faciliter la compréhension, faîtes vos jobs en interactif. En production ce N'EST PAS LA BONNE METHODE. N'oubliez pas de libérer vos ressources

Premiers contacts avec les machines

- Processeurs
  cat /proc/cpuinfo
- Nœuds NUMA
  numactl -- show
  numactl -- hardware
  numactl –show
  nodeinfo (healthphy)

CPU & Mémoire

LIBERATE IT

CPU - DGEMM

- Sur healthphy
  - benchmarks/PERFS
  - Vérifier le fichier Make.healthy
  - Compiler le code : make arch=healthphy
  - Exécuter
    - Exécution directe
      - → Quelle performance ? Pourquoi ?
      - → Forcer la fréquence avec la variable d'environnement FREQ
    - Faites varier le nombre de threads OpenMP (OMP_NUM_THREADS)
    - Faites varier la taille du tableau (argument de la commande)
      - → Par défaut c'est 1000
      - → Alloue trois tableau de double ; controler votre estimation de la mémoire avec nodeinfo
    - Sur deux noeuds (ncpus=8) avec OMP_NUM_THREADS=4
      - → Jouer sur le placement :
        - FREQ=1600 OMP_NUM_THREADS=4 numactl --membind=2 --cpunoodebind=2 bin/dgemm-omp

CPU - DGEMM

- Sur tao
  - Rejouer un certains nombre des commandes précedentes (en adaptant) et comparer les perfs brutes (même nombre de coeurs).

Mémoire - STREAM

- Sur healthy
- benchmarks/stream
  - Compiler le code (-O2 -ip -openmp)
    - La taille des tableau est contrôle par le flag de compile -DN=<taille>
    - La taille par défaut est 2000000.
  - Exécuter le sur 1 noeud 4 ou 5 fois pour la taille par défaut
  - Idem en multipliant N par 10
    - Que constatez vous ?
  - Sur 2 noeuds, 8 process, taille standard, multiple répétitions
    - Que constatez vous ?
  - Sur 2 noeuds, 4 process, jouer avec le placement déduisez-en le facteur numa
    OMP_NUM_THREADS=4 numactl --membind=n --cpunodebind=m ./stream
    - Est-ce la seule possibilité ?
  - Avancé : commentez la ligne 173 ; exécuter sur 1 et 2 noeuds.

Mémoire - STREAM

- Sur tao (un seul serveur)
  - Compiler le code (-O2 -ip -openmp)
    - La taille des tableau est contrôle par le flag de compile -DN=<taille>
    - La taille par défaut est 2000000.
    - Augmenter la taille (x10)
  - Exécuter le sur 1 noeud
  - Recompiler avec l'option '-opt-streaming-stores always'
    - Que constatez nous ? Pourquoi ? (le man de icc peut vous aider !)
  - Sur 1 serveur, 4 process, jouer avec le placement déduisez-en le facteur numa entre les sockets.
    OMP_NUM_THREADS=4 numactl --membind=n --cpunodebind=m ./stream
    - Est-ce la seule possibilité ?
  - Avancé :
    - Commentez la ligne 173 ; exécuter sur 1 et 2 noeuds.
    - Augmenter la taille (x100)

LMBENCH3

- Sur tao et healthphy
- benchmarks/lmbench3
- Vérifier le makefile et compiler
- Executer :
  - numactl --localalloc ./lat_mem_rd -P 1 -N 1 -t 50
- Déduisez-en :
  - Le profil de la hiérarchie mémoire sur les machines
  - Les latences (a fortiori !) d'accès.

llcbench

- Sur tao et healthphy
- benchmarks/llcbench
- Vérifier le makefile et sys.def ; compiler cache-bench et blas-bench
  - Faites juste make pour l'aide
- Cache-bench :
  - Executer (pas via la commande make)
    - numactl --localalloc ./lat_mem_rd -P 1 -N 1 -t 50
  - Déduisez-en :
    - Le profil de la hiérarchie mémoire sur les machines ; comparer aux resultats le lmbench
  - Avancé : Sur tao : recompiler avec l'option '-opt-streaming-stores always' ; comparer aux résultats sans l'option. Des idées ?
- BLAS-bench
  - Executer et comparer les deux architecture : votre choix ?

Applicatifs

LIBERATE IT

Applicatifs

- Sur tao et healthphy (faites le travail en parallèle sur les deux machines)
- benchmarks/NPB3.2.1/NPB3.2-OMP
  - Vérifier config/make.def
  - Compiler CG pour différente « class » de cas
  - Exécuter, comparer.
- benchmarks/ONE-ZONE
  - Compiler

    make arch=intel clean all
    - Que constatez-vous entre les deux plateformes ?
  - Executer

    cd BENCH

    ../SRC/<arch>/1zone.<arch> pks2155-HESS-1.param
  - Comparer
    - En supposant que tout est sur la fréquence ?

Interconnect & MPI

LIBERATE IT

Jobs MPI : premier job sur TAO

- Sur tao
  - Faites vous la main sur un 'Hello World'
- Benchmarks/hw
- Compiler

  mpicc -o hw hw.c
- Executer

  mpirun –rsh=ssh -np <np> ./hw

  mpdallexit
- Executer 2 processus MPI avec
  - 1 noeud, 2 PPN -ppn 2 -np 2
  - 2 noeuds, 1 PPN -ppn 1 -np 2
- Sur 2 noeuds, 16 processus, 1 ppn (???) :

  mpirun –rsh=ssh -ppn 1 -np 16 ./hw
- Bravo, vous êtes un vrai utilisateur de MPI ; Vous êtes prêt pour le grand bain !

OMB (OSU MPI Benchmarks)

- Sur tao
- benchmarks/osu_benchmarks
- Compiler
- Executer
  - 1 noeud, 2 PPN
  - 2 noeuds, 1 PPN
    - osu_bw
    - osu_lat

IMB (Intel MPI Benchmarks)

- Sur tao
- benchmarks/IMB-3.2
- Compiler

  make -f make_tao
- Executer
  - 1 noeud, 2 PPN
  - 2 noeuds, 1 PPN
    - PingPong, PingPing, AlltoAll ...
  - Comparer avec OSU
- Avancé : détermination du taux de messages
  - PingPing en mode multi sur 2 noeuds, 16 processus, 1 ppn

    mpirun --rsh=ssh -ppn 1 -np 16 ./IMB-MPI1 -multi 1 PingPing

HPL

- Sur tao
  - But : faire tourner un HPL intranoeud.
- benchmarks/hpl-2.0
- Verifiez le fichier Make.nehalem
- Compiler

  make arch=nehalem
- Dans bin/nehalem
  - Ecrasez le fichier HPL.dat avec ../../HPL.dat.nehalem
    - Modifier p,q
    - Calculer N pour 25% de la mémoire du nœud (24 GiB/Noeud)

      → Une estimation en temps ?
    - Vérifier votre estimation !