

Ubuntu 运行 AMSS-NCKU-Python 报错信息及处理办法

(不定期更新中)

报错类型一：导入输入文件报错

报错信息

```
import NR_xiaoqu_input
```

报错原因

这是因为用了老的输入文件名 NR_xiaoqu_input.py，没有更新，新的输入文件名为 AMSS_NCKU_Input.py

应对措施

用户可自行更改为

```
import NR_xiaoqu_input ---> AMSS_NCKU_Input.py
```

报错类型二：MPI 运行 ABE 报错

报错信息一

There are not enough slots available in the system to satisfy the 64 slots that were requested by the application: ABE

Either request fewer slots for your application, or make more slots available for use.

报错原因

用户电脑没有 64 个 CPU 核心，却使用 MPI 进行 64 核心进行并行计算

应对措施

在输入文件 AMSS_NCKU_Input.py 中设定 MPI 并行计算的进程数

MPI_processes = 32 ---> MPI_processes = 8

根据自行电脑的 CPU 核心数设定

```
12 #####
13
14 ## 设置程序运行目录和计算资源
15
16 File_directionary = "xiaoqu_Results_BBH_test" ## 程序运行目录
17 Output_directionary = "output_file" ## 存放二进制数据的子目录
18 MPI_processes = 64 ## 想要调用的进程数目
19
20 GPU_Calculation = "no" ## 是否开启 GPU 计算, 可选 yes 或 no
21 CPU_Part = 0.5
22 GPU_Part = 0.5
23
24 #####
--
```

报错信息二

未能找到 ABE 生成的文件（包括但不限于 setting.par、Error.log、bssn_BH.dat、bssn_ADMQs.dat、bssn_psi4.dat、bssn_constraint.dat）

No such file or dictionary:

```
Output directory: xiaoqu_Results_GW150914_test/AMSS_NCKU_output
Traceback (most recent call last):
  File "/amssncku-master--bbh/AMSS_NCKU_Program.py", line 351, in <module>
    shutil.copy( AMSS_NCKU_error_file_path, os.path.join(output_directory, "AMSSNCKU_setting_parameter"
) )
  File "/usr/lib/python3.10/shutil.py", line 417, in copy
    copyfile(src, dst, follow_symlinks=follow_symlinks)
  File "/usr/lib/python3.10/shutil.py", line 254, in copyfile
    with open(src, 'rb') as fsrc:
FileNotFoundError: [Errno 2] No such file or directory: 'xiaoqu_Results_GW150914_test/AMSS_NCKU_output/out
put_file/setting.par'
```

报错原因

有可能是 ABE（或 ABEGPU）程序运行异常，可进入 AMSS_NCKU_output 文件夹下查看 ABE_out.log（或 ABEGPU_out.log）中的报错信息

有一种报错是如下的行为

```
xiaoqu_Results_GW150914_test > AMSS_NCKU_output > ≡ ABE_out.log
1 -----
2 mpirun has detected an attempt to run as root.
3
4 Running as root is *strongly* discouraged as any mistake (e.g., in
5 defining TMPDIR) or bug can result in catastrophic damage to the OS
6 file system, leaving your system in an unusable state.
7
8 We strongly suggest that you run mpirun as a non-root user.
9
10 You can override this protection by adding the --allow-run-as-root option
11 to the cmd line or by setting two environment variables in the following way:
12 the variable OMPI_ALLOW_RUN_AS_ROOT=1 to indicate the desire to override this
13 protection, and OMPI_ALLOW_RUN_AS_ROOT_CONFIRM=1 to confirm the choice and
14 add one more layer of certainty that you want to do so.
15 We reiterate our advice against doing so - please proceed at your own risk.
16 -----
17
```

表示计算机阻止在 root 用户下运行 mpirun

应对措施

方案一：在其它（非 root）用户下使用程序

方案二：修改 python 脚本运行 ABE/ABEGPU 程序的命令（mpirun -np XXX ABE 或 mpirun -np XXX ABEGPU），后面补上 --allow-run-as-root 这个命令在 makefile_and_run.py 中的 run_ABE 函数中

```

94 #####
95
96 ## 这个函数运行 AMSS-NCKU 主程序 ABE
97
98 def run_ABE():
99
100     print()
101     print(" 正在运行 AMSS-NCKU 主程序 ABE/ABEGPU ")
102     print()
103
104     ## 定义要运行的命令, 要使用 str 将其它转换为字符串
105
106     if (input_data.GPU_Calculation == "no"):
107         mpi_command = "mpirun -np " + str(input_data.MPI_processes) + " ABE"
108         mpi_command_outfile = "ABE_out.log"
109     elif (input_data.GPU_Calculation == "yes"):
110         mpi_command = "mpirun -np " + str(input_data.MPI_processes) + " ABEGPU"
111         mpi_command_outfile = "ABEGPU_out.log"
112
113     ## 使用 subprocess.Popen 来执行命令, 并实时打印输出
114     mpi_process = subprocess.Popen(mpi_command, shell=True, stdout=subprocess.PIPE, stderr=subprocess.STDOUT, text=True)
115
116     ## 将 ABE 的运行结果写入文件中
117     with open(mpi_command_outfile, 'w') as file0:
118         ## 循环读取输出并打印
119         for line in mpi_process.stdout:
120             print(line, end='') # 实时打印输出
121             file0.write(line) # 将行写入文件
122             file0.flush() # 确保每行都被立即写入文件, 可选
123     file0.close()
124
125     ## 等待进程结束
126     mpi_return_code = mpi_process.wait()

```

如有新的报错, 请联系我, 邮箱 chenkaiqiao@cqut.edu.cn