WITS UNIVERSITY

School of Computer Science and Applied Mathematics
UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

COMS3008A: Parallel Computing

2021-9-30

A simple tutorial for using MSL cluster

1. Logging into the cluster:

   ```
   ssh␣<username>@146.141.21.100
   ```

   or

   ```
   ssh␣-X␣<username>@146.141.21.100
   ```

   for opening a GUI on your local machine. Enter your login passwd at the prompt following the `ssh` command. If it is the first time you are logging in the cluster, you will see a message something like “The authenticity of the host cannot be … Are you sure you want to continue connecting (yes/no) ?”. Type **yes** to continue. This will add the remote server to the list of known hosts on your local machine.
2. Once you log in, you can use Linux shell commands to perform operations such as creating (`mkdir <directory name>`) or removing (`rm <directory name>`) a directory. Use

   ```
   scp␣</path/to/the/sourcefile>␣</path/to/the/destination>
   ```

   to copy a file between a local machine and a remote machine. For example, from my local machine to copy a file to the cluster:

   ```
   scp localFile.txt <myusername>@146.141.21.100:~/myFolder/
   ```

   copies a file ‘localFile.txt’ from my local machine to the remote machine. The symbol ‘~’ means your home directory.
3. Type `logout` in the terminal to exit the remote machine.
4. When you login, you will be logging into the head node of the cluster, which manages all the accounts. It is very important that you should never run your program directly on the head node; to run your program on the cluster, you should submit a job, usually a batch job that will run when the resources become available.
5. Note the following when you write a job script for Slurm, which is the cluster management and job scheduling tool that is used in the MSL cluster.

- Include the type of shell you are using as the first line in your batch script — `#!/bin/bash`.
- All `#SBATCH` lines must come before shell script commands.
- Batch scheduling script (statements that start with `#SBATCH`) is interpreted upon job submission. Shell scripting (shell commands) is only interpreted at runtime. Therefore, it is possible to successfully submit a job that has shell script errors which won't cause problems until later when the job actually runs.
- The `sbatch` command is used to submit your job. For example, `sbatch myjobscript`, where `myjobscrip` is the file contains batch and shell scripts.

6. For instance, the following script specifies some essential values such as partition, time limit, memory allocation, and number of cores. It also specifies additional parameters such as jobname and output file. (For more details, go to https://slurm.schedmd.com/quickstart.html)

```
#!/bin/bash
# These lines are for slurm
# specify a partition by its name in which to run your job. In this case, the name of the partition is batch
#SBATCH -p batch
# specify number of nodes you are requesting to run your job
#SBATCH -N 2
# or
#SBATCH --nodes=2
# specify number of tasks
#SBATCH --ntasks=8
# or
#SBATCH -n 8
# specify memory required per node. In this case 10MB is requested
#SBATCH --mem=10
# specify the wall clock time limit for the job. In this case, it is 5 mins
#SBATCH --time=00:05:00
# specify the job name
#SBATCH -J PC-com
# or
#SMATCH --job-name=PC-com
# specify the filename to be used for writing output. You need to put the correct path and the output file name
```

```
#SBATCH -o /home-mscluster/<username>/slurm.%N.%j.out
# specify the filename for stderr
#SBATCH -e /home-mscluster/<username>/slurm.%N.%j.err
# The following are shell commands -- which will be executed according to the settings specified above

mpiexec -n 8 ./mpi_mat_vect_mult 64 64
```

7. You can submit your job on the cluster using command

   ```
   sbatch <jobscript_name>
   ```

   Upon submission, a job ID is returned and the job is spooled for execution.
8. You may also run a job interactively using `srun`. For example,

   ```
   srun -p ha -N4 -n8 mpiexec -n 8 ./mpi_mat_vect_mult 64 64
   ```

   requests 8 processes job using 4 nodes in ha partition. If there are available interactive nodes, job will run immediately. Otherwise, it will wait until there are free nodes to run it.
9. To list information on job queues
   - `squeue -u <username>` — lists all the current jobs for a user
   - `squeue -u <username> -t RUNNING` — lists all running jobs for a user.
   - `squeue -u <username> -t PENDING` — lists all pending jobs for a user.
   - `squeue -u <username> -p batch` — lists all current jobs in the `batch` partition for a user.
10. To cancel a job
    - `scancel <jobid>` — cancels a job with `jobid`.
    - `scancel -u <username>` — cancels all the jobs for a user.
    - `scancel -t PENDING -u <username>` — cancels all the pending jobs for a user.
11. You can get some information on the partitions using `sinfo`.