

LASP使用入门8-编写脚本使用外部程序计算能量

Lasp.in

```
potential external
External.script   lasp.external.sh

explore_type ssw

Run_type 15      #      15 for SSW with v
SSW.Crystal_extra_step T

SSW.SSWsteps 1      # 0 single point
                   # 1 stru opt
                   # > 1 SSW global

# output
SSW.output T
SSW.printevery T
~
```

- potential external
使用外部程序计算能量
- External.script lasp.external.sh
计算能量的外部脚本文件名，在作业目录中
(默认为lasp.external.sh)
- 使用方法：串行运行lasp程序
\$ /xxx/xxx/xx/lasp
- 外部程序的并行运算在lasp.external.sh中实现

LASP使用入门8-编写脚本使用外部程序计算能量

当使用“external”关键词时，程序会输出external.coord文件，该文件记录了结构信息。格式如下所示，单位为Å。

external.coord

```
[root@node200 varcell-opt]# cat external.coord
  9.9837890828    -0.0121117269    0.0014524903
 -0.0121106163    9.9872370732   -0.0005931053
  0.0014511626   -0.0005909443    9.9834700806
H      9.6255395562    1.1470322347    0.2084619492    1
H      2.2654172027    7.8254639239    0.0670656523    2
H      9.5133321679    0.5318202520    1.8712028553    3
H      8.5195433374    9.7816469869    0.5884174033    4
C      1.4797877240    8.5181404733    0.2796863964    5
C      0.5851214027    9.2938699369    0.5281289162    6
C      9.5133833626    0.2353001411    0.8128794454    7
N      1.6908973474    1.6898260917    8.3473739659    8
N      1.7358098254    2.0170399842    7.2821656366    9
```

晶胞矩阵 $a(x, y, z)$
 $b(x, y, z)$
 $c(x, y, z)$

原子坐标

元素符号

直角坐标 x y z

原子序数

LASP使用入门8-编写脚本使用外部程序计算能量

外部程序执行完之后，应该生成“external.ene”文件，该文件记录了能量，力和应力。格式如下。能量单位为eV，力的单位为eV/Å，应力单位为eV/Å³。LASP将读入该文件。

external.ene

```
[root@node200 varcell-opt]# cat external.ene
-56.43617084
-0.007803 0.015615 -0.008783
0.023107 -0.015220 -0.006609
0.008895 0.006302 0.035834
-0.037494 0.005672 -0.014869
0.068528 -0.096523 -0.005443
-0.052428 0.052567 0.014929
-0.016659 -0.010732 0.017194
0.002815 0.079204 -0.051676
0.011038 -0.036886 0.019424
-.00032223617459440454 -.00010547993369832738 -.00007130443518006931
-.00010547993369832738 -.00032442613893214124 -.00000625847606610075
-.00007130443518006931 -.00000625847606610075 -.00045143402481289868
```

- 第一行为能量

- 每个原子的力，共3n个（n为原子数，读入顺序为fx(1), fy(1), fz(1), fx(2), fy(2), fz(2)

应力

LASP使用入门8-编写脚本使用外部程序计算能量

lasp.external.sh脚本实例1：调用VASP，该脚本在Example/lasp-external/external-vasp

```
#!/bin/bash

#---Provide any necessary defination here -----
exec='/home7/bin/vasp-original-node200'
nprocs=`cat ./nprocs`
machinefile=./machinefile

#---Prepare the "parameter" input file here ----
if [ ! -f INCAR.lasp.external ]; then
  cp INCAR INCAR.lasp.external
  sed -i '/ISTART/d' INCAR.lasp.external
  sed -i '/ICHARG/d' INCAR.lasp.external
  sed -i '/LWAVE/d' INCAR.lasp.external
  sed -i '/LCHARG/d' INCAR.lasp.external
  sed -i '/IBRION/d' INCAR.lasp.external
  sed -i '/NSW/d' INCAR.lasp.external
  sed -i '/ISIF/d' INCAR.lasp.external
  sed -i '/ISYM/d' INCAR.lasp.external
  sed -i '/NWRITE/d' INCAR.lasp.external
  echo "ISTART = 1" >> INCAR.lasp.external
  echo "ICHARG = 1" >> INCAR.lasp.external
  echo "LWAVE = .TRUE." >> INCAR.lasp.external
  echo "LCHARG = .TRUE." >> INCAR.lasp.external
  echo "IBRION = 0" >> INCAR.lasp.external
  echo "ISYM = 0" >> INCAR.lasp.external
  echo "NSW = 0" >> INCAR.lasp.external
  echo "ISIF = 3" >> INCAR.lasp.external
  echo "NWRITE = 2" >> INCAR.lasp.external
fi
\cp -f INCAR.lasp.external INCAR

#---Prepare the "coordinate" input file here ---
echo " For LASP external use" >POSCAR
echo "1.0000" >>POSCAR
echo "2.0000" >>POSCAR
```

- 脚本执行程序
- 定义必要的变量，比如VASP的地址。VASP并行计算的核数，machinefile等。
- 对输入文件的参数部分进行调整，总体来说要实现如下功能：
 1. 重用上次的电子结构进行SCF计算
 2. 进行单点计算
 3. 如果要对晶胞进行优化需要计算应力
 4. 进行必要的输出（能量，力，应力）
- 如果准备好的INCAR中已经满足上述要求，这一部分原则上可以为空白

LASP使用入门8-编写脚本使用外部程序计算能量

```
#---Prepare the "coordinate" input file here -----
echo " For LASP external use" >POSCAR
echo "1.0000" >>POSCAR
head -3 external.coord >>POSCAR
a=(`grep VRHFIN POTCAR |awk '{print $2}'|sed 's/=//'|sed 's/://'\` )
echo ${a[*]} >> POSCAR
j=0
\rm -f lasp.coord.tmp
for i in ${a[*]}
do
    b[$j]=`grep "$i " external.coord|wc -l`
    grep "$i " external.coord >>lasp.coord.tmp
    let "j=j+1"
done
echo ${b[*]} >>POSCAR
echo "C" >>POSCAR
cat lasp.coord.tmp|awk '{print " "$2" "$3" "$4}' >>POSCAR

#---Run the executable file here -----
if [ "$1" == "T" ]; then \rm -f CHGCAR WAVECAR;fi
mpirun -machinefile ./machinefile -np $nprocs $exec
```

- 利用external.coord文件生成VASP的结构输入文件POSCAR
- 需要注意的是POSCAR是根据POTCAR的元素顺序排序的，因此或者在准备lasp.str文件时就预先调整好原子顺序，或者在生成POSCAR时要记住原子在external.coord中的顺序。

- 调用VASP进行单点能计算。优化晶胞会导致VASP的计算误差，因此LASP认为有必要初始化电子结构计算时会在第一参数位置输出“T”，此时应删除CHGCAR和WAVECAR。（ " \$1" == " T " ）

LASP使用入门8-编写脚本使用外部程序计算能量

```
#---Extract energy here -----
grep TOTEN OUTCAR |tail -1|awk '{print $5}' >external.ene

#---Extract force here -----
n=`cat lasp.coord.tmp |wc -l`
for ((i=1;i<=$n;i++))
do
    echo f.$i.f >>external.ene
done
let "n=$n+1"
grep -A$n TOTAL-FORCE OUTCAR |sed 1,2d|awk '{print $4" "$5" "$6}'>lasp.force.tmp
j=1
for i in `cat lasp.coord.tmp|awk '{print $NF}'`
do
    f=`sed -n "$j"p lasp.force.tmp`
    sed -i "s/f.$i.f/$f/" external.ene
    let "j=$j+1"
done

#---Extract stress here -----
c=(`grep 'FORCE on cell' OUTCAR -A13|tail -1|awk '{$1=""}1'`)
v=`grep "volume of cell" OUTCAR|tail -1|awk '{print $NF}'`
for i in {0..5}
do
    c[$i]=`echo "${c[$i]}/$v"|bc -l`
done
echo ${c[0]} ${c[3]} ${c[5]} >>external.ene
echo ${c[3]} ${c[1]} ${c[4]} >>external.ene
echo ${c[5]} ${c[4]} ${c[2]} >>external.ene
```

- 从OUTCAR中提取能量
- 从OUTCAR中提取力
- 注意如果之前有调整原子顺序，则需要调整回来。
- 从OUTCAR中提取应力

LASP使用入门8-编写脚本使用外部程序计算能量

lasp.external.sh脚本实例1：调用Gaussian，该脚本在Example/lasp-external/external-gaussian

```
#!/bin/bash

#---Provide any necessary definition here -----
exec='g09'
inputfile='gaussian.inp'
outputfile='gaussian.out'
charge=0
multi=1
#mpirun=''
#machinefile=''
#nprocs=''

#---Prepare the "parameter" input file here -----
cat $inputfile.pre >$inputfile
a=`grep '%chk' $inputfile.pre|sed 's=/ /g'|awk '{print $2}'`
if [ -f "$a" ]; then echo "Guess=Read">>$inputfile;fi
echo "" >>$inputfile
echo "lasp_external" >>$inputfile
echo "" >>$inputfile
echo $charge " " $multi >>$inputfile

#---Prepare the "coordinate" input file here -----
sed 1,3d external.coord|awk '{print $1,$2,$3,$4}' >>$inputfile
echo "" >>$inputfile
if [ -f "$inputfile.after" ]; then
  cat $inputfile.after >>$inputfile
fi
echo "" >>$inputfile
echo "" >>$inputfile
```

- 将体系的电荷和多重度定义在这里
- 由于gaussian的参数和结构都定义在同一个文件中。这里将输入文件拆分成3部分：
 1. 头部为gaussian.inp.pre，预先准备好
 2. 中部为电荷、多重度和结构，动态生成
 3. 尾部为gaussian.inp.after，包含其他信息，如溶剂化参数。如果没有可以不准备。

LASP使用入门8-编写脚本使用外部程序计算能量

```
echo "" >>$inputfile
#---Run the executable file here -----
\rm -f $outputfile.old;mv $outputfile $outputfile.old
$exec <$inputfile >$outputfile

#---Extract energy here -----
grep "Total Energy" Test.FChk|awk '{printf("%.15f\n", $4*27.211383)}' >external.ene

#---Extract force here -----
b=`grep -ml -n 'Cartesian Gradient' Test.FChk|sed 's/:/ /'|awk '{print $1}'`
c=`grep -ml -n 'Cartesian Force Constants' Test.FChk|sed 's/:/ /'|awk '{print $1}'`
let "d=$c-$b-1"
grep -ml -A$d 'Cartesian Gradient' Test.FChk|sed 1d|awk -v m=-51.422057 '{printf("%.15f\n", $4*m)}' >external.frc

#---Extract stress here -----
#---Cleanance -----
```

- 注意进行单位转换
- Gaussian无应力计算，可以空白。

LASP使用入门8-编写脚本使用外部程序计算能量

关于队列系统：如果使用队列系统进行作业管理，则需要从队列系统获得资源信息并传递给执行脚本，以超算系统中常用的PBS队列系统为例：

```
#!/bin/bash -x

#PBS -l nodes=1:ppn=20
#PBS -j oe
#PBS -N lasp-pbs
#PBS -q example

cd $PBS_O_WORKDIR

cat $PBS_NODEFILE | wc -l > ./nprocs
cat $PBS_NODEFILE > ./machinefile

exec=/xxx/xx/xx/lasp
./$exec
```

- 向队列系统申请资源，该资源用于外部程序（如VASP，SIESTA）的并行运算

- 将总核数记录下来
- 将分配的节点记录下来
- 定义LASP可执行文件的路径
- 串行运行LASP

- nprocs, machinefile用于lasp.external.sh中执行mpirun（见P4，P5）

LASP使用入门8-编写脚本使用外部程序计算能量

注意事项：

1. 应力和力的符号！有些代码输出的梯度，力为梯度的负数！最好自己测试一下。
2. 注意单位转换
3. 由于硬盘读写速度远慢于cpu运算速度，因此该方法不适用于运算量小的力场计算。
4. 由于是从输出文件中读取能量和力，因此精度受输出格式的影响。
5. 可以用任何语言实现，如python, fortran, c++等。如果是脚本语言，为以防万一需在第一行定义脚本运行程序。