

POSCARtoolkit.py

在学习下面 **neb** 的计算前，思考了很久，感觉有必要先给大家介绍一个非常实用的 **python** 小脚本，经历过反复的修改，最终在 **TAMAS** 的努力下，完成了一个稳定的版本，
本节我们介绍一下这个脚本的具体使用方法。

1 这个脚本有 2 个版本，分别适用于 python3 和 python2 的 2.6 及以上版本，大家根据自己使用的 python 版本进行下载，如果运行的时候出现错误，请先检查是不是版本的问题。

2 使用前的准备工作：

2.1) 下载脚本，将脚本命名为：POSCARtoolkit.py

2.2) 可执行化：chmod u+x POSCARtoolkit.py

2.3) 将脚本移到~/bin 文件夹下面： mv POSCARtoolkit.py ~/bin

注释：

2.2 和 2.3 步的顺序可以颠倒，不影响使用

如果没有~/bin 文件夹，那么 mkdir ~/bin 手动创建一个即可。

3 实现分数坐标（Direct）向笛卡尔坐标（Cartesian）的转换：

VASP 的输出结果是以分数坐标的形式存在 CONTCAR 中，但我们在操作模型的过程中，移动原子都是以 Å 来进行的，所以讲分数坐标转化为笛卡尔坐标对于搭建模型很有帮助。

脚本用法： POSCARtoolkit.py -i POSCAR

注意：如果你的 POSCARtoolkit.py 脚本在和 POSCAR 一个目录下，使用下图中的命令，如果你已经完成了前面 2 中的步骤，脚本前面的 python 不用输入。

```
C:\Users\ppliv\Desktop>python POSCARtoolkit.py -i POSCAR

#####
# For VASP 5.2 or higher versions #
#           Author:Li,Q;Xu,N       #
#           Verision 1.2           #
#####

File to be handled is *****POSCAR*****

This POSCAR has Direct Coordinations, Conversion is starting...

-----
POSCAR with Cartesian Coordinations is named as POSCAR_C
-----
```

描述 1) -i 代表 input 的意思, 后面紧跟你要转化的文件, 可以是 POSCAR, CONTCAR, 也可以是其他的 VASP 的坐标文件, 比如你把 POSCAR 命名成 bigbro, 也是可以直接转换的。POSCARtoolkit.py -i bigbro 即可。

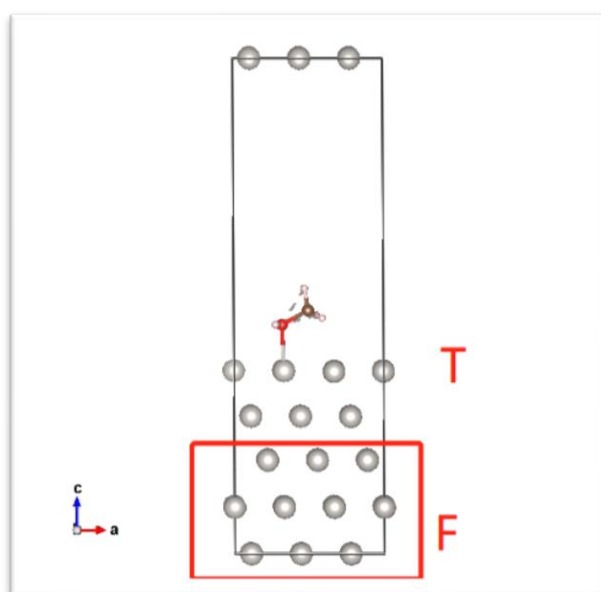
描述 2): 脚本将含有分数坐标的 POSCAR 转换成笛卡尔坐标的 POSCAR_C。输出的文件被命名为 POSCAR_C。

描述 3) 如果你预先把 POSCAR 命名为 bigbro, 那么运行程序后, 输出结果为 bigbro_C。

描述 4) 如果 POSCAR 已经为笛卡尔坐标, 则转换终止, 大家可以尝试一下转换一个笛卡尔坐标的 POSCAR。

4 原子层数的固定

除了可以实现坐标系的转换, 该脚本还可以根据 Z 方向的原子坐标, 固定底部几层原子。这在 slab 模型的相关计算中非常实用, 如下图所示, 在 slab 模型中, 我们经常将底部的原子层固定, 只放开表面的来进行计算。



比如我们把底部的三层原子进行固定。脚本的用法如下：

POSCARtoolkit.py -i POSCAR -f

```
C:\Users\ppliv\Desktop>python POSCARtoolkit.py -i POSCAR_C -f

#####
# For VASP 5.2 or higher versions #
#           Author:Li, Q;Xu, N           #
#           Verision 1.2                   #
#####

File to be handled is *****POSCAR_C*****

-----

Cartesian Coordinates found, only for fixing atoms!
Then type how many layers to be fixed, from bottom to top.
-----

Found 7 layers, choose how many layers to be fixed----->3

-----

POSCAR with Cartesian Coordiations is named as POSCAR_C_C

Atom fix completed, if layers are not separated properly,
Please alter threstholds with flags -y or type -h for help!
-----
```

描述 1) 脚本会根据阈值（默认 1.5 Å）划分层，这里的 1.5 Å 指的是层间距。

描述 2) 用户可以通过在脚本后面增加参数 -y 1.0 自己定义更小的阈值

描述 3) 如果不想每次使用的时候使用参数-y，用户也可以直接在脚本里面修改阈值的大小，如下如，在第 40 行中，默认的阈值 1.5 被修改 0.5。

```
35 parser.add_option("-s", "--select",
36                   action="store_true", dest="selected", default=False,
37                   help="select atoms to fix or relax atoms. with -r or -f")
38
39 parser.add_option("-y", "--threshold",
40                   type = 'float',dest="threshold", default=0.5,
41                   help="thresholds of distance to separate layers!")
42
43 (options, args) = parser.parse_args()
44 #print options.selected
45 print ""
```

当然，修改脚本默认的参数后，如果使用 -y，还是会按照 -y 后面的阈值进行操作。

描述 4) 输入完命令，回车后，会提醒用户输入固定的层数，这里我们输入 3，回车，即固定底下三层原子，而其他原子放开。

描述 5) 上图命令中，我们对 POSCAR_C 进行操作，输出文件为 POSCAR_C_C。

5 固定和放开用户选择的原子。

除了固定层数外，该脚本允许用户选择部分原子放开，固定或者部分放开。这个功能实现的前提是 POSCAR 或者 CONTCAR 中有 “Selective dynamics” 信息。如果没有的话。则可通过功能 4 先固定任意的原子层 (POSCARtoolkit.py -i CONTCAR -f)，这样的话 “Selective dynamics” 就会被写入到输出文件 POSCAR_C 中。然后用户再对 POSCAR_C 进行原子选择性操作。下面我们先详细介绍一下脚本的用法，然后再加一些实例的操作来帮助用户理解。

用法: POSCARtoolkit.py -i POSCAR_C [-f or -r] -s [your selections]

描述 1): POSCARtoolkit.py -i POSCAR_C 我们对输入文件 POSCAR_C 进行操作

描述 2): -f 和 -r 配合后面的 -s 进行操作。

-f 表示表示固定 (fix) 选中的原子，

-r 代表放开 (relax) 选中的原子，由于原子在 xyz 三个方向上都可以选择放开，所以使用 -r 的时候要配合 F T 来进行操作。如下：

-r FFT 代表只放开 z 方向，同理 -r TTF， -r TFT， -r TTT， -r FTT 这些你就知道是怎么回事了。

-r 后面的 FFT 这三个字母之间可以有空格，也可以没有。也就是说：

FFT 和 F FT， FF T 以及 F F T 效果是一样的。

描述 3) -s 选项表示选择部分原子，后面是你要选择的原子，选择项如下：

all 表示选中所有原子

1-5 6 9 表示选中 第 1-5 个和 6,9 号原子

Pt 表示选中所有的 Pt 原子

1-5 6 9 Pt 表示选中 第 1-5 个和 6,9 号原子和所有的 Pt 原子

描述 4) 其他未选择的原子，限制信息保持不变。

实例操作 1: 如果我们想固定 POSCAR_C 中所有的原子：

POSCARtoolkit.py -i POSCAR_C -f -s all

注释: -f 表示 fix， -s all 表示选择所有的原子，

POSCAR_C 中必须有 “Selective dynamics” 这一行

```
C:\Users\ppliv\Desktop>python POSCARtoolkit.py -i POSCAR_C -f -s all
```

```
#####  
# For VASP 5.2 or higher versions #  
#       Author:Li,Q;Xu,N         #  
#       Version 1.2              #  
#####
```

```
File to be handled is *****POSCAR_C*****
```

```
-----  
Cartesian Coordinates found, only for fixing atoms!  
Then atoms selected will be fixed.  
-----
```

```
-----  
POSCAR with Cartesian Coordinations is named as POSCAR_C_C
```

```
Atom fix completed, if layers are not separated properly,  
Please alter thresholds with flags -y or type -h for help!  
-----
```

实例操作 2: 将所有的原子只在 z 方向上放开:

```
POSCARtoolkit.py -i POSCAR_C -r FF T -s all
```

注释: 放开 z 方向上使用 -r FF T, FFT 之间有无空格均可。

```
C:\Users\ppliv\Desktop>python POSCARtoolkit.py -i POSCAR_C -r FF T -s all
```

```
#####  
# For VASP 5.2 or higher versions #  
#       Author:Li,Q;Xu,N         #  
#       Version 1.2              #  
#####
```

```
File to be handled is *****POSCAR_C*****
```

```
Your selective choice is ***** FFT *****!
```

```
-----  
Cartesian Coordinates found, only for relaxing atoms!  
Then atoms selected will be relaxed.  
-----
```

```
-----  
POSCAR with Cartesian Coordinations is named as POSCAR_C_C
```

实例操作 3: 将所有的 C, H, O 原子在 xy 方向上放开:

```
POSCARtoolkit.py -i POSCAR_C -r TTF -s C H O
```

实例操作 4: 将所有的 Pt 原子和 40 号原子在 z 方向放开:

```
POSCARtoolkit.py -i POSCAR_C -r FFT -s Pt 40
```

6 批量转化:

前面介绍的都是针对一个文件进行操作, 由于固定原子层数的功能我们设置了一个交互, 需要用户指定需要固定的层数 (z 轴从下到上)。但这对于层数批量固定操作来说是个累赘, 我们不想每操作一个文件就输入一次层数。所以**如果想跳过交互, 可以通过管道连接符 | 实现**。比如固定底部 4 层: `echo 4 | python POSCARtoolkit.py -i CONTCAR -f`

知道了这一点, 我们就可以通过一个 for 循环进行批量固定层数的操作了。

比如我们有 POSCAR1, POSCAR2 到 POSCAR100 个文件,

1) 我们想批量固定它们的底部 3 层。

```
for i in POSCAR{1..10}; do echo 3 | python POSCARtoolkit.py -i $i -f; done
```

2) 我们想批量将 C H O 原子在 z 方向上放开, zy 方向上固定。(这个不需要管道连接符)

```
for i in POSCAR{1..10}; do python POSCARtoolkit.py -i $i -r FFT -s C H O; done
```

7 评价:

这个脚本在模型操作的过程中, 实用性很强, 也很方便。实在是 VASP 计算中的一大利器。

本脚本是群友 [tamas-zju-VASP](#) (小 Tamas) 费煞苦心完成的。如果感觉不错, 欢迎打赏。