

“太乙”

HiQ

手

册

编写时间：2020 年 11 月

一、 介绍.....	3
1.1. 简介.....	3
1.2. 太乙中的HiQ.....	3
二、 使用软件.....	3
2.1. 输入文件样例.....	3
2.2. 修改输入文件.....	3
三、 作业样例.....	3
3.1. 提交脚本.....	4
3.2. 提交并查看作业.....	6
3.3. 作业结果.....	6
四、 注意事项.....	6
4.1. 关于测试样例.....	6
4.2. 联系方式.....	6

## 一、介绍

### 1.1. 简介

华为量子计算软件HiQ目前包含一套从量子线路图形和程序输入，到编译、模拟的整套量子计算编程框架。HiQ基于并兼容ProjectQ。目前HiQ支持多硬件后端和多API前端扩展，并在华为云上提供HiQ量子计算模拟器仿真平台服务。其中华为云上的模拟器前端包括量子线路编程API、经典-量子混合编程BlockUI以及量子线路编排GUI；后端兼容ProjectQ API，可以支持编译成多种量子计算芯片指令集接口语言、提供华为云上的高性能C++并行和分布式量子计算模拟服务等。HiQ量子计算模拟平台（simulator），从功能上又包括单幅度模拟器、全幅度模拟器，以及量子纠错线路模拟器等。

更多内容，请参考官方网站：<https://hiq.huaweicloud.com>。

### 1.2. 太乙中的HiQ

太乙集群上面已经测试可以在用户环境下安装配置HiQ Simulator软件。安装脚本所在位置：`/data/ccse-tangh/scripts/install-hiq.sh`。

用户可以在自己的家目录创建一个文件夹，复制安装脚本并安装程序

```
mkdir -p ~/softwares/hiq/  
cp /data/ccse-tangh/scripts/install-hiq.sh ~/softwares/hiq/  
cd ~/softwares/hiq/ && nohup sh install-hiq.sh &
```

## 二、使用软件

### 2.1. 输入文件样例

输入样例文件位置：`~/softwares/hiq/libraries_source/random_number.py`，用户可进入此目录，进行测试。`~/softwares/hiq/HiQsimulator/examples`目录中也有例子。

### 2.2. 修改输入文件

太乙中当前版本HiQ 软件需要将测试的输入文件进行修改才能正常运行测试。

```
# 替换当前文件夹输入文件中的所有 “from hiq.projectq” 为 “from projectq”  
sed -i 's/hiq.projectq./projectq./g' ./*
```

## 三、作业样例

### 3.1. 提交脚本

安装脚本会生成一个用于太乙lsf的任务脚本：`~/softwares/hiq/hiq.bsub`, 用户可自行复制该文件到自己目录, 进行测试。

脚本内容:

```
#!/bin/bash
#BSUB -q short
#BSUB -n 40
#BSUB -e %J.err
#BSUB -o %J.out
#BSUB -R "span[ptile=40]"
hostfile=`echo $LSB_DJOB_HOSTFILE`
NP=`cat $hostfile | wc -l`
export HIQ_DIR=/data/ccse-tangh/hiq #此文件夹需要修改为实际的安装文件夹
export LIB_DIR=${HIQ_DIR}/libraries
export gcc_DIR=$LIB_DIR/gcc/10.2.0
export ucx_DIR=$LIB_DIR/ucx/1.8.1
export openmpi_DIR=$LIB_DIR/openmpi/4.0.5

export boost_DIR=${LIB_DIR}/boost/1.74.0
export gflags_DIR=${LIB_DIR}/gflags/2.2.2
export glog_DIR=${LIB_DIR}/glog/0.4.0
export hwloc_DIR=${LIB_DIR}/hwloc/2.1.0

export gcc_ROOT=${gcc_DIR}/bin
export gcc_LIBRARY=${gcc_DIR}/lib64

export openmpi_ROOT=${openmpi_DIR}/bin
export openmpi_LIBRARY=${openmpi_DIR}/lib

export boost_ROOT=${boost_DIR}/include/boost
```



```
export boost_LIBRARYDIR=${boost_DIR}/lib
export BOOST_ROOT=${boost_DIR}/include
export BOOST_INCLUDE=${boost_DIR}/include
export BOOST_LIBDIR=${boost_DIR}/lib

export gflags_ROOT=${gflags_DIR}/include
export gflags_LIBRARY=${gflags_DIR}/lib

export glog_ROOT=${glog_DIR}/include
export glog_LIBRARY=${glog_DIR}/lib
export glog_INCLUDE_DIR=${glog_DIR}/include

export hwloc_ROOT=${hwloc_DIR}/include
export hwloc_LIBRARY=${hwloc_DIR}/lib

export PATH="${gcc_ROOT}:${ucx_ROOT}:${openmpi_ROOT}:${cmake_ROOT}:${boost_ROOT}:${PATH}"
export PATH="${hwloc_ROOT}:${glog_ROOT}:${gflags_ROOT}:${PATH}"
export LD_LIBRARY_PATH="${cmake_LIBRARY}:${LD_LIBRARY_PATH}"
export LD_LIBRARY_PATH="${gcc_LIBRARY}:${ucx_LIBRARY}:${openmpi_LIBRARY}:${boost_LIBRARYDIR}:
${LD_LIBRARY_PATH}"
export LD_LIBRARY_PATH="${gflags_LIBRARY}:${glog_LIBRARY}:${hwloc_LIBRARY}:${LD_LIBRARY_PATH}"

export python_DIR=${LIB_DIR}/python/anaconda3/2020.07
export python_ROOT=${python_DIR}/bin
export PATH="${python_ROOT}:${PATH}"
source ${python_DIR}/etc/profile.d/conda.sh
source activate

#并行方式运行，按需修改使用
mpirun -np 40 python3 ${LIB_SOURCE_DIR}/test/random_number.py > $LSB_JOBID.log 2>&1

#单节点，单线程运行
```

```
#python3 ./ser.py > $LSB_JOBID.log 2>&1
```

```
##最后两行运行脚本指令按需更改和调试
```

### 3.2. 提交并查看作业

```
bsub < hiq.bsub
```

```
bjobs
```

### 3.3. 作业结果

结果文件为\$LSB\_JOBID.out

## 四、 注意事项

### 4.1. 关于测试样例

- 本文档测试样例，只是验证改软件能正常运行。
- 欢迎使用该软件的用户，提供更加好的样例，以便提高该软件的使用效率。

### 4.2. 联系方式

- 本文档由科学与工程计算中心编写，联系邮箱：[hpc@sustech.edu.cn](mailto:hpc@sustech.edu.cn)，联系人：谢作扬，电话：0755-88015831