GHDDI Cluster Manual

luyao.ma@ghddi.org

Thursday 12th December, 2019

1 Release Note

- 2019-12-12 将 Word 版改为 pdf 版本，但未增加实质内容。
- 2019-11-15 增加了一个 sockets 方式编译的 gamess。
- 2019-11-15 Comput34 上的所有 GPU 卡改为共享模式。
- 2019-09-16 安装了 Matlab Runtime，位于/home/soft/matlab/v92。
- 2019-08-17 Zinc 数据库下载完毕。
- 2019-04-16 改进了多核多 GPU 卡任务的提交方法，但原有方法继续有效。
- 2018-12-25 增加了 Desmond GPU 版本。

2 基本情况

- 登录地址是: 111.204.125.107
- 用户名请找马路遥建立之。

3 系统组成

一共有 21 台服务器，登陆节点 1 台；计算节点 16 台；GPU 节点 4 台。comput1～comput16 是 16 个刀片节点。每个节点 24 核。comput31 ～34 是 4 个 GPU 节点。每节点 24 核心 CPU 配 8 张 Nvidia P100 加速卡。

4 作业提交系统 SGE

SGE 系统，又名为 Gridengine。划分为 2 个 queue 分别为 rocket 和 p100 。

parallel 环境有三个，分别为 rocket rocketp p100 。

- 在刀片上提交单节点并行任务，请使用：

  qsub -q rocket -pe rocket 并行数

- 在刀片上提交多节点并行任务，请使用：

  qsub -q rocket -pe rocketp 并行数

- 在 GPU 节点提交单节点并行任务，请使用：

  qsub -q p100 -pe p100 并行数

4.1 GPU Job 提交

使用 GPU 软件，一般应指定 CUDA_VISIBLE_DEVICES 环境变量。请参考这里。上述参考文献中，在 python 中指定使用的设备，GHDDI 环境中，应该由环境变量指定。comput31 ～34，4 台 GPU 服务器，每台 8 卡，CUDA_VISIBLE_DEVICES 的数字为 0-7。

由于大家都要提交作业，所以应该在 SGE 提交系统中指定参数。

SGE 中设定了相应的 GPU 参数，请大家在提交作业的之时，用以下格式的脚本，其中标红处是重点。

```
#$ -l ngpus=X
#$ -pe p100 Y
#$ -q p100
source /home/cudasoft/bin/startcuda.sh
Run Your JOB Here
source /home/cudasoft/bin/end_cuda.sh
```

其中 ngpus=X 处，X 定义使用 gpu 卡的个数。可定义的范围是 1-8。必须在提交脚本中定义。否者多人的 GPU 任务可能会共用同一张卡，互相拖累速度。

SGE 提交脚本中，写为 #$ -l ngpus=5 和在命令行执行 qsub -l ngpus=5 相同。

startcuda.sh 中会先 sleep 35 ，用于核实具体设备，停顿 35 秒。所以当 qstat 命令看到已经开始执行 job 时，还要等待 35 秒才会真正执行。

最后一句 end_cuda.sh,会释放刚刚使用完毕的 CUDA_VISIBLE_DEVICES。

使用多个 CPU 的 -pe 参数，与 -l ngpus=X 参数可同时使用。

同时使用-l ngpus=X 和 -pe p100 Y 参数时，SGE 会分配（X 乘以 Y）个 GPU 。

由于 comput31 ～34 都是 24 核 8 卡，所以：

- X 必须小于等于 8
- Y 必须小于等于 24
- (X 乘以 Y) 也必须小于 8

可以在脚本中使用 $CUDA_VISIBLE_DEVICES 来查看 SGE 系统分配的 GPU 卡编号，使用 $NSLOTS 查看 SGE 系统分配的 CPU 个数。

此方式可以在单机上执行，如果执行跨节点并行，跨节点使用 GPU，请和管理员，也就是老马协商。

Comput34 上，所有卡都设置为共享模式，其中第 7 块不会被 SGE 分配任务，大家留作调试用途。GPU 卡因为是共享，可能出现显存不足等问题，请大家自行注意之。

5 已安装软件

未说明安装位置的，可以直接运行。/home/cudasoft 下的软件，只能在 GPU 节点运行。

5.1 /home/soft 安装的软件

软件名称和版本	安装位置	备注
gamess/2018aug02	new_gamess/2018aug02/	gamess.00.x 是 OpenMPI 编译的版本。
gamess/2018aug02	new_gamess/2018aug02/	gamess.01.x 是 sockets 方式编译的版本。
gamess/2019june30	new_gamess/2019june30/	gamess.00.x 是 OpenMPI 编译的版本。
gamess/2019june30	new_gamess/2019june30/	gamess.01.x 是 sockets 方式编译的版本。
Amber18	amber18/	使用前执行 source amber.sh 以获得正确的环境变量。包含非并行版、mpi 并行版和 cuda 版。
dock 3.7	dock64	
gromacs 2018.1	gromacs/	mpi 版，无 cuda 支持。
Desmond 3.6.1	desmond361/	mpi 版，无 cuda 支持。
UCSF Chimera 1.14	UCSF/Chimera64-1.14/	
Matlab Runtime v92	matlab/	需要将 LD_LIBRARY_PATH 设置中加上 v92 中的 runtime/glnxa64 bin/glnxa64 sys/os/glnxa64

5.2 /home/cudasoft 下安装的软件

软件名称和版本	安装位置	备注
Namd 2.12	namd212/	

5.3 其他已经安装的软件

默认在可执行文件 /usr/bin 下，具体可用 rpm -ql SOFT_NAME 命令查询。

- gromacs-2018.2-1.el7.x86_64
- coot-0.8.1-1.el7.x86_64

- vmd-1.9.3-3.el7.x86_64
- autoDock-4.2.5.1-1.184.x86_64
- R-3.6.0-1.el7.x86_64

6 安装的 Zinc 数据库

位于 /data01/zinc 下，数据库很大，有 42 TB 之多。所以切勿将任何文件 Copy 到其他路径下，解压使用后请立刻删除。

7 SGE 提交范例

老马准备了一些 SGE 提交的范例，在 /home/ghddi_public/bench，每个目录是一个范例，默认是进入目录后执行 qsub ./run.sh 。

每个例子解释如下：

7.1 desmond_one_node

这是个 Desmond 3.6.11 的并行算例，由于不会改输入文件，只能以 2/4/8/16 在单机并行。qsub ./run.sh 默认是一个 16 核并行任务。

7.2 gromacs_multi_node_mpi

192 核跨节点并行的 gromacs 算例。

7.3 mpitest_multi_node

也是个跨节点并行的简单 mpi 程序。

7.4 mpitest_one_nodea

单节点并行的简单 mpi 程序。

7.5 multi_gpu_one_node

单节点多 GPU 卡并行的程序。执行./sub.sh 后，会提交 8 个从 1 卡到 8 卡的 GPU 并行测试程序。该程序会自动侦测出全部 GPU 卡，但根据 SGE 给出的 CUDA_VISIBLE_DEVICES 变量，选择 GPU 卡运行。

7.6 openmm_one_node_gpu

OpenMM 算例，单机单卡使用。

8 关于各种 Python 版本

系统默认 Python 是 python2.7 也就是 /usr/bin/python 或 /usr/bin/python2.7 。

8.1 python3.6

需要执行:

export LD_LIBRARY_PATH=/home/soft/rdkit_2018_03_1/lib:$LD_LIBRARY_PATH 之后，才可以使用 rdkit 模块

8.2 /home/cudasoft/python349

支持 cuda,只能在 comput31 ～34 4 个节点上运行。使用前请执行:source /home/cudasoft/python349/env.sh

有安装 deepchem、tensorflow 等模块，支持 GPU 。