CSYE7105 HW2

Instructor Dr. Handan Liu

2020-10-11

1. **Show the node list, CPU numbers, etc. of the “debug” partition. 2pts**

sinfo -p debug

1. **When in-person reviewing (Zoom), tell the TA the meaning of the following code of the job states and what you should do if it happens: PD, R, S, CG 8pts**

pd – The job is waiting for resource allocation. It will eventually run.

If your waiting job cannot complete before a maintenance/reservation begins, showq will display its state as \*\*WaitNod\*\* ("Waiting for Nodes"). The job will remain in this state until Frontera returns to production.

r – The job currently is allocated to a node and is running.

s – A running job has been stopped with its cores released to other jobs. Jobs get suspend on a cluster if a high-priority job comes in. Add the S flag handling in the SLURM executor.

cg - The job is finishing but some processes are still active. it happens to a job that cannot be terminated, probably because of an I/O operation

1. **Show how to use srun to request one node and two tasks for 40 minutes with 16GB memory on the “express” partition. 5pts**

srun -p express --pty /bin/bash

srun -N 1 -n 2 -t 40 --mem=16GB --partition=express ./Newzy

1. **Show how to check the job information in the question 3, show its job\_id and the node name, and the job state. 5pts**

squeue -u sharma.new

1. **Show how to cancel the job in the question 3. 2pts**

scancel -u sharma.new

1. **In your $HOME, create a new directory: csye7105; then go to this directory, create a sub- directory: homework2 4 pts**

mkdir csye7105

mkdir homework2

1. **Transfer the previous files the instructor taught (and uploaded to Canvas): mm\_omp.c, mm\_mpi.c to your $HOME/csye7105/homework2.**

mv /Users/newzysharma/Documents/Documents/hw2/mm\_omp.c /Users/newzysharma/test/hw2

1. **Move to any compute node by srun**

srun -p express --pty /bin/bash

1. **On this compute node,**

**(1) Compile mm\_omp.c with gcc compiler with OpenMP flag.**

**(2) OpenMP parallel run the executable on 4 threads**

gcc -o omp -fopenmp mm\_omp.c

export OMP\_NUM\_THREADS=4

./omp

1. **On this compute node,**

**(1) Compile mm\_mpi.c with MPI compiler wrapper.**

**(2) MPI parallel run the executable on 4 processors.**

mpicc -o ompi mm\_mpi.c

mpirun -np 4 ./ompi

**Write a Slurm script for submitting jobs, require:**

1. **define the names of the job, output file and error file in the script;**
2. **define the running time for 5 minutes;**
3. **request one compute node from the “express” partition;**
4. **define your work directory, and go to this work directory;**
5. **load module of openmpi (recommend the default version 3.1.2);**
6. **commands to run your compiled MPI executables.**

**Submission: save this script as a batch file and submit it on Canvas.**

Part 3: 24 points

Three files of C code are given for matrix multiplication in serial, MPI and OpenMP.  
In this example (3 files), set matrix A is 500x500, matrix B is 500x300, matrix C is 500x300 That is to say: set NRA=500, NCA=500, and NCB=300

1. Please set timers in the 3 files. In parallel code, it requires to use the timer to calculate the elapsed time of the entire parallelized code and write the print-out code for the time; while the timer should be set the corresponding sections in the serial code. 18pts, each 6 pts.
2. Please compile the 3 files using appropriate compilers. 6 pts, each 2pts

Note: please comment the screen-printing lines of the matrix C in the original files because the new matrices are too large.

Submission: save your modification in the original files and submit these 3 C files.

Part 4: 18 points

Run the executables completed in Part 3:

1. Run the serial executable and print out the elapsed time. 2pts
2. Run the OpenMP parallel executable on 2, 4 and 8 threads, and print out the elapsed time respectively. 6pts, each 2pts
3. Run the MPI parallel code on 2, 4 and 8 processors, and print out the elapsed time respectively. 6pts, each 2pts

Note: You can operate the above 3 tasks in interactive mode or in batch mode.

1. Put all the recorded elapsed times into one table (in Excel or Word); plot them in the same graph. 4pts, each 2pts

Tips: You can plot by using Excel, Matplotlib, MATLAB or other tools.  
Submission: submit the table in an Excel or Word file. The image can be saved as an image file (jpg or png) or directly put together with the table in Excel or Word.

Part 5: 3 points

Pack (and/or compress) all files you should submit in this assignment as a tarball using tar and submit this tarball via Canvas. The tarball name should be your NEU username.tar (or tgz).  
*If you submit your assignment in other different ways, you will loss the 3 points.*