Gridengine

From reading group / nlp lunch

Contents

Aim

The aim of introducing gridengine is to share the computing resources fairly for users. Currently, new GPU resources are intended to be shared among users. For this purpose, please do not execute the command directly on klytaem or ichi (Please do not ssh to these machines directly and run jobs manually) . We intend for users to run their job using gridengine (i.e. by the command *qsub*).

Configuration of gridengine

An entry node is metamor.cis.uni-muenchen.de (10.153.181.138). Execution hosts are klytaem.cis.uni-muenchen.de (GTX 980Ti x 2), ichi.cis.uni-muenchen.de (GTX 980 Ti x 2), and omega (Tesla20). GPU nodes are only these two at this moment. Note that among various variants of Sun Gridengine, we installed Son of Gridengine. The choice of Son of Gridengine is due to that this is a free software.

There are various other software as well. Leibniz-Rechenzentrum (LRZ) (https://servicedesk.lrz.de, Linux-Cluster) uses SLURM (Quick note on Leibniz Supercomputer Centre (http://nlp.cis.uni-muenchen.de/index.php5/LRZ)). The computer lab at CIS/Informatics at the basement uses Sun Grid Engine SGE6.2u5p2 (This is almost the same as Son of Gridengine.)

kibo:/data/gpu is mounted at /kiboHome on all the machines : metamor, klytaem and ichi. (This is equivalent to {alpha, delta, omega,...}:/mounts/work/gpu) After creating your own directory structures at /kiboHome, you can look up your files in the same manner from these three machines.

Name	IP	RAM	CPU type	CPU rate	Num cores	GPU type	year
metamor	10.153.181.138(v4) 2001:4ca0:4f01:2::138(v6) Freimann	4 GB	Intel(R) Core(TM)2 Duo CPU T5670	1.80GHz	2	None	2008? (old machine)
klytaem	129.187.148.139 Ottingenstr	64 GB	Intel(R) Core(TM) i5-6600K CPU	3.50GHz	4	GTX 980 Ti x 2	2016
ichi	10.153.181.137(v4) 2001:4ca0:4f01:2::137(v6) Freimann	32 GB	Intel(R) Core(TM) i5-6600K CPU	3.50GHz	4	GTX 980 Ti x 2	2016
omega	129.187.148.5 Otinggenstr	256 GB	Intel(R) Xeon(R) E5-2630	2.30GHz	24	Tesla K20Xm	2013

The OS installed on these machines are lubuntu 15.10.

Preparation (Login info)

- metamor can be accessed from 129.187.148.XX(Oettingenstr) and 10.153.181.XX(Freimann) but not from outside (The firewall is set up). (However, in case if you have problems despite that the IP address of your machine is 129.187.148.XX or 10.153.181.XX, could you try to access it from alpha, delta, omega, etc.)
- The account on metamor is independent from {alpha, delta, omega, ...}.cis.uni-muenchen.de machines. The user name will be the same as {alpha, delta, omega} (Password info will be informed soon). Among metamor, klytaem, and ichi, username and password are unique.
- In order to change the password, this will need a ldappassword command.
- Current list of users are as follows:

```
beroth, davidk, ebert, fraser, heike, hs, irina, kann, liane, mhuck, ryan, sascha,
schmid, tamchyna, thomas, tsuyoshi, wenpeng, yadollah.
```

How to use gridengine (Hello World Example)

Log in to metamor.cis.uni-muenchen.de. (You should normally not log in to execution hosts such as klytaem and ichi. See the interactive session for compiling, debugging, and so on. Please use qlogin instead of ssh.). --- Read the section Preparation above. When you run experiments on omega, the results will be written on your kiboHome directory {omega,ichi,metamor,klytaem}:/kiboHome/.

Read the environment variables by executing the following command.

```
metamor% . /usr/lib/gridengine/default/common/settings.sh (bash)
or
metamor% source /usr/lib/gridengine/default/common/settings.csh (csh)
```

Set other environmental variables (PATH of cuda tools) in .bashrc (for bash).

```
export PATH=$PATH:/usr/local/cuda-7.5/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/cuda-7.5/lib64
```

Prepare the file called "test1.sh" on directory /kiboHome/tsuyoshi/helloWorld. Suppose that this is your job you want to execute on GPUs.

```
#!/bin/bash

#$ -N MyProj2-test1
#$ -cwd
#$ -q klytaem.gpu0.q,omega.gpu.q
#$ -l gpu=1
#$ -M tsuyoshi@cis.uni-muenchen.de
#$ -m be
source ${HOME}/.bashrc
echo "hello world" > ./helloWorld.output
```

In first five lines, # does not mean these are commented out but these are necessary for qsub command (Please do not confuse with PBS commands if you know PBS commands). There are other ways to express these as command line options. In this case, this is equivalent to

```
metamor% qsub -N MyProj2-test1 -cwd -l gpu=1 -M tsuyoshi@cis.uni-muenchen.de -m be (job)
```

The first option -N MyProj2-test1 means that the name of this job is MyProj2-test1. This will create MyProj2-test1.eXXX (error file) and MyProj2-test1.oXXX (standard output) in your current directory. These two files contain important run-time information on exec hosts if the job does not run. The second option -cwd means that this job is executed at the directory where the job is qsub-mitted. Most cases this option will be necessary. The option -l specifies the required resources. The execution limit such as h_rt=00:20:00 and the RAM requirement such as h_vmem=32G are the examples of these. Note that klytame has 64G RAM but ichi has 32G RAM. The option -M (combined with the option -m) specifies to inform this address about the starting of the job (in case -m b) and about the finish of the job (in case of -m e). This option -M can be dropped if you do not want to receive a lot of emails.

Note that above submission did not specify the queue, this will invoked on any GPU queue on klytaem and ichi. In case you want to specify the queue by the name of the queue, such as klytaem.gpu0.q, add the following line.

```
#$ -q klytaem.gpu0.q
```

The name of the queues are {klytaem,ichi}.gpu{0,1}.q.

Submit your job by 'qsub' command.

```
metamor% qsub test1.sh
```

Then, you will find the message that your job is submitted. qstat is a command that you can see the status of the jobs on gridengine. Below shows that the job test1.pbs is arrived at the gridengine.

```
metamor% qsub test1.sh
Your job 61 ("test1.sh") has been submitted
metamor% qstat -u "*"
job-ID  prior   name       user         state submit/start at     queue                          slots
  -----------------------------------------------------------------------------------------------------
   50 0.55500 test2.sh   tsuyoshi     r     02/11/2016 19:39:30 ichi.gpu0.q@ichi
   57 0.55500 test2-run1 tsuyoshi     r     02/11/2016 20:26:30 klytaem.gpu0.q@klytaem
```

```
    61 0.00000 test1.sh   tsuyoshi     qw    02/12/2016 18:04:49
```

Although all the executions are taken place on the execution nodes, you do not need to care about where the results are returned. This is since the directory structure of /kiboHome/<user name>/... is the same for metamor, klytaem and ichi (unless your job submission is from somewhere different). The results of the job will be stored in the directory that is intended to be in your job submission.

```
metamor% qstat -j 50
```

```
==============================================================
job_number:                 249
exec_file:                  job_scripts/249
submission_time:            Mon Mar  7 11:37:42 2016
owner:                      tsuyoshi
uid:                        31232
group:                      cisintern
gid:                        30001
sge_o_home:                 /kiboHome/tsuyoshi
sge_o_log_name:             tsuyoshi
sge_o_path:                 /usr/lib/gridengine/bin:/usr/lib/gridengine/bin/lx-amd64:.:/usr/local/cuda
sge_o_shell:                /bin/bash
sge_o_workdir:              /kiboHome/tsuyoshi/dl4mt-material-master/session0
sge_o_host:                 metamor
account:                    sge
cwd:                        /kiboHome/tsuyoshi/dl4mt-material-master/session0
hard resource_list:         gpu=1
mail_list:                  tsuyoshi@metamor
notify:                     FALSE
job_name:                   Myproj-test1
jobshare:                   0
hard_queue_list:            ichi.gpu1.q
env_list:                   TERM=NONE
script_file:                test1.sh
binding:                    NONE
job_type:                   NONE
scheduling info:            queue instance "ichi.gpu0.q@ichi" dropped because it is full
                        queue instance "ichi.gpu1.q@ichi" dropped because it is full
                        queue instance "all.q@klytaem" dropped because it is full
                        queue instance "all.q@ichi" dropped because it is full
                        cannot run in queue "klytaem.gpu0.shortq" because it is not contained in its h
                        cannot run in queue "klytaem.gpu1.shortq" because it is not contained in its h
                        cannot run in queue "klytaem.gpu0.q" because it is not contained in its hard q
                        cannot run in queue "klytaem.gpu1.q" because it is not contained in its hard q
                        cannot run in queue "ichi.gpu0.shortq" because it is not contained in its hard
```

By some reason, if you do not want to continue to run your job or if you do not want to wait your job on the queue, you can kill your job by the qdel command. This qdel command will need the job-ID which can be obtained by the qstat command.

```
metamor% qdel 50
```

Useful Commands

'qhost -u "*"' shows the status of all the machines/queues. You can check which ExecHosts are available for your job. Below shows that one job is running on ichi while nobody uses klytaem at this moment (Note that if somebody invokes the command directly not using gridengine this cannot be detected.)

```
tsuyoshi@metamor ~> qhost -u "*"
HOSTNAME                ARCH         NCPU NSOC NCOR NTHR  LOAD  MEMTOT  MEMUSE  SWAPTO  SWAPUS
----------------------------------------------------------------------------------------------
global                  -               -    -    -    -     -       -       -       -       -
ichi                    lx-amd64        4    0    0    0  0.15   31.3G    1.8G   18.6G     0.0
   job-ID  prior   name       user         state submit/start at     queue      master ja-task-ID
   ----------------------------------------------------------------------------------------------
   442     0.55500 s0         tsuyoshi     r     03/26/2016 10:05:11 ichi.gpu.l MASTER
klytaem                 lx-amd64        4    0    0    0  2.65   62.8G    5.1G   22.6G     0.0
metamor                 lx-amd64        2    0    0    0  0.02    3.8G  693.7M    9.3G     0.0
```

'qhost -q' shows all the queues (From time to time, the queue configuration will be changed. Please check it by this command). Several queues may be reserved for particular persons or projects. Currently klytaem.gpu.longq is reserved for Sascha and ichi.gpu.longq is reserved for Tsuyoshi. If those who do not have permission wait at these queues, these jobs will not be executed forever. Please 'qdel' your job.

```
tsuyoshi@metamor ~> qhost -q
HOSTNAME                ARCH         NCPU NSOC NCOR NTHR  LOAD  MEMTOT  MEMUSE  SWAPTO  SWAPUS
----------------------------------------------------------------------------------------------
global                  -               -    -    -    -     -       -       -       -       -
ichi                    lx-amd64        4    0    0    0  0.90   31.3G   10.9G   18.6G     0.0
   ichi.gpu.longq       BIP   0/1/1
   ichi.gpu0.q          BIP   0/0/1
   ichi.gpu1.q          BIP   0/0/1
   all.q                BIP   0/0/0
   ichi.cpu.q           BIP   0/0/1
klytaem                 lx-amd64        4    0    0    0  2.60   62.8G    5.1G   22.6G     0.0
   klytaem.gpu1.q       BIP   0/0/1
   all.q                BIP   0/0/0
   klytaem.gpu.longq    BIP   0/0/1
   klytaem.gpu0.q       BIP   0/0/1
   klytaem.cpu.q        BIP   0/0/1
metamor                 lx-amd64        2    0    0    0  0.01    3.8G  695.1M    9.3G     0.0
```

'qstat -s z -u "*"' shows a long list of job history.

```
tsuyoshi@metamor ~> qstat -s z -u "*"
job-ID  prior   name       user         state submit/start at     queue                          s
---------------------------------------------------------------------------------------------------
    335 0.55500 v21-4-o1   tsuyoshi     z     03/13/2016 15:24:33
    336 0.00000 v21-4-o1   tsuyoshi     z     03/13/2016 15:29:10
    337 0.55500 v21-4-o1   tsuyoshi     z     03/13/2016 15:57:14
    338 0.00000 v21-4-o1   tsuyoshi     z     03/13/2016 16:02:42
```

'qacct -u' shows the cumulative usage of WallClock/CPU/Memory (cisguest1 or cisguest4 are used for the testing purpose by Tsuyoshi).

```
tsuyoshi@metamor ~> qacct -u
OWNER         WALLCLOCK         UTIME         STIME           CPU             MEMORY
===================================================================================================
cisguest1         17334     14222.656      2788.108     17110.334         154015.098             55.9
cisguest4        315919       217.844         5.388    314001.382         585212.621            137.0
sascha           380946        27.072        50.392    365530.698         339548.304            880.5
tsuyoshi        3068843   1070045.784    404962.788   3055146.372        6110034.574           3335.4
```

'qacct -h' shows the summary of utilization of machines.

```
tsuyoshi@metamor ~> qacct -h
```

```
HOST        WALLCLOCK         UTIME         STIME           CPU             MEMORY                 IO
===================================================================================================
ichi          3020684   1065448.380    404216.728   3000635.756        5994938.868           3730.357
klytaem        762357     19064.196      3589.756    751152.058        1193871.727            678.660
metamor             1         0.780         0.192         0.972              0.002              0.003
```

Other environmental variables

Cuda tools are installed at {klytaem,ichi}:/usr/local/cuda-7.5. You will need to set the path of /usr/local/cuda-7.5/bin in your $PATH.

After submitting a job, you should check whether the job is running actually on the GPU mode by the following command on klytaem and ichi (In this case, you are need to ssh to klytaem or ichi. Please do not execute unnecessary commands on klytaem or ichi without submitting a job). At the same time you can get useful information by your error and standard output files corresponding to your job at your current directory.

```
klytaem% nvidia-smi
Mon Mar  7 12:50:45 2016
+------------------------------------------------------+
| NVIDIA-SMI 352.63     Driver Version: 352.63         |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  GeForce GTX 980 Ti  Off  | 0000:01:00.0     Off |                  N/A |
| 44%   52C    P2    87W / 250W |    984MiB /  6143MiB |     75%   E. Process |
+-------------------------------+----------------------+----------------------+
|   1  GeForce GTX 980 Ti  Off  | 0000:02:00.0     Off |                  N/A |
| 38%   27C    P8    13W / 250W |     20MiB /  6143MiB |      0%   E. Process |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID  Type  Process name                               Usage      |
|=============================================================================|
|    0     18523    C   python                                       906MiB |
+-----------------------------------------------------------------------------+
```

For Theano application, please do not specify gpu number but just write 'gpu'.

```
THEANO_FLAGS=device=gpu,floatX=float32,exception_verbosity=high python ./train.py
```

Software Installed on Exec Hosts (klytaem and ichi)

(If you want to use some software which is not installed on klytaem or ichi, please email tsuyoshi@cis.lmu.de. Some software will take time to install it by the user mode.)

Each exec hosts need to install the software. If you want to run the software on klytaem, klytame requires that the corresponding software is ready. If it is ichi, ichi requires the corresponding software.

```
* Theano, blocks
* torch (/usr/local/torch/install/bin)
* tensorflow
* caffe (only ichi)
```

GPU mode should work for Theano and torch but not for tensorflow (As of now 7/March/2016 tensorflow supports Cuda-7.0 but not Cuda-7.5. One way would be to downgrade to Cuda-7.0 but will require the downgrade of Ubuntu as well. For the moment, I think it is better to wait until tensorflow supports Cuda-7.5 for a couple of months).

gcc/g++ 4.9 and 5 is installed on klytaem and ichi, which can be selected. (Theano requires 4.X in its Cuda compiler. Hence, gcc 4.9 will be necessary for Theano-GPU users).

```
ichi% update-alternatives --config gcc
There are 2 choices for the alternative gcc (providing /usr/bin/gcc).
  Selection    Path             Priority   Status
------------------------------------------------------------
  0            /usr/bin/gcc-5     20        auto mode
* 1            /usr/bin/gcc-4.9   10        manual mode
  2            /usr/bin/gcc-5     20        manual mode
Press <enter> to keep the current choice[*], or type selection number:
```

Other software installed on klytaem and ichi can be checked by the commands (lubuntu 15.10):

```
  klytaem% dpkg -l
```

The partial lists are as follows:

```
* virtual env
* editor (emacs, gedit and vim)
* gcc, g++, gfortran, autoconf.
* java (/usr/lib/jvm/java-8-openjdk-amd64), ant.
* cuDNN v4 is installed at the same directory with cuda toolkit 7.5 (i.e. /usr/local/cuda-7.5/lib64).
```

```
* postfix (email)
* LDAP (for user account management)
```

Harddisk (/kiboHome/<UserName>

```
* /kiboHome is mounted from kibo. (Files under kibo is back-up-ed).
* there is no quota at this moment.
```

```
* From metamor, ichi, and klytaem, /kiboHome/<Username> is accessed transparently.
Hence, when you execute your job by qsub, you can access the results from other machine
as well so long as you can access the executed directory at kiboHome.
```

FAQ

(1) I cannot reach metamor?

If you cannot reach metamor from your computer, please try to ssh from alpha, delta, sigma,...

(2) ldappasswd fails?

If ldappasswd fails, please try to switch "cisintern" with "student" in the middle of this command.

```
ldappasswd -H ldap://10.153.181.137 -x -D "cn=tsuyoshi,cn=cisintern,
ou=cis,dc=cis,dc=uni-muenchen,dc=de" -W -S
```

(3) qstat does not work?

Please check whether you actually run the following script or not. This script reads the necessary environmental variables especially the binary PATH.

```
metamor% . /usr/lib/gridengine/default/common/settings.sh (bash)
or
metamor% source /usr/lib/gridengine/default/common/settings.csh (csh)
```

(4) If you want to run either klytaem.gpu0.q or klytaem.gpu1.q, you can write this in your job script file as the following. Note that -q is the definition of queue.

```
#$ -q klytaem.gpu0.q,klytaem.gpu1.q
```

Current Configuration of Queues

Configuration of queues will be changed from time to time. Such information will be posted here (or check the current status of queues/hosts by 'qhost -u "*"').

Although the (logical) queue definition may be more than two, the physical GPUs on klytaem and ichi are limited to two. Once two physical GPUs are occupied, the job scheduler will not be able to allocate more jobs.

Although klytaem.gpu.longq and ichi.gpu.longq are reserved for Sascha and Tsuyoshi, this does not mean that these persons occupy the first GPU all the time. When GPUs are not used, the physical GPU can be used via klytaem.gpu{0,1}.q or ichi.gpu{0,1}.q.

Under the current configuration, all the queues are defined as slot=1. This means that if one queue is occupied by somebody, this queue will not be used by others until the job is finished or killed. Check this by 'qhost -u "*"' whether some queue is occupied or not.

queue	permission	description
omega.cpu.q	CIS	This queue is intended for CPU.
omega.gpu.q	Hinrich and Alex groups	This queue is intended for GPU (Tesla20).
klytaem.gpu.longq	Sascha	This queue does not have constraint, i.e. no time limit (unless you specify the timing constraint on your job script).
ichi.gpu.longq	Tsuyoshi	This queue does not have constraint, i.e. no time limit.
klytaem.gpu0.q	Hinrich and Alex groups	Time constraint is 1 day. If the job is still running after 1 day, the job will be automatically killed instantly by the job scheduler. No memory constraint.
ichi.gpu0.q	Alex group	Time constraint is 1 day. If the job exceeds this time limit, the job will be killed instantly by the job scheduler. No memory constraint.

Note that if you want to run either klytaem.gpu0.q or klytaem.gpu1.q, you can write this in your job script as follows.

```
#$ -q klytaem.gpu0.q,klytaem.gpu1.q
```

Above is in terms of the constraint on the job scheduler. (Under the current configuration, you may omit the request of time. However, this option will be obligation sometime in the future). You can specify your timing constraint of your job in the following manner.

```
#$ -l s_cpu=01:00:00
```

Interactive Session

queue	permission	description
klytaem.cpu.q	Hinrich and Alex group	This queue is for interactive session (compiling, debugging, data preparation, and others). Time constraint is 1 hour.
ichi.cpu.q	Alex group	This queue is for interactive session (compiling, debugging, data preparation, and others). Time constraint is 1 hour.

In interactive session is invoked by 'qlogin command'. Please specify the cpu queue here.

```
metamor% qlogin -q klytaem.cpu.q
metamor% qlogin -q ichi.cpu.q
```

Similarly, if you want to use GPU with interactive session,

```
metamor% qlogin -q klytaem.gpu0.q
metamor% qlogin -q ichi.gpu0.q
```

Parallel Environment (MPI)

Resources

Since there are minor differences please look at the resources on SGE tutorial specified using "Son of Grid Engine 8.1.8".

- https://arc.liv.ac.uk/trac/SGE (Developer of Son of Grid Engine)
- https://www.uibk.ac.at/zid/systeme/hpc-systeme/common/tutorials/sge-howto.html
- http://cbi.utsa.edu/book/export/html/29
- https://newton.utk.edu/bin/view/Main/UsingTheGridEngine

Most reliable information can be also obtained by the man page at metamor (since this is the version installed on metamor, ichi, and klytaem).

```
metamor% man qsub
```

The motivation to introduce gridengine is by the PhD meeting at 28/Jan/2016. Some minimal introduction of gridengine is here.

- Tsuyoshi's presentation slides (3 pages) (http://www.cis.uni-muenchen.de/~tsuyoshi/grid.pdf)

Retrieved from "http://nlp.cis.uni-muenchen.de/index.php5?title=Gridengine&oldid=1007"

- This page was last modified on 8 April 2016, at 15:19.