常见问题解答 (II)

报告人：王文昊

北京航空航天大学信息办高性能运算中心

2021.11.18

目录

1 重点问题回顾

2 GPU问题解答

3 Nvidia Ampere系列

Part 1

重点问题回顾

本部分将回顾**上周重要**的通用问题。

通用问题解答

Q1：账户被锁，无法登陆高性能运算平台

A1：首先尝试全部节点ln01、ln02、ln03，ip分别为10.212.66.4、10.212.66.5、10.212.70.128；如果均无法连接，则在用户交流群中提供用户名以及用户IP联系管理员王玉龙协助解锁。

~~目前安全政策是：如果一个用户在某一个节点输入一次错误密码，则该用户的账户以及该用户的IP均会被锁死；并且不会自动解锁。三个节点锁定机制独立。~~

目前安全政策是： 3次密码输错就锁用户，5分钟后用户自动解锁，10次错误锁用户的IP。

所以建议大家将比较复杂的密码通过复制粘贴输入，并且尽量不要多人使用同一账户。

如密码忘记，添加管理员王玉龙微信来重置密码。

通用问题解答

Q2：网路无法连接到任意登陆节点

A2：如果是在校内，检查是否使用校园网，如果使用的是校园网，检查是否使用网络路由器并是否设置正确。如所有设置均无问题，可以使用VPN暂时应急，并联系信息化办公室老师解决网络问题。

如果在校外，检查VPN是否登陆正常。如果登陆正常仍无法连接，退出并重新连接。

通用问题解答

Q3：无法在登陆节点连接网络

A3：目前网络政策为ln01、ln03节点可以连接网络，ln02不能，所以不要在ln02上进行联网下载。

Q4：下载配置环境过程中终端自动断开

A4：目前ln01节点执行的系统设定阈值较为严格，如在ln01节点下载配置环境时总被杀死，则选择在ln03节点进行下载配置环境。但，首先应该使用ln01节点。

如果程序不需要网络下载，只需要编译，应提交作业到计算节点进行编译。

通用问题解答

Q5：新手从未使用过linux和slurm系统

A5：一定不要直接上手HPC平台，首先阅读用户手册，并在网络查找学习linux系统相关基础命令，如cp、ls、mkdir等。

Q6：本地和高性能运算平台传输数据

A6：校内快速传输数据采用scp命令 (100M/s)、支持断点续传rsync命令，举例如下：

scp -r /Volumes/HDD/dy wangwh@10.212.70.128:/gs/home/wangwh

rsync -avzP -r /Volumes/HDD/dy wangwh@10.212.70.128:/gs/home/wangwh

如果数据量大，而且又有急用，可以联系王玉龙，把硬盘寄给他拷贝数据

校外高速传输数据可以采用阿里云的ossutil64 (100 ~ 500M/s)，具体参考相关阿里云实例。

通用问题解答

Q7：需要root权限来辅助安装程序

A7：root权限在任何情况下都不会提供给用户。目前HPC平台绝大部分环境依赖已经安装完毕，如缺少特定依赖可在群中提出。另外一个解决办法为在家目录下安装相关环境，具体方法参考覃波的PPT。

Q8：安装特定的软件

A8：开源软件安装方法参考覃波的PPT。对于商业软件，如果有正版软件许可，可以联系管理员进行安装；注意：平台不建议用户安装破解或者盗版软件，用户须自行负责相关软件的版权问题。

通用问题解答

Q9：平台环境依赖版本过低

A9：出于系统稳定性，平台的基本编译器和库文件版本较低。可以通过module available命令查看相关依赖。加载使用时，如：module load gnu8/8.3.0得到8.3.0的gcc。

如所需依赖平台未提供，可以在家目录下安装，安装方法参考覃波的PPT。

```
------------------------------------------ /opt/ohpc/pub/modulefiles ------------------------------------------
   EasyBuild/3.9.2          autotools            cuda/11.4          hwloc/2.0.3              matlab/R2019b
   OpenFOAM/OpenFOAM-6      cmake/3.14.3         gnu/5.3.0          intel/18.0.3.222         petsc/3.11.4
   anaconda2/2019.10        cuda/8.0             gnu7/7.3.0         intel/19.0.5.281 (D)     pmix/2.2.2
   anaconda3/2019.10        cuda/9.0             gnu8/8.3.0         jdk/1.8.0_201            prun/1.3
   ansys/v192               cuda/10.1    (D)     gromacs/2019.4     lammps/20191030          valgrind/3.15.0

  Where:
   D:  Default Module

Use "module spider" to find all possible modules.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".
```

Q10：For循环多核心并行

A10：我们有时面对如下情景：

```
for i in range(0,100_000):
    ……
```

如直接运行会只使用单核心，可以将其拆分为20个部分，使用20个核心。

平台提供了Parallel软件，使用如下：

```
parallel -vk ::: \
'python test.py --num 0' \
'python test.py --num 1'
```

这是使用两个核心的例子；以此类推，可以使用20个、甚至更多的核心。

Part 2

GPU问题解答

本部分将对**手册中**提及以及**近期各位同学问到**还有**我发现**的GPU问题进行解答。

GPU问题解答

Q1：配置自定义Anaconda环境

A1：我们这里介绍一种超简便，并且拥有极高可扩展性的安装anaconda的方法。

1. 下载anaconda包：

wget https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/Anaconda3-2019.10-Linux-x86_64.sh --no-check-certificate

2. 增加执行权限：

chmod +x Anaconda3-2019.10-Linux-x86_64.sh

3. 安装 (按照提示操作)：

bash Anaconda3-2019.10-Linux-x86_64.sh

4. 加载环境变量：

source ~/.bashrc

GPU问题解答

Q2：用自定义Anaconda环境安装库

A2：首先我们先配置pip清华源：

pip config set global.index-url https://pypi.tuna.tsinghua.edu.cn/simple

安装PyTorch只需要：

V100: conda install pytorch==1.7.1 torchvision==0.8.2 cudatoolkit=10.1 -c pytorch

A100: conda install pytorch==1.8.0 torchvision==0.9.0 torchaudio==0.8.0 cudatoolkit=11.1 -c pytorch -c conda-forge

安装其他包，如numpy：pip install numpy

创建虚拟环境，如：conda create -n test python=3.7

其他复杂的库参考《GPU环境配置-1》、《GPU环境配置-2》。

GPU问题解答

Q3：在自定义conda环境中安装的库无法找到

A3：首先确定是否的确安装：

在登陆节点 conda activate XXX, python, import xxx

确定无误后：

提交作业时需要退出虚拟环境，在base环境提交：

conda activate XXX

python xxx.py

GPU问题解答

Q4：运行Python程序时无法实时输出

A4：实时输出有两种办法：

(1) 使用print实时输出，在print函数里面加上“, flush=True”参数；

(2) 自定义写log的方法或者安装库loguru，将log文件写入硬盘中。

Q5：在GPU分区无法调用cuda

A5：因为版本问题，cuda在任何地方均未默认加载。可以采用如下方式加载：

(1) 加载平台提供的cuda，如：module load cuda/10.1；

(2) 使用自己的特定版本的cuda，参考《GPU环境配置-1》。

GPU问题解答

Q6：无法调用GPU

A6：可能有多种原因无法使用GPU：

(1) 没有正确安装库，如安装的是CPU版本的PyTorch、不兼容的cuda版本的PyTorch；

(2) 检查是否申请了GPU，即是否有：#SBATCH --gres=gpu: 1；

(3) 个别节点的个别GPU可能会故障。

Q7：安装特定版本的CUDA、CuDNN、cmake、gcc

A7：参考《GPU环境配置-1》。

GPU问题解答

Q8：在一个GPU上同时跑多个程序

A8：由于各种原因，很多时候GPU利用率较低，所以在一个GPU上跑多个程序是一个提高利用率的方法。

我们在每个计算节点上均安装了parallel程序，该程序的一个作用就是可以让一个GPU同时运行两个及以上的程序。

申请一个GPU，在脚本中写入：

```
parallel -vk  ::: \
'python train.py --num 0' \
'python train.py --num 1'
```

即可满足同时在一张卡上跑多个程序的目标，这样可以节约资源，提高GPU的利用率。

GPU问题解答

Q9：PyTorch or Tensorflow?

A9：在HPC平台上利用PyTorch调用GPU资源至少有以下的优点：

1. 调用单卡、多卡GPU简单方便；
2. 语法简洁，上手容易；
3. 社区友好且强大，遇到bug总能找到相关的解决办法；
4. 文档规范，有官方介绍文档，方便查找学习；
5. 资源多，学术界大部分开源代码均用PyTorch实现；
6. ***遇到问题时，可以向我咨询讨论，有技术支持。***

GPU问题解答

Q10：安装需要编译的GPU库，如：Apex

A10：参考《GPU环境配置-2》。

Q11：从CPU迁移到GPU的优势

A11：使用GPU做运算的具体优势：

1. 运算速度极快：运算速度一般为CPU的10～20倍；
2. 价格低廉：同等算力所需要的GPU的价格远低于所需的CPU的价格；
3. 使用方便：随着深度学习开源框架的兴起，如使用PyTorch调用GPU，只需要.cuda()即可。

GPU问题解答

Q12：在Linux系统调用GPU的优势

A12：使用Linux系统进行GPU调用、深度学习的优点是：

1. 运算效率很高：相较于Windows，相同资源情况下，运算速度可以快1倍左右；
2. 操作方便简洁：可以完全使用命令行进行操作，完全避免了Windows的复杂图形界面；
3. 更适合科研：科研所依赖的环境均可在linux下得到完美支持，相反Windows经常报错。

Q13：配置GPU环境是否需要申请GPU

A13：绝大部分配置GPU环境不需要申请GPU，极其个别时候编译需要GPU参与。

GPU问题解答

Q14：显存及内存使用优化

A14：目前，北航HPC平台配置的V100显卡均为32G显存、每个GPU节点拥有384G内存；以上配置可以满足我们的绝大部分应用。

如果仍出现显存/内存不足的问题，应考虑以下方面：

1. 是否设置的batch size超过合理范围，可以适当调小batch size并调整相关学习率，或调用多卡GPU并行；
2. 需要计算一个batch时，才应该把该batch读入内存；避免把全部数据集读入内存；
3. 可以考虑使用混合精度计算代替原有的32位精度，可以降低显存、内存压力。

GPU问题解答

Q15：运算效率低的问题

A15：由于北航HPC为大规模集群，采用的是存储节点和计算节点分开的设计方式，所以从存储节点读取数据再传输到计算节点运算存在时间延迟。

该时间延迟和读取数据的大小有很多关系，实际测试发现，当在GPU计算时需要频繁读入大量小文件时，延迟现象明显；相反，如果只是读入大文件，延迟现象可以忽略不计。

对于读取数据时间过长，可能的解决办法：

1. 尽量读取大文件，避免频繁读取小文件；
2. 使用多线程的方法读入文件；
3. 多卡并行时，采用分布式训练的方式；
4. 如必须读取小文件，对小文件进行打包整合，而不是使用原有的格式，如jpg.

GPU问题解答

Q16: 在登陆节点安装PyTorch、Faiss等需要cuda的库时，报环境冲突错误，提示：无法找到对应的driver，如下：

```
(demo_1104) [statchao@ln03 ~]$ conda install -c conda-forge faiss-gpu cudatoolkit=11.1
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: failed with repodata from current_repodata.json, will retry with next repodata source.
Collecting package metadata (repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: /
Found conflicts! Looking for incompatible packages.
This can take several minutes.  Press CTRL-C to abort.
failed

UnsatisfiableError: The following specifications were found to be incompatible with each other:
CUDA driver:

  - cudatoolkit=11.1 -> __cuda[version='>=11.1']

Your installed CUDA driver is: not available
```

A16：这其实是conda版本不对，升级即可：conda install conda==4.10.3

GPU问题解答

Q17：使用conda安装相关包时，下载完毕但验证无法通过，如下：

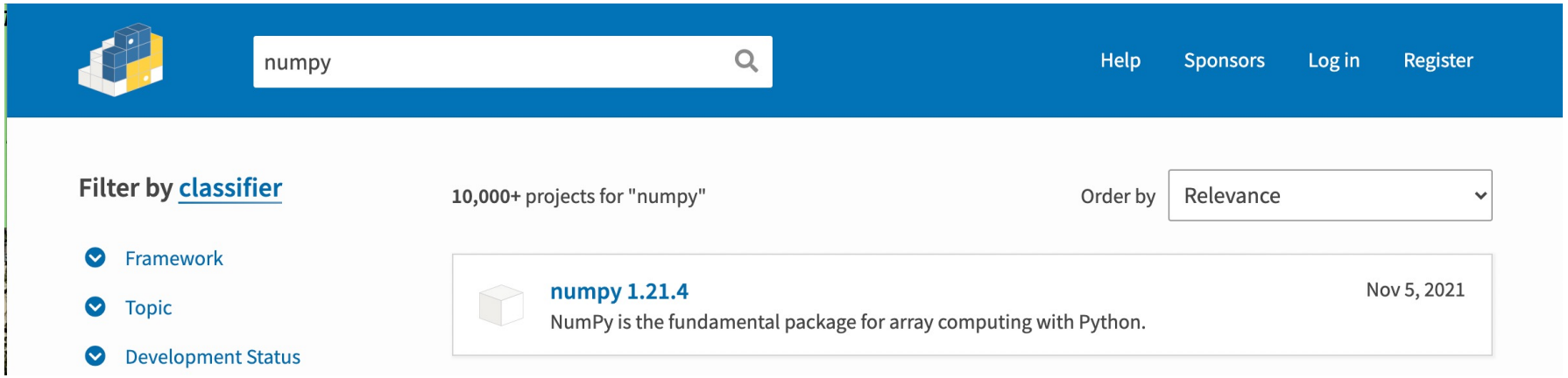
```
Preparing transaction: done
Verifying transaction: -
SafetyError: The package for python located at /home/wangwenhao/anaconda3/pkgs/python-3.7.11-h12debd9_0
appears to be corrupted. The path 'lib/python3.7/__pycache__/_sysconfigdata_m_linux_x86_64-linux-gnu.cpython-37.pyc'
has an incorrect size.
  reported size: 18159 bytes
  actual size: 18336 bytes
```

A17：找到出现问题的文件，如/home/wangwenhao/anaconda3/pkgs/python-3.7.11-h12debd9_0；删除之，并重新运行conda install …

GPU问题解答

Q18：离线pip安装包

A18：从https://pypi.org/搜索并下载相关对应的whl文件到本地并将该whl文件上传到HPC平台。



如：numpy-1.21.4-cp37-cp37m-manylinux_2_12_x86_64.manylinux2010_x86_64.whl

安装：

pip install numpy-1.21.4-cp37-cp37m-manylinux_2_12_x86_64.manylinux2010_x86_64.whl

Q19：离线建立Anaconda虚拟环境

A19：在无网络的情况下，可以将其他linux系统创建的环境复制到HPC平台上，具体如下：

在其他linux系统创建的环境在~/anaconda3/envs/下，找到所需要的那个压缩、上传、解压缩到HPC平台对应的~/anaconda3/envs/下。

conda activate XXX即可在HPC上使用。

如果后续有网络了，想更新这个虚拟环境内的内容，需要修改：

~/anaconda3/envs/XXX/bin/pip这里的python路径

GPU问题解答

Q20：CUDA兼容问题，报错：

```
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/gs/home/wangwh/anaconda3/envs/pytorch_cuda11/lib/python3.7/site-packages/torch/cuda/__init__.py", line 172, in _la
zy_init
    torch._C._cuda_init()
RuntimeError: CUDA driver initialization failed, you might not have a CUDA gpu.
```

```
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/gs/home/statchao/anaconda3/envs/test_cuda102/lib/python3.7/site-packages/torch/cuda/__init__.py", line 214, in _la
zy_init
    torch._C._cuda_init()
RuntimeError: The NVIDIA driver on your system is too old (found version 10010). Please update your GPU driver by downloadi
ng and installing a new version from the URL: http://www.nvidia.com/Download/index.aspx Alternatively, go to: https://pytor
ch.org to install a PyTorch version that has been compiled with your version of the CUDA driver.
```

A20：目前HPC平台V100计算节点驱动最高支持cuda10.1，应安装相应版本的PyTorch等；未来上线的A100计算节点仅支持cuda11及以上，应安装相应版本的PyTorch等。

Part 3

Nvidia Ampere系列

Nvidia公司于2020年5月正式发布了Ampere架构，A100 40G计算卡作为基于该架构的首款GPU，将在我们HPC平台三期上线。

随着时间的流逝，基于Ampere架构的专业计算卡不断丰富，同时也出现了使用Ampere架构的游戏卡。

我们将在下次报告中系统介绍Nvidia Ampere系列的全部专业卡、游戏卡。这不仅方便大家在未来使用HPC平台，同时也给各实验室/个人使用基于Ampere架构的显卡提供参考。

THANKS FOR LISTENING

北京航空航天大学信息办高性能运算中心

2021.11.18