

状态方程与态密度计算

王越超 许熙 张旻烨

状态方程计算

态密度计算

参考材料

XmGrace

- ▶ 官方Tutorial, UG, FAQ
- ▶ XMGR - Grace Graphics Refresher Training & Reference Guide
- ▶ Grace typesetting for titles, legends, tick marks, greek letters

态密度

- ▶ GPAW PDOS

状态方程计算

为什么用状态方程优化晶胞体积

状态方程 (Equation of state):

$$E \equiv E(V)$$
$$V_0 := \arg \min E$$

- ▶ 避免Pulay Stress 对结果的影响
- ▶ 对于大体系节省计算开销, 不用担心结构收敛的问题

简单算例: Si I

参考的脚本: `cd Si loop.sh`

```
#!/bin/bash
BIN=/path/to/your/vasp/executable
rm WAVECAR SUMMARY.dia
for i in 5.1 5.2 5.3 5.4 5.5 5.6 5.7 ; do
cat >POSCAR <<!
cubic diamond
    $i
    0.0      0.5      0.5
    0.5      0.0      0.5
    0.5      0.5      0.0
    2
Direct
    -0.125 -0.125 -0.125
    0.125  0.125  0.125
!
echo "a= $i" ; mpirun -n 2 $BIN
E=`awk '/F=/ {print $5}' OSZICAR` ; echo $i $E >>SUMMARY.dia
done
```

简单算例: Si II

获取 Si POTCAR (替换 `cwd` 为你的工作文件夹)

```
cp /opt/software/vasp/vasppot-5.4/potpaw_PBE/Si/POTCAR cwd
```

思考

- ▶ 如何计算或从 VASP 输出提取晶格体积?
- ▶ 在 ENCUT=300 下, 和 ISIF=3 的计算结果比较, 有何差别?
- ▶ 和 ENCUT=550 的 ISIF=3 的结果比较

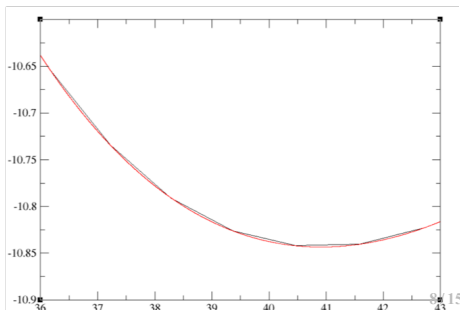
状态方程拟合

Birch-Murnaghan equation of state ([wikipedia](#))

$$E(V) = E_0 + \frac{9V_0B_0}{16} \times \left\{ \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^3 B'_0 + \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^2 \left[6 - 4 \left(\frac{V_0}{V} \right)^{\frac{2}{3}} \right] \right\}$$

Grace 拟合参数模板: BMEOS.fit

1	#	Volume A^3	Energy eV
2	36.17578125	- .10654493E+02	
3	37.21925000	- .10734168E+02	
4	38.28259375	- .10790780E+02	
5	39.36600000	- .10826148E+02	
6	40.46965625	- .10842124E+02	
7	41.59375000	- .10840383E+02	
8	42.73846875	- .10822378E+02	



态密度计算

定义

给定能量范围内的态的数目

$$D(E) = \sum_n \delta(E - E_n)$$

VASP INCAR

- ▶ NEDOS
- ▶ EMIN
- ▶ EMAX
- ▶ LORBIT
- ▶ ISMEAR
- ▶ SIGMA

计算流程

1. 结构优化/静态计算
2. 态密度计算
 - ▶ 固定电荷密度
 - ▶ 关闭对称性
 - ▶ 增大 NEDOS
 - ▶ 增大 k 点密度

主要输出文件

- ▶ DOSCAR
- ▶ PROCAR
- ▶ vasprun.xml

使用 py_dos.py

在工作站上首先载入 tmckit

```
module load tmckit
```

查看帮助

```
py_dos.py --help
```

示例: wurtzite GaN

```
py_dos.py -d dos -i vasp --format=%s-%l -g
```

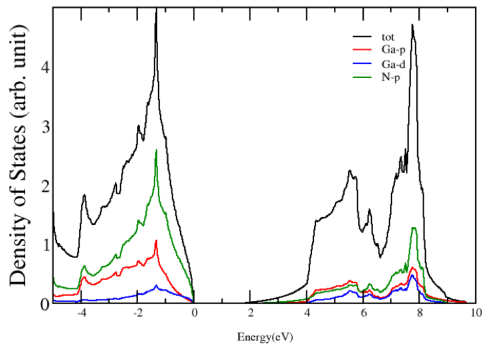


Figure 1: w-GaN