



Quantum Espresso基础使用简介

高性能助管：谢天焯

2023/11/30



高性能助管：郑彩虹

2023/11/16

高性能的、简洁的国内自主研发的
第一性原理商用计算软件

计算固体为主

付费



高性能助管：温馨

2023/11/23

广为人知的第一性原理商用计算
软件

计算分子为主

付费



高性能助管：谢天烨

2023/11/30

拥有大量开发者的第一性原理开
源计算软件

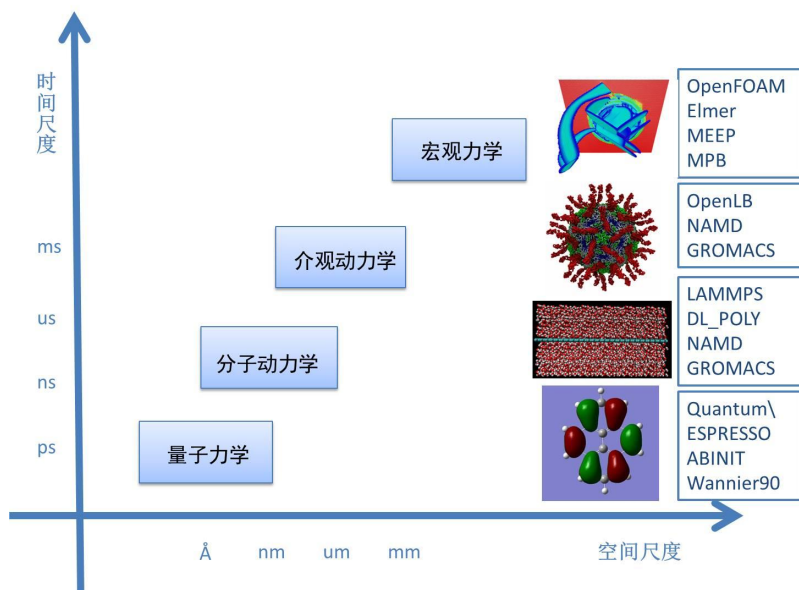
计算固体为主

开源（免费）



一、简介

What is Quantum Espresso?



1. 高效&开源(免费!)的物理材料模拟软件包

2. 使用量子力学方法进行纳米尺度体系的模拟 (~1000电子)

3. 基于密度泛函理论 (DFT) 计算材料电子性质

4. 主要使用赝势 (Pseudopotentials) 和平面波基组 (Plane Waves) 作为其主要计算方法

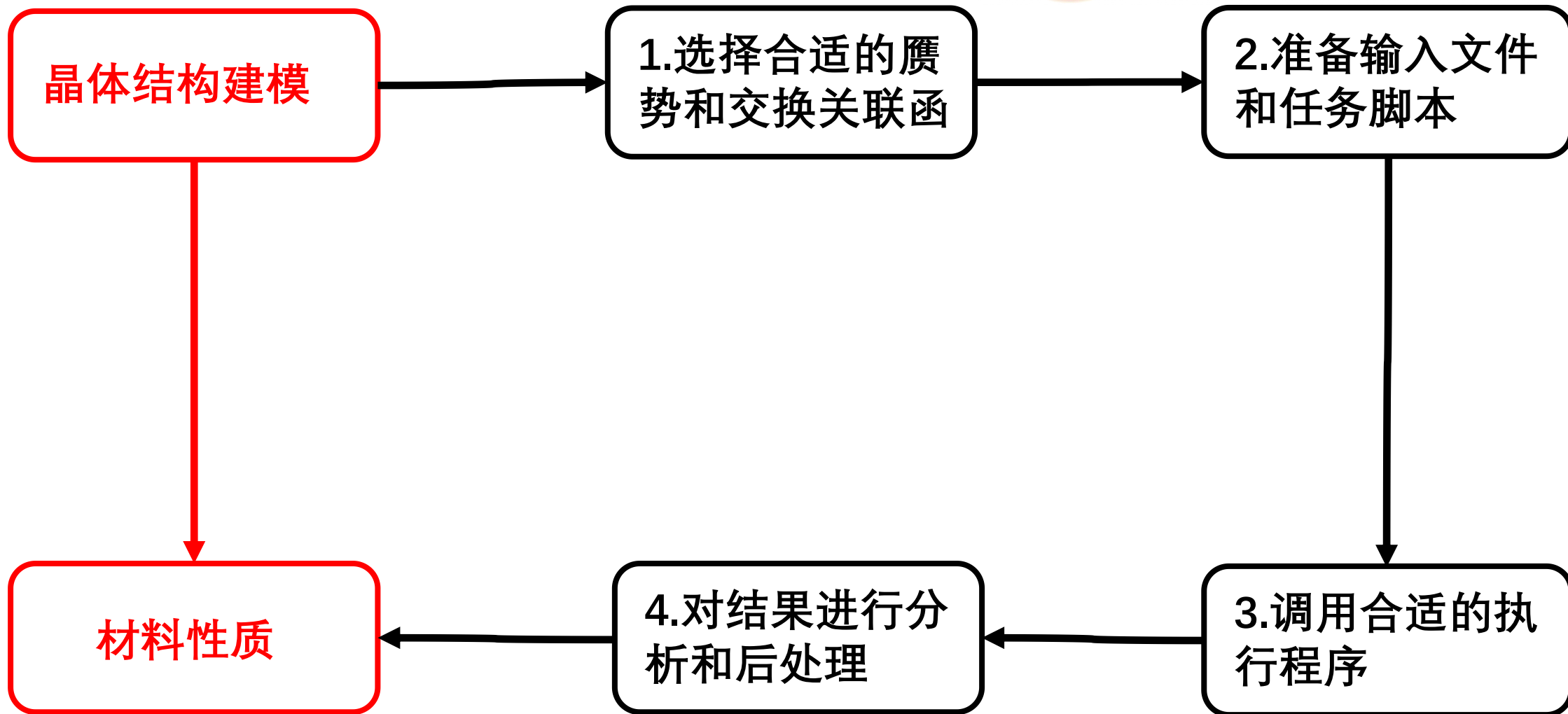
5. 拥有大量开源计算模块: 溶剂化模型environ等

What Can Quantum Espresso Do?



1. 晶体结构计算
2. 电子性质计算: 能带, 态密度
3. 分子动力学计算
4. 光谱性质计算
5. 量子输运等其他计算

How to apply a QE calculation?



数据库

晶体结构建模

建模软件

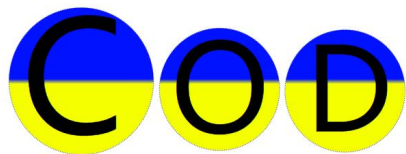
1. Material project (晶体)



<https://legacy.materialsproject.org/>

2. COD (晶体)

Crystallography Open Database



<http://www.crystallography.net/cod/>

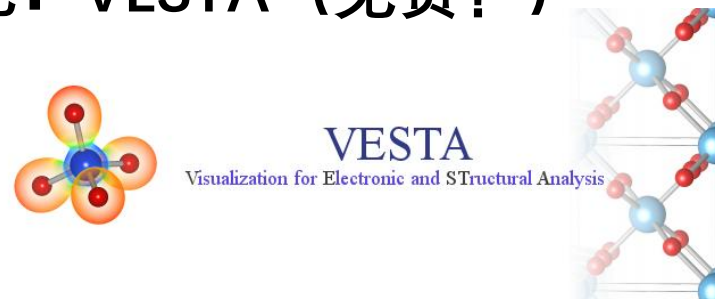
3. PubChem (分子)



PubChem About Docs Submit Contact

<https://pubchem.ncbi.nlm.nih.gov/>

1. 可视化: VESTA (免费!)



<http://jp-minerals.org/vesta/en/download.html>

2. Python库: ASE (开源)

Danmarks Tekniske Universitet - DTU
<https://wiki.fysik.dtu.dk/ase/>

[Atomic Simulation Environment — ASE documentation](https://wiki.fysik.dtu.dk/ase/)

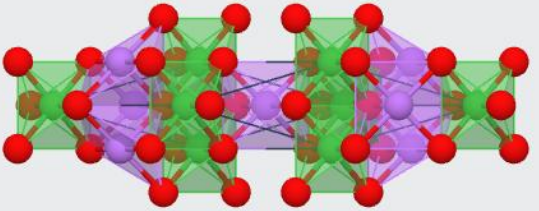
<https://wiki.fysik.dtu.dk/ase/>

3. 可视化: Material Studio (付费)

1.1.1 晶体结构建模： 下载结构

MATERIAL ID: **LiNiO₂** **mp-25411** [Show Help Guides](#)

[Electronic Structure](#) [X-Ray Diffraction](#) [X-Ray Absorption](#) [Substrates](#) [Similar Structures](#) [Synthesis Descriptions](#)
[Calculation Summary](#) [User Contributions](#) [Provenance/Citation](#)



Material Details

Final Magnetic Moment
1.001 μ_B

Magnetic Ordering
FM

Formation Energy / Atom
-1.399 eV

Energy Above Hull / Atom
0.012 eV

Density
4.62 g/cm³

Decomposes To
[LiNiO₂](#)

Band Gap
0.000 eV

Lattice Parameters

a 2.906 Å α 73.383°
b 2.906 Å β 73.383°
c 5.086 Å γ 59.992°
Volume 35.095 Å³

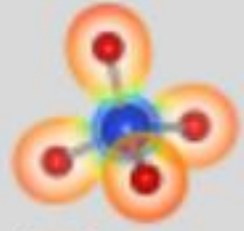
Final Structure [CIF](#)
Fractional Coordinates

Li		
a	b	c
0.5	0.5	0.5

Ni		
a	b	c
1	1	0

Atoms Bonds
 Unit Cell Polyhedra

[CIF](#)

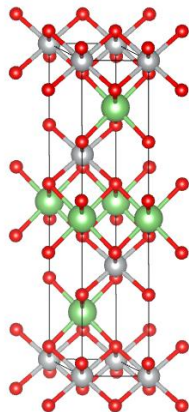
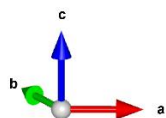


LiNiO₂_m
p-25411_s
ymmetrize
d (2).cif

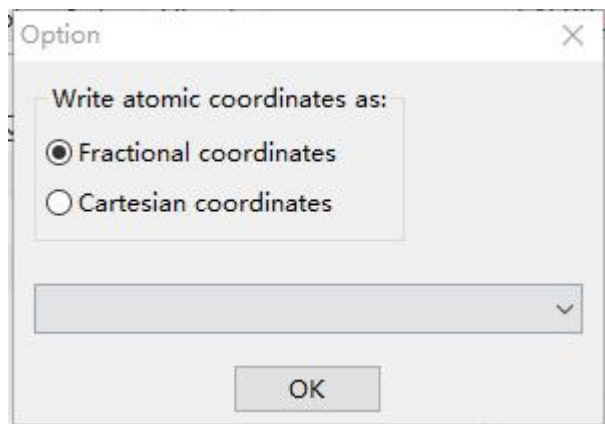
[CIF](#)

- Computed
- Conventional Standard
- Primitive Cell
- Symmetrized

1.1.2 晶体结构建模：编辑结构

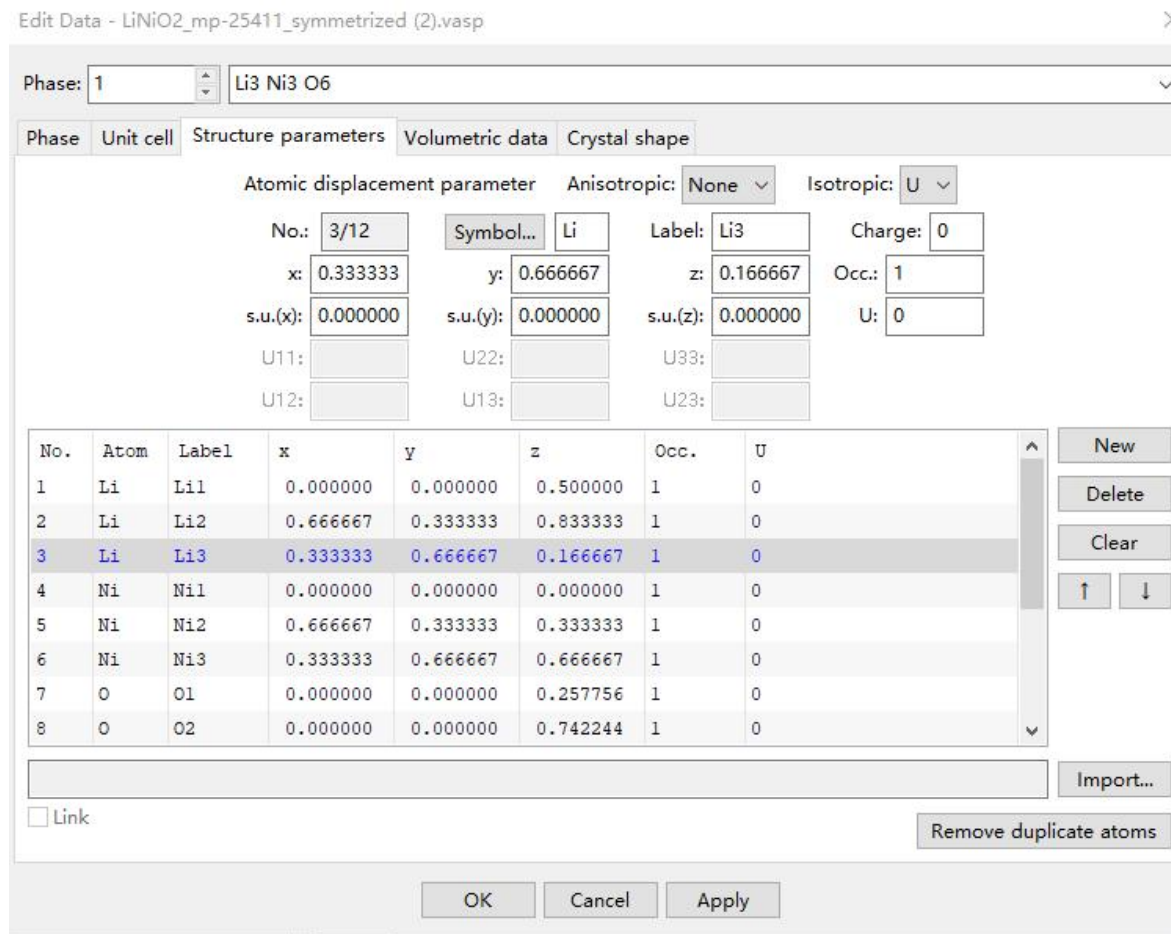


1. 转化成.vasp格式：
File-Export Data-.vasp格式



2. 打开.vasp格式，编辑结构

Edit-Edit Data-Structure Parameters



Phase: 1 | Li3 Ni3 O6

Phase | Unit cell | Structure parameters | Volumetric data | Crystal shape

Atomic displacement parameter | Anisotropic: None | Isotropic: U

No.: 3/12 | Symbol... Li | Label: Li3 | Charge: 0

x: 0.333333 | y: 0.666667 | z: 0.166667 | Occ.: 1

s.u.(x): 0.000000 | s.u.(y): 0.000000 | s.u.(z): 0.000000 | U: 0

U11: | U22: | U33: |

U12: | U13: | U23: |

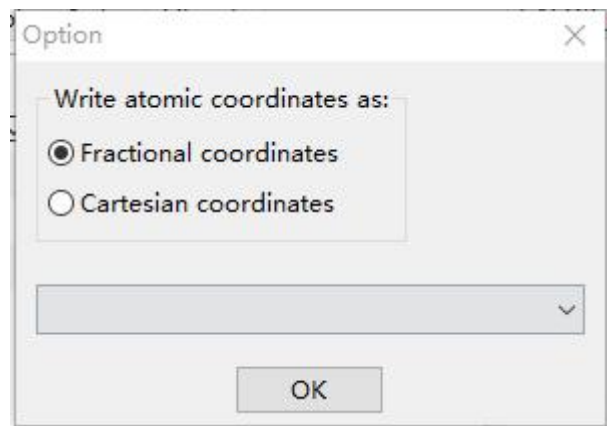
No.	Atom	Label	x	y	z	Occ.	U
1	Li	Li1	0.000000	0.000000	0.500000	1	0
2	Li	Li2	0.666667	0.333333	0.833333	1	0
3	Li	Li3	0.333333	0.666667	0.166667	1	0
4	Ni	Ni1	0.000000	0.000000	0.000000	1	0
5	Ni	Ni2	0.666667	0.333333	0.333333	1	0
6	Ni	Ni3	0.333333	0.666667	0.666667	1	0
7	O	O1	0.000000	0.000000	0.257756	1	0
8	O	O2	0.000000	0.000000	0.742244	1	0

Buttons: New, Delete, Clear, Import..., Link, Remove duplicate atoms, OK, Cancel, Apply

1.1.2 晶体结构建模：编辑结构

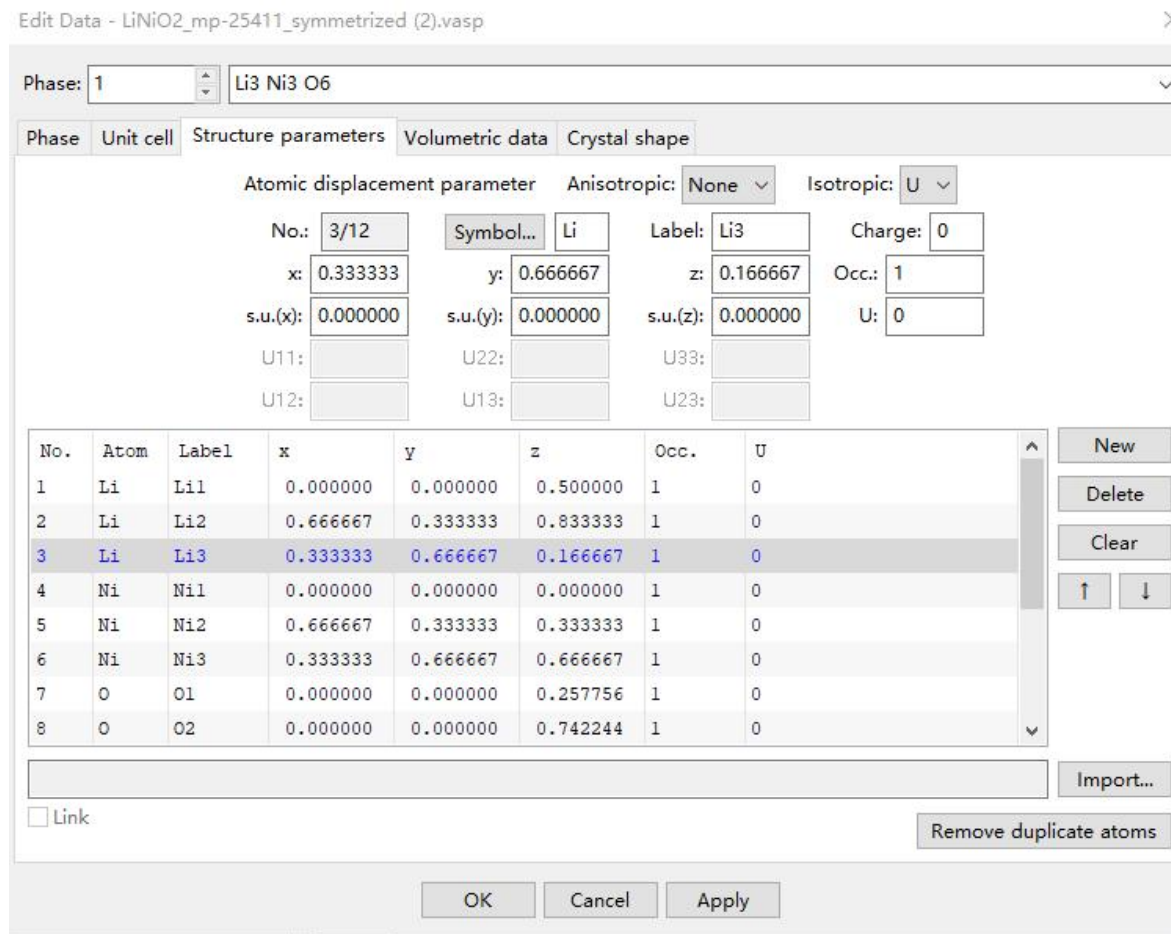
3. 编辑完成后导出：

File-Export Data-.vasp格式



2. 打开.vasp格式，编辑结构

Edit-Edit Data-Structure Parameters



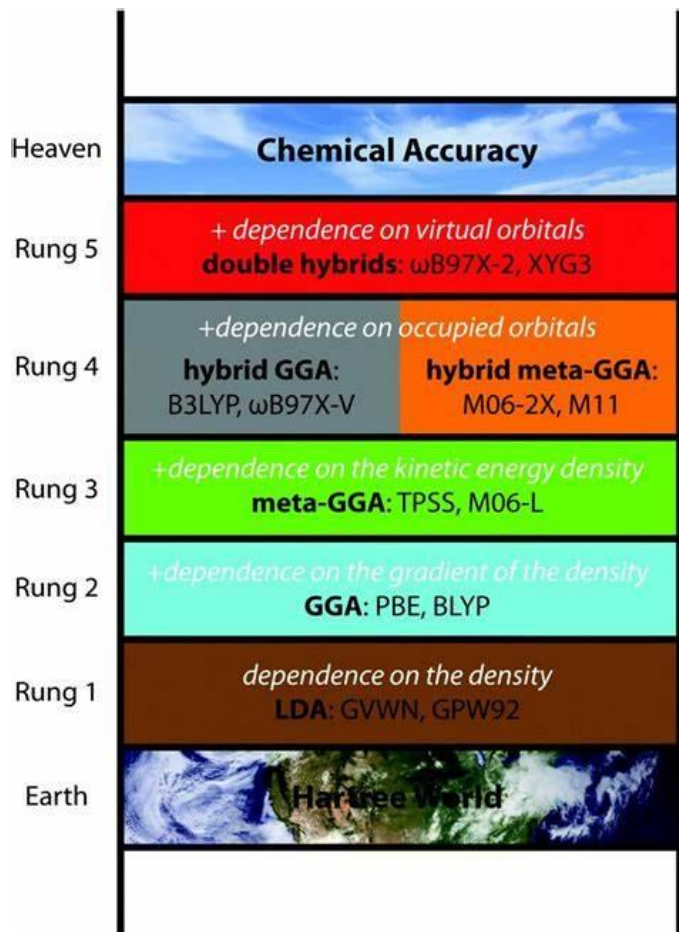
交换关联泛函

选择合适的赝势和交换关联泛函

赝势

Kohn-Sham
Equation

$$E[\rho] = T_s[\rho] + \int d\mathbf{r} v_{\text{ext}}(\mathbf{r})\rho(\mathbf{r}) + E_H[\rho] + E_{\text{xc}}[\rho],$$



为了减少计算量，对内层（非价电子）电子和核对价电子的作用打包起来简化成一个简单的势

赝势通常是通过拟合方法构造的，以确保它在原子核外的行为与实际电子相互作用在某种程度上是一致的

膜守恒赝势

超软赝势

投影缀加波赝势

1.2.1 输入文件

管理设置：
计算模式，输入输出
目录等

体系设置：
原子数量，截断能，
自旋，占据计算等

电子步设置：
电子步收敛标准，新
的密度混合比例等

离子步设置：
离子步移动步长等

元素以及赝势设置

布里渊区采样精度

晶格参数

原子坐标

```
&control
  calculation = 'relax'
  restart_mode='from_scratch',
  prefix='silicon',
  tstress = .true.
  tprnfor = .true.
  pseudo_dir = './'
  outdir='./'
/
&system
  ibrav= 0,
  celldm(1)=1.8897259886d0,
  nat= 2,
  ntyp= 1,
  ecutwfc =50.0,
  nspin = 1,
  occupations = 'fixed',
/
&electrons
  mixing_mode = 'plain'
  mixing_beta = 0.7
  conv_thr = 1.0d-8
/
&ions
/
ATOMIC_SPECIES
  Si 28.08 Si.SG15.PBE.UPF
K_POINTS {automatic}
  8 8 8 0 0 0
CELL_PARAMETERS
  0.00000000 2.71535000 2.71535000
  2.71535000 0.00000000 2.71535000
  2.71535000 2.71535000 0.00000000
ATOMIC_POSITIONS (crystal)
  Si 0.00000000 0.00000000 0.00000000
  Si 0.25000000 0.25000000 0.25000000
```

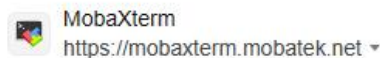
Bing搜索: pw.x



二、实操

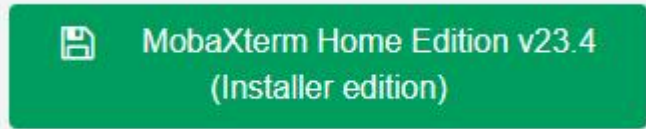
请大家下载 MobaXterm (用bing搜索)

①安装:



MobaXterm free Xserver and tabbed SSH client for Windows

网页 2023年1月3日 · MobaXterm is a portable and powerful tool for remote computing that combines X server, SSH client, network tools and Unix commands in a single application. ...



<https://mobaxterm.mobatek.net/download-home-edition.html>

②登录设置:



第一步: 点击Session图标, 弹出Settings界面

第二步: 点击SSH图标按照下图, 修改host、username和Port

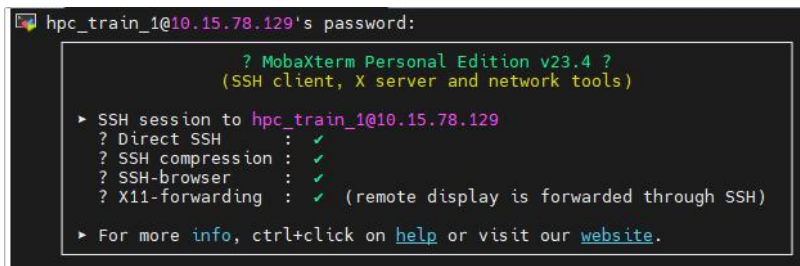


IP(remote host):
10.15.78.129

本次培训使用的账号仅为教学使用, 后续大家需要申请高性能平台计算账号:

[高性能账号申请 \(shanghaitech.edu.cn\)](http://shanghaitech.edu.cn)

③输入密码 (密码是盲打!):



[高性能集群使用手册 \(shanghaitech.edu.cn\)](http://shanghaitech.edu.cn)

0.列出当前的目录下的内容

ls

1.复制例子目录为自己的目录

cp -r 20231130 xiety(你自己想要创建的目录)

2.打开自己的目录

cd xiety(你自己想要创建的目录)

3.列出当前的目录(自己的目录) 下的内容

ls

```
18. 10.15.78.129 (hpc_train) x
? MobaXterm Personal Edition v23.3 ?
(SSh client, X server and network tools)
? SSH session to hpc_train@10.15.78.129
? Direct SSH : ✓
? SSH compression : ✓
? SSH-browser : ✓
? X11-forwarding : ✓ (remote display is forwarded through SSH)
? For more info, ctrl+click on help or visit our website.

Last login: Thu Nov 30 14:16:29 2023 from 10.19.52.160
[hpc_train@hpc-login-telperion ~]$ ls
20231026          GSM5688xxx          perl5
20231116          gzh                  PWmat
20231123          titel                 PYGAMD
20231130          java.log.14362       R
admin             jslzyszh             samtools
a.pbs             main.c                sist
BBa.st            matlab_crash_dump.13706-1 slst
bianyi.pbs        matlab_crash_dump.13917-1 slurm
bin               matlab_crash_dump.13917-2 spst
default_pipeline.star matlab_crash_dump.29582-1 spst_2021
demo.zip          matlab_crash_dump.29582-2 Test
DNAMAN            matlab_crash_dump.4636-1 turtle
Downloads         matlab_crash_dump.9968-1 tutorialData-1K.tar.gz
echo              out                    usability_test_mononode_zx_xxy.timelog
faston            out.fasta             vasp
gcc_bianyi        out.list              vasp.5.4.4.tar.gz
gmxtest           pbs_gen.sh            zhixing.pbs
[hpc_train@hpc-login-telperion ~]$
```

计算例子

```
[hpc_train@hpc-login-telperion 20231130]$ ls  
demo-bands demo-relax demo-Si-bands demo-Si-relax psp results-Si-bands results-Si-relax scripts
```

课后例子

实操例子

赝势文件包

实操例子结果

Python程序包

Linux 基本操作:

1.打开目录: `cd {目录名}`

2.返回上一级目录: `cd ..`

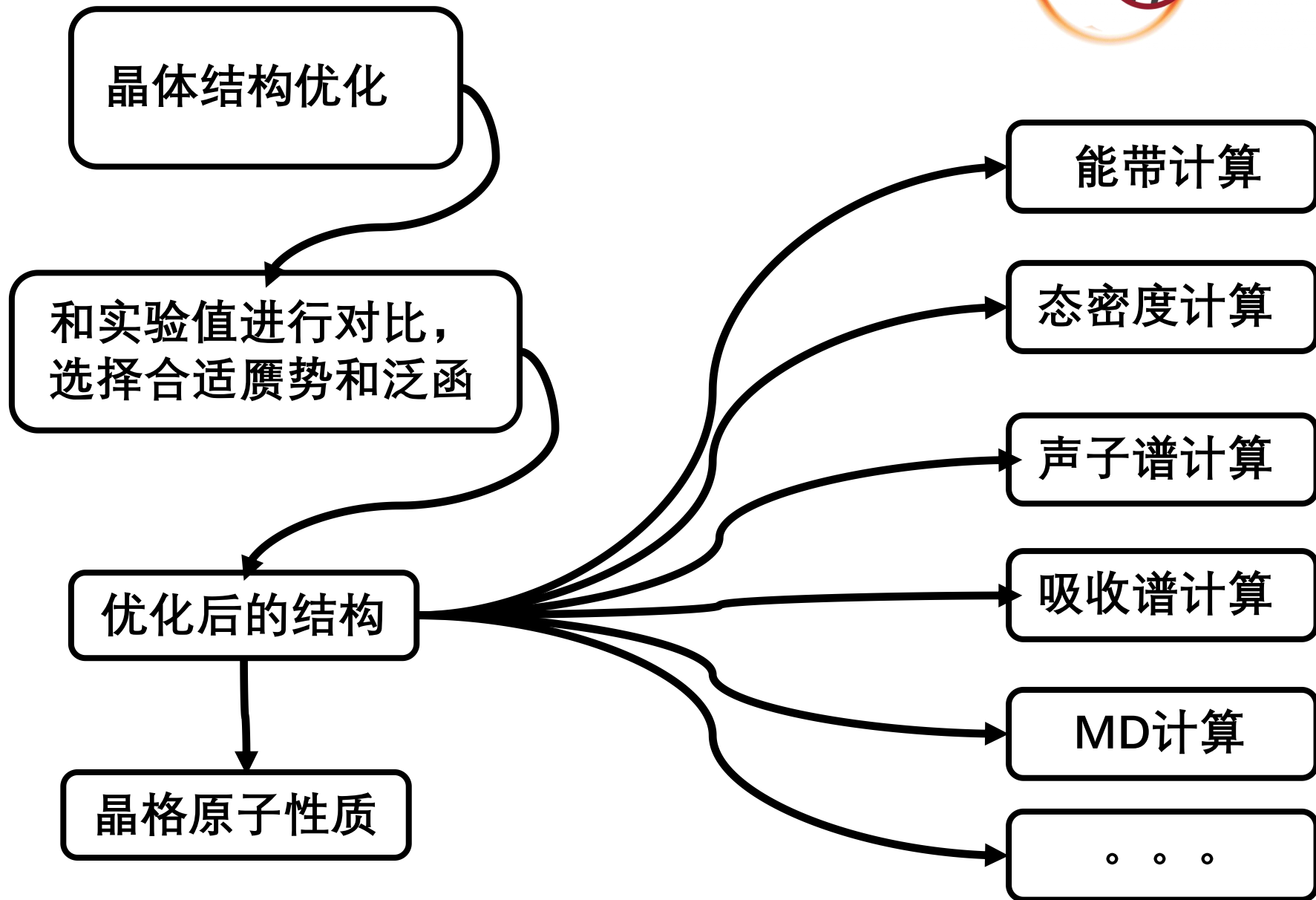
3.打开文件: `vi {文件名}`

4.在文件中编辑: 按下i, $\uparrow\downarrow\leftarrow\rightarrow$ 移动光标, 在光标位置进行输入/修改

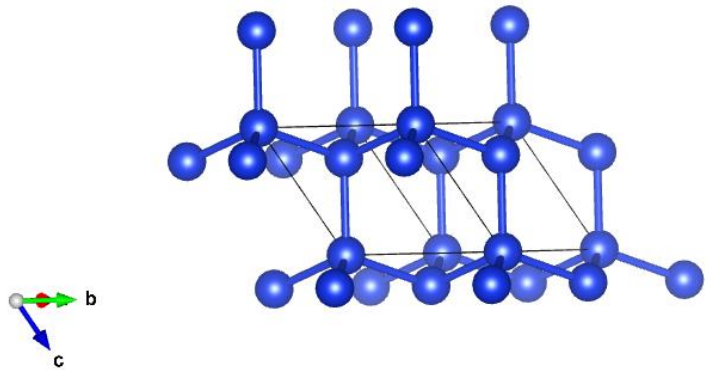
5.保存对文件的编辑: 按下ESC, 然后同时按下shift和 ":" 两个键, 输入wq并回车

6.编辑文件但是不保存: 同5, 但是输入q! 并回车

计算流程



2.1 晶体结构优化——relaxation计算 以Si为例



准备文件

1. relax输入文件

2. 赝势

3. 提交任务的bash脚本

```
demo-Si-relax
```

1. cd demo-Si-relax (进入该目录)
2. ls (列出该目录下所有文件)

relax输入文件

提交任务的bash脚本

赝势

```
in_relax job_script Si.SG15.PBE.UPF
```

PBE交换关联泛函

膜守恒赝势

2.1.1 relaxation 输入文件



in_relax

1. vi in_relax (打开文件)
2. Shift键和“:”同时按，按完以后wq加回车 (保存并且退出)

```
&control
  calculation = 'relax'
  restart_mode='from_scratch',
  prefix='silicon',
  tstress = .true.
  tprnfor = .true.
  pseudo_dir = './'
  outdir='./'
/
&system
 ibrav= 0,
  cellldm(1) =1.8897259886d0,
  nat= 2,
  ntyp= 1,
  ecutwfc =50.0,
  nspin = 1,
  occupations = 'fixed',
/
&electrons
  mixing_mode = 'plain'
  mixing_beta = 0.7
  conv_thr = 1.0d-8
/
```

模式为relax

原子数量
元素数量

Scf收敛标准

```
ATOMIC_SPECIES
  Si 28.08 Si.SG15.PBE.UPF
K_POINTS {automatic}
  8 8 8 0 0 0
CELL_PARAMETERS
  0.00000000 2.71535000 2.71535000
  2.71535000 0.00000000 2.71535000
  2.71535000 2.71535000 0.00000000
ATOMIC_POSITIONS (crystal)
  Si 0.00000000 0.00000000 0.00000000
  Si 0.25000000 0.25000000 0.25000000
```

元素；相对原子质量；赝势名称

在布里渊区k点采样个数

晶胞参数

原子位置

2.1.2 提交任务bash脚本



`job_script`

1. vi job_script (打开文件)
2. Shift键和“:”同时按，按完以后wq加回车 (保存并且退出)

```
#!/bin/bash
#PBS -N Si                      计算任务名称
#PBS -l nodes=1:ppn=8                      计算调用节点数和核数
#PBS -l walltime="00:30:00"                      计算预计时间
#PBS -S /bin/bash
#PBS -j oe
#PBS -q pub_jx                      计算节点分区
[]
cd $PBS_O_WORKDIR

NPROC=`wc -l < $PBS_NODEFILE`

echo This job has allocated $NPROC proc > log

module load compiler/intel/2021.3.0
module load mpi/intelmpi/2021.3.0                      计算需要加载的模块
module load apps/quantum-espresso/intelmpi/6.7

mpirun --bind-to core -np $NPROC -hostfile $PBS_NODEFILE pw.x -npool 2 -ndiag 4 < in_relax >& out_relax
```

计算relax的执行程序

输入文件

输出文件

2.1.3 提交任务



1. qsub job_script (提交bash脚本)
2. qstat (查看脚本运行情况)

Job id	Name	Username	Time Use	S	Queue
2359340	LiNC5200LiNiSIG	scfa1465	00:10:46	R	v5_192
2361534	LiNC5101LiNiSIG	scfa1465	00:01:55	R	v5_192
2361538	LiNC5101LiNiSIG	scfa1465	00:01:54	R	v5_192
2361541	LiNC5101LiNiSIG	scfa1465	00:01:54	C	v5_192
2361680	LiNC5101LiNi	scfa1465	00:01:22	R	v5_192
2361953	LiNC5101LiNi	scfa1465	00:00:18	R	v5_192

7306314.node1	LiNC3101	zhengfan_2	05:38:35	R	spst_cal
7306315.node1	LiNC3303	zhengfan_2	05:08:17	R	spst_cal
7306328.node1	LiNC3105	zhengfan_2	03:01:46	R	spst_cal
7306358.node1	LiNC3207	zhengfan_2	0	Q	spst_pub
7306359.node1	LiNC3304	zhengfan_2	0	Q	spst_pub

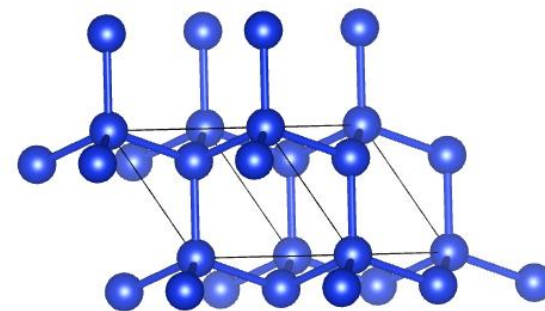
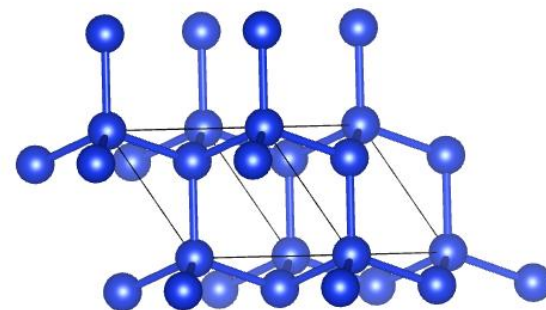
2.1.4 查看优化的结构



1. `python ../scripts/qein2car.py in_relax xsf`
(从**输入**文件提取**优化前**的结构)

2. `python ../scripts/read_qeout_relax.py out_relax`
(从**输出**文件提取**优化后**的结构)

3. `VESTA in_relax.xsf out_relax.xsf`
(用VESTA可视化结构)



2.2 电子结构计算——能带计算

以Si为例



注意：bands计算需要在relax计算后得到的波函数基础上进行计算，因此在实际计算中需要和relax计算在同一个文件夹内进行

demo-Si-bands

准备文件

1.bands输入文件

2.赝势

3.提交任务的bash脚本

1. cd demo-Si-bands (进入该目录)
2. ls (列出该目录下所有文件)

```
in_bands job_script silicon.save silicon.xml Si.SG15.PBE.UPF Si.vasp
```

Relax计算得到的波函数文件

2.2.1 bands 输入文件



in_bands

1. vi in_bands (打开文件)
2. Shift键和“:”同时按，按完以后wq加回车 (保存并且退出)

```
control
  calculation = 'bands'
  restart_mode='from_scratch',
  prefix='silicon',
  tstress = .true.
  tprnfor = .true.
  pseudo_dir = './'
  outdir='./'
  verbosity='high'
/
&system
  ibrav= 0,
  cellldm(1) =1.8897259886d0,
  nat= 2,
  ntyp= 1,
  nbnd= 10,
  ecutwfc =50.0,
  nspin = 1,
  occupations = 'fixed',
/
&electrons
  mixing_mode = 'plain'
  mixing_beta = 0.7
  conv_thr = 1.0d-8
/
&ions
/
ATOMIC_SPECIES
Si 28.08 Si.SG15.PBE.UPF
```

模式为bands

nbnd数量: 允带数量*1.2

2.2.1 bands 输入文件



在布里渊区k点路径

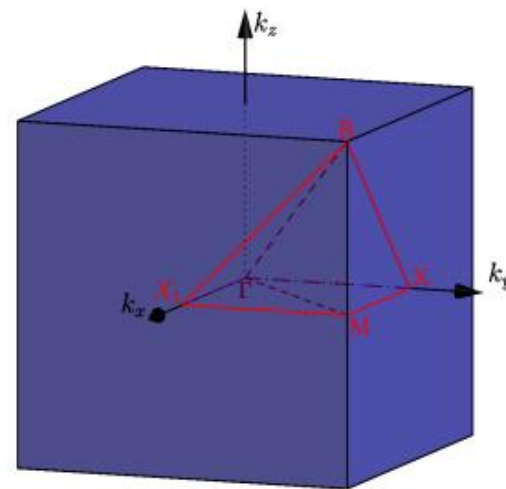
```
K_POINTS {crystal_b}
17
0.0 0.0 0.0 1.0 !  $\Gamma$ 
0.125 0.0 0.0 1.0
0.25 0.0 0.0 1.0
0.375 0.0 0.0 1.0
0.5 0.0 0.0 1.0 ! X
0.5 0.125 0.0 1.0
0.5 0.25 0.0 1.0
0.5 0.375 0.0 1.0
0.5 0.5 0.0 1.0 ! M
0.375 0.375 0.0 1.0
0.25 0.25 0.0 1.0
0.125 0.125 0.0 1.0
0.0 0.0 0.0 1.0 !  $\Gamma$ 
0.125 0.125 0.125 1.0
0.25 0.25 0.25 1.0
0.375 0.375 0.375 1.0
0.5 0.5 0.5 1.0 ! R

CELL_PARAMETERS
0.00000000 2.71535000 2.71535000
2.71535000 0.00000000 2.71535000
2.71535000 2.71535000 0.00000000

ATOMIC_POSITIONS (crystal)
Si 0.00000000 0.00000000 0.00000000
Si 0.25000000 0.25000000 0.25000000
```

QEtoolkit

<https://www.densityflow.com/>



[brillouin_zones.pdf \(mit.edu\)](http://brillouin_zones.pdf.mit.edu)

2.1.2 提交任务bash脚本



`job_script`

1. vi job_script (打开文件)
2. Shift键和“:”同时按，按完以后wq加回车 (保存并且退出)

```
#!/bin/bash
#PBS -N Si _____ 计算任务名称
#PBS -l nodes=1:ppn=8 _____ 计算调用节点数和核数
#PBS -l walltime="00:30:00" _____ 计算预计时间
#PBS -S /bin/bash
#PBS -j oe
#PBS -q pub_jx _____ 计算节点分区
[]
cd $PBS_O_WORKDIR

NPROC=`wc -l < $PBS_NODEFILE`

echo This job has allocated $NPROC proc > log

module load compiler/intel/2021.3.0
module load mpi/intelmpi/2021.3.0
module load apps/quantum-espresso/intelmpi/6.7

mpirun --bind-to core -np $NPROC -hostfile $PBS_NODEFILE pw.x -npool 2 -ndiag 4 < in_bands >& out_bands
```

计算bands的执行程序

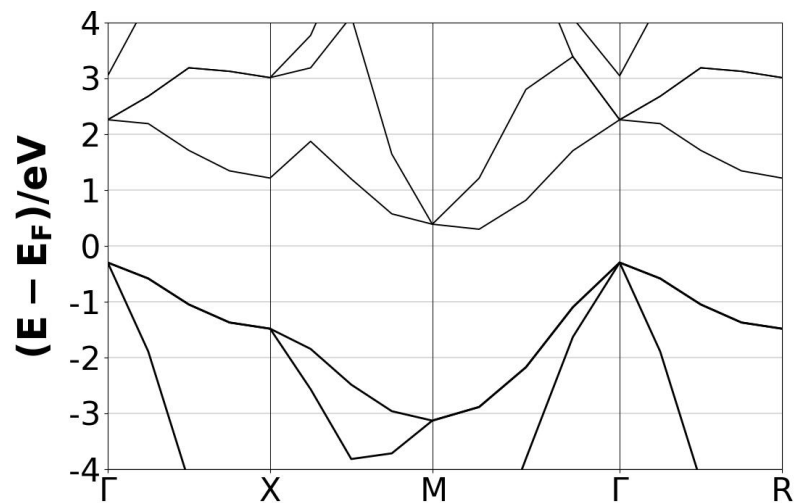
输入文件

输出文件

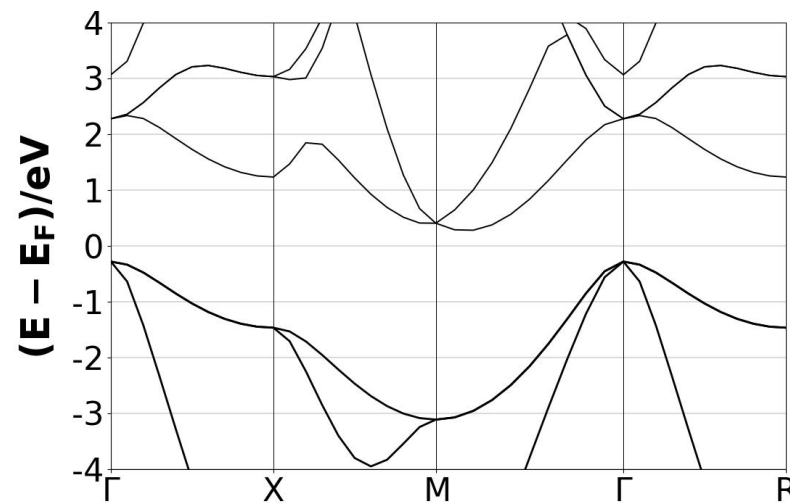
2.2.3 查看结果



python ../scripts/titledbandstrv5.1.py out_bands 4 4 4 (画出能带)



```
K_POINTS {crystal_b}
17
0.0 0.0 0.0 1.0 ! Gamma
0.125 0.0 0.0 1.0
0.25 0.0 0.0 1.0
0.375 0.0 0.0 1.0
0.5 0.0 0.0 1.0 ! X
0.5 0.125 0.0 1.0
0.5 0.25 0.0 1.0
0.5 0.375 0.0 1.0
0.5 0.5 0.0 1.0 ! M
0.375 0.375 0.0 1.0
0.25 0.25 0.0 1.0
0.125 0.125 0.0 1.0
0.0 0.0 0.0 1.0 ! Gamma
0.125 0.125 0.125 1.0
0.25 0.25 0.25 1.0
0.375 0.375 0.375 1.0
0.5 0.5 0.5 1.0 ! R
```



```
K_POINTS {crystal_b}
41
0.0 0.0 0.0 1.0 ! Gamma
0.05 0.0 0.0 1.0
0.1 0.0 0.0 1.0
0.15 0.0 0.0 1.0
0.2 0.0 0.0 1.0
0.25 0.0 0.0 1.0
0.3 0.0 0.0 1.0
0.35 0.0 0.0 1.0
0.4 0.0 0.0 1.0
0.45 0.0 0.0 1.0
0.5 0.0 0.0 1.0 ! X
0.5 0.05 0.0 1.0
0.5 0.1 0.0 1.0
0.5 0.15 0.0 1.0
0.5 0.2 0.0 1.0
0.5 0.25 0.0 1.0
0.5 0.3 0.0 1.0
0.5 0.35 0.0 1.0
0.5 0.4 0.0 1.0
0.5 0.45 0.0 1.0
0.5 0.5 0.0 1.0 ! M
0.45 0.45 0.0 1.0
0.4 0.4 0.0 1.0
0.35 0.35 0.0 1.0
0.3 0.3 0.0 1.0
0.25 0.25 0.0 1.0
0.2 0.2 0.0 1.0
0.15 0.15 0.0 1.0
0.1 0.1 0.0 1.0
0.05 0.05 0.0 1.0
0.0 0.0 0.0 1.0 ! Gamma
0.05 0.05 0.05 1.0
0.1 0.1 0.1 1.0
0.15 0.15 0.15 1.0
0.2 0.2 0.2 1.0
0.25 0.25 0.25 1.0
0.3 0.3 0.3 1.0
0.35 0.35 0.35 1.0
0.4 0.4 0.4 1.0
0.45 0.45 0.45 1.0
0.5 0.5 0.5 1.0 ! R
```

K_point number: 17 VS 41

K点采样数量越多，能带计算的精度越高

3.1 晶体结构优化——计算一个大体系



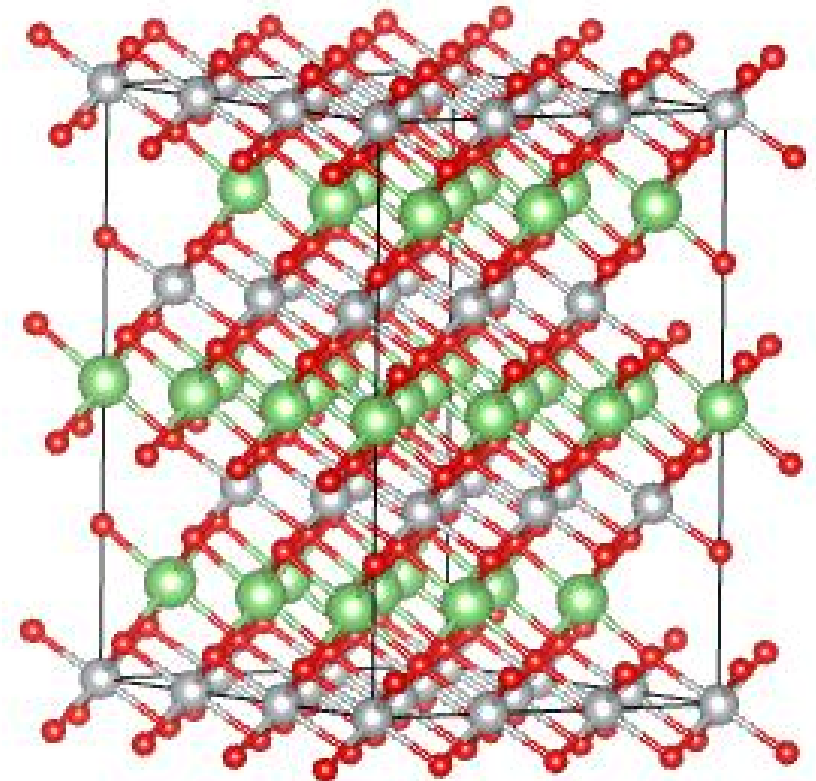
准备
文件

1. relax输入文件（自己构建）

2. 赝势（psp中复制）

3. 提交任务的bash脚本

```
demo-relax
```



2.1.1 relaxation 输入文件



```
in_relax
```

```
&control
  calculation = 'relax'
  restart_mode='from_scratch',
  prefix='silicon',
  tstress = .true.
  tprnfor = .true.
  pseudo_dir = './'
  outdir='./'
/
&system
 ibrav= 0,
  celldm(1) =1.8897259886d0,
  nat= 2,
  ntyp= 1,
  ecutwfc =50.0,
  nspin = 1,
  occupations = 'fixed',
/
&electrons
  mixing_mode = 'plain'
  mixing_beta = 0.7
  conv_thr = 1.0d-8
/
```

模式为relax

原子数量 (需要修改)
元素数量 (需要修改)

```
ATOMIC_SPECIES
  Si 28.08 Si.SG15.PBE.UPF
K_POINTS {automatic}
  8 8 8 0 0 0
CELL_PARAMETERS
  0.00000000 2.71535000 2.71535000
  2.71535000 0.00000000 2.71535000
  2.71535000 2.71535000 0.00000000
ATOMIC_POSITIONS (crystal)
  Si 0.00000000 0.00000000 0.00000000
  Si 0.25000000 0.25000000 0.25000000
```

元素；相对原子质量；赝势名称

在布里渊区k点采样个数：改为3 3 2

晶胞参数

原子位置

2.1.2 提交任务bash脚本



`job_script`

1. vi job_script (打开文件)
2. Shift键和“:”同时按，按完以后wq加回车 (保存并且退出)

```
#!/bin/bash
#PBS -N Si
#PBS -l nodes=2:ppn=32
#PBS -l walltime="05:30:00"
#PBS -S /bin/bash
#PBS -j oe
#PBS -q pub_jx

cd $PBS_O_WORKDIR

NPROC=`wc -l < $PBS_NODEFILE`

echo This job has allocated $NPROC proc > log
module load compiler/gcc/7.3.1
module load compiler/intel/2021.3.0
module load mpi/intelmpi/2021.3.0
module switch compiler/intel/2021.3.0
module switch mpi/intelmpi/2021.3.0
module load apps/quantum-espresso/intelmpi/6.7

mpirun --bind-to core -np $NPROC -hostfile $PBS_NODEFILE pw.x -npool 4 -ndiag 16 < in_relax >& out_relax
```

计算调用节点数和核数 (跨节点计算)

计算预计时间 (更长)

计算节点分区

计算需要加载的模块

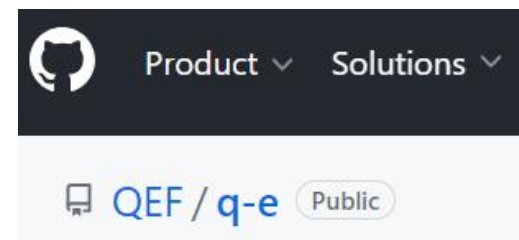
计算relax的执行程序

输入文件

输出文件



<https://www.quantum-espresso.org/>



<https://github.com/QEF/q-e>



<http://bbs.keinsci.com/>



<https://www.youtube.com/@realquantumnerd/videos>



<https://xh125.github.io/archive/?tag=QUANTUM+ESPRESSO>

关于提高计算精度的方法：（需要和计算成本进行取舍）


1. 选择高精度的赝势和交换关联泛函

```
K_POINTS {automatic}  
8 8 8 0 0 0
```

2. 选择布里渊区较密集的K点采样

3. 选择较大的截断能

```
&system  
ibrav= 0,  
celldm(1) =1.8897259886d0,  
nat= 2,  
ntyp= 1,  
ecutwfc =50.0,  
nspin = 1,  
occupations = 'fixed',  
/
```



关于VESTA配置:

1. 申请一个超算账号
2. 右侧命令行输入`mkdir ~/bin`
3. 右侧命令行输入`cd ~/bin`
4. 左侧导航栏打开bin, 上传VESTA-gtk3.tar
5. 右侧命令行输入`tar -xvf VESTA-gtk3.tar`
6. 右侧命令行输入`ln -s VESTA-gtk3/VESTA .`
7. (查看自己所在绝对路径): 右侧命令行输入`pwd`
(返回`/hpc/data/home/hpc_train/bin`) 返回值依人而定
8. (配置环境变量) `vi ~/.bashrc`
9. 在最后输入: `export PATH= /hpc/data/home/hpc_train/bin:$PATH`
10. Esc然后shift+“:”然后wq保存退出
11. 右侧命令行输入`bash`

关于VESTA使用:

VESTA {你需要查看的文件名}