

LASP: Large-scale Atomic Simulation with neural network Potential

Tutorial

Cheng Shang, Zhipan Liu

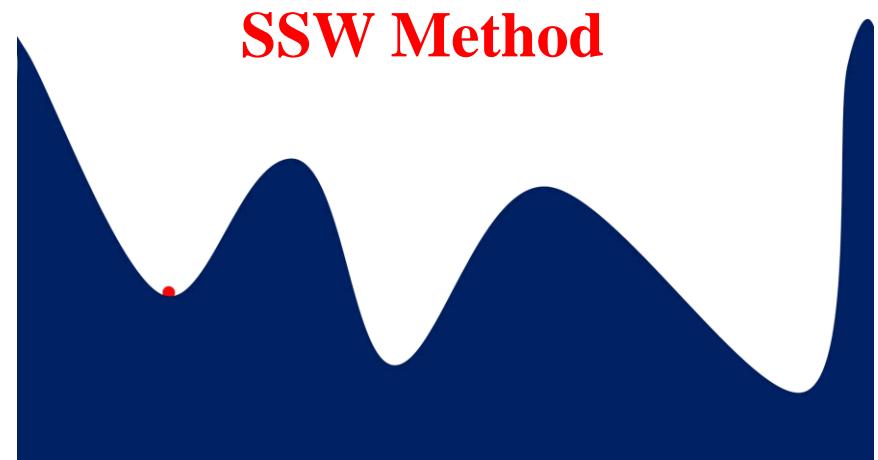
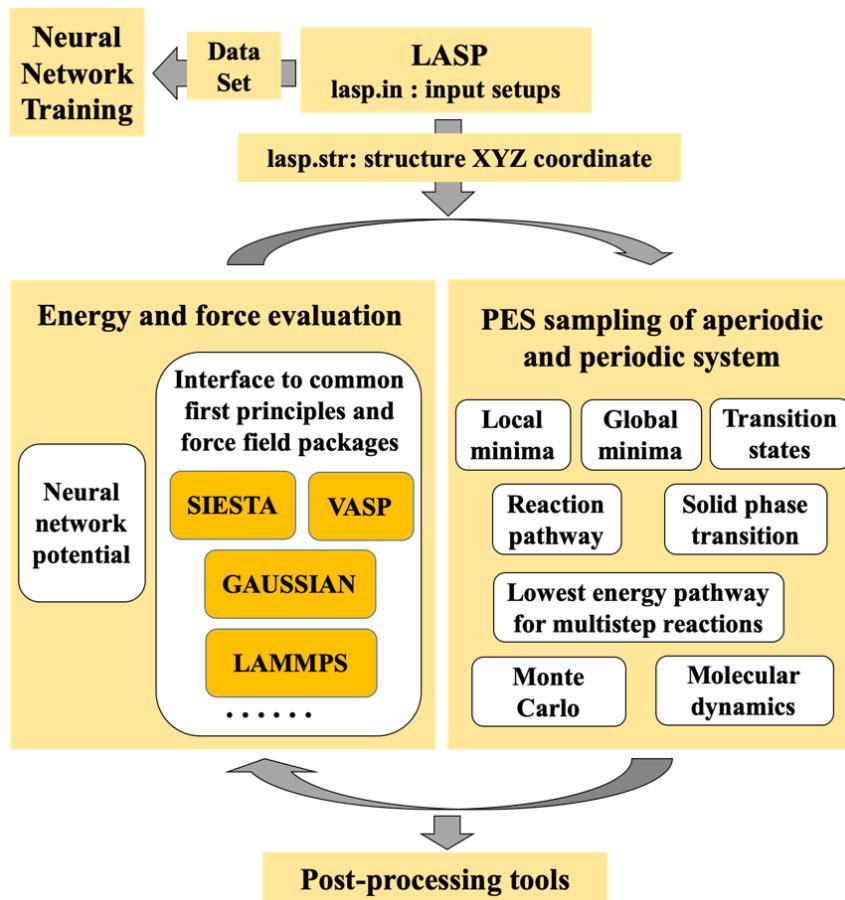
Department of Chemistry, Fudan University
Shanghai, China



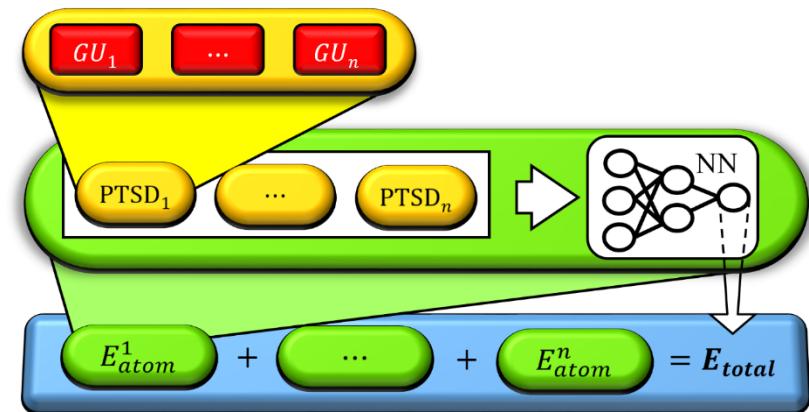
Content:

- **What is LASP**
- **How to get**
- **How to compile**
- **How to run**
- **How to discuss**

Architecture and the modular map of LASP



Neural Network Potential



www.lasphub.com

LaSpHUB

Large-scale Atomic Simulation with neural network Potential

English | 中文 | Register | Login

[HomePage](#)

[Software Introduction](#)

[User Manual and](#)

[Examples](#)

[NN Library](#)

[LASP download](#)

[LASP theoretical](#)

[background](#)

[LASP applications](#)

New design of organic reactions and drugs using
machine learning

[LASP tutorial](#)

New design of solid materials using
machine learning

LASP Software Platform

----- For solving complex problems in materials and chemistry.

Molecules

Solids

最近新闻

- (2019-7-11) LASP 2.1.7 released
- (2019-6-20) LASP 2.1.6 released

www.lasphub.com

LaSpHUB

Large-scale Atomic Simulation with neural ne

HomePage

Sofware Introduction

User Mannual and

Examples

NN Library

LASP download

LASP theoretical

background

LASP applications

LASP tutorial

LASP Software Download

Welcome

Expire on the 15th of the next month

LASP Academic Version

LASP Professional Version

laspmaster@lasphub.com

LASP TEAM @ 2013-2018

The quick links for useful resources are given below

LASP Manual can be found here ([pdf download](#))

LASP Examples can be found here ([download](#))

Free-trial NN potentials can be found here ([download](#))

| Version | PES | System |
|---|--------------------------------|--------|
| LASP 2.0-ac MPI for multi processors | NN | Linux |
| LASP 2.0-ac MPI for multi processors | Expert (Interface Sourcecodes) | Linux |

Pre-compiled, use directly



- ◆ NN potential & external script (e.g. to run Gaussian)
- ◆ The executable file: `./Src/lasp`

Compile before use



- ◆ NN potential & external script (e.g. to run Gaussian)
- ◆ VASP / SIESTA / LAMMPS / CP2K / D3
- ◆ Other program (Do some coding)

LASP 2.0-ac

MPI for multi processors

Expert (Interface Sourcecodes)

Linux

Intel MPI

(4.0.0.028 or above)

◆ NN potential & external script (e.g. to run Gaussian)

- Go to ./Src
- check \$MKLROOT environment variable (**echo \$MKLROOT**)
- make

LASP 2.0-ac

MPI for multi processors

Expert (Interface Sourcecodes)

Linux

Intel MPI

(4.0.0.028 or above)

◆ VASP / SIESTA / LAMMPS / CP2K (Follow the “README” instruction)

1. Already build directories

```
0.NNlib          2.siestalib   5.cp2klib
103.SSWlib-OOP  3.Lammps      README
1.vasplib        4.d3-lib
```

2. Go to the corresponding
directories, e.g. 1.vasplib

```
genlib.sh        makefile.vasp
makefile.lib     patch.vasp.tar.gz
```

3. Copy code to the
corresponding directories

```
genlib.sh        patch.vasp.tar.gz
makefile.lib     vasp.5.3.5.tar.gz
makefile.vasp    vasp.5.lib.tar.gz
```

4. Run ./genlib.sh

| | | | |
|------------------------|--------------------------|--------------------------|--------------------------|
| genlib.sh | makefile.lib | vasp.5.3 | vasp.5.lib.tar.gz |
| libvasp.a | makefile.vasp | vasp.5.3.5.tar.gz | |
| libvasp-gamma.a | patch.vasp.tar.gz | vasp.5.lib | |

5. Go to ./Src
make clean
make

```
mpiifort -I../Libraries/103.SSWlib-OOP -I./modules -I/home7/intel19/compilers_and_libraries_2019.2.187/linux/mkl/include -I../Libraries/0.NNlib/cpp -I../Libraries/0.NNlib/strucdescrip -I../Libraries/1.vasplib/ -O2 -fPIC -DMPI -DNNN -DVASP -c otherpot.F90 -o otherpot.o
```

LASP 2.0-ac

MPI for multi processors

Expert (Interface Sourcecodes)

Linux

Intel MPI

(4.0.0.028 or above)

◆ Other program

➤ other energy functions + SSW: Modify otherpot.F90

```
#ifdef SIESTA
case("siesta")
  call siesta_forces( 'siesta', strt%nalocal, strt%xcart, strt%lat, strt%ene, strt%fcart, strt%stress33)
#endif
```



➤ NN + other move functions: Modify othermove.F90

```
case("mymove")
  call mymove(strt%nalocal,strt%ene,strt%fcart,strt%stress33,strt%xcart,strt%lat)
```



<http://www.lasphub.com/tutorial/LASP-lessons.pdf>

Input files

General input:

lasp.in or input (parameter)
lasp.str or input.arc (initial structure)

Energy computation related:

xxxx.pot (NN)
INCAR, POTCAR (VASP)
in.simple (LAMMPS)
lasp.external.script (external)
.....

PES exploration related:

uncm.arc (TS search)
Rigidbody (rigidbody)
.....

Training related:

xxx.input (input pot)
adjust_factor (weight parameter)
TrainStr.txt, TrainFor.txt (training set)

Output files

General output:

lasp.out
allkeys.log

PES exploration related:

SSWtraj
all.arc
SSWPath.arc, TSmode.arc
allstr.arc, allfor.arc
md.arc
.....

Energy computation related:

OUTCAR etc.

Training related:

xxx.pot

lasp.in: input parameters

Free format; Case insensitive; Order insensitive

```
potential NN
explore_type sswoop
Ewaldflag 0
%block netinfo
Co Co0.pot
0 Co0.pot
%endblock netinfo
PrintChg 0

Run_type 15 # 12 for VC_DESW
# 15 for SSW_crystal
# 2 for DESW
# 5 for SSW
# 6 for ranMC_SSW
# 16 for ranMC_SSW_crystal

SSW.SSWsteps 10000 # 0 single point
# 1 stru opt
# > 1 SSW global opt

# for SSW search start
SSW.Temp 100
SSW.NG 10
SSW.NG_cell 8
SSW.inirot 1
SSW.ftol 0.05
SSW.strtol 0.05
```

Potential select Energy evaluation module

(NN / US / external) / VASP / SIESTA / LAMMPS / D3
(Multiple values are allowed, e.g. potential NN D3)

Explore_type select PES exploration/function module

SSW / rigidssw / NVE / NVT / NPT / train / test

lasp.str: input coordinate

```

!BIOSYM archive 2
PBC=ON
!DATE
PBC    Energy      5243      0.0433     -226.948405
Co    8.4709    5.6825   11.2059   90.0004   67.7928   89.9986
Co    5.865352171  3.542956874 10.341614008 CORE   1 Co Co  0.0000   1
Co    6.288756626  0.701772230 3.079438787 CORE   2 Co Co  0.0000   2
Co    3.747541522  0.701736357 5.154441268 CORE   3 Co Co  0.0000   3
Co    2.053402560  4.490091423 3.079723819 CORE   4 Co Co  0.0000   4
Co    4.170952429  1.648821996 8.267233490 CORE   5 Co Co  0.0000   5
Co    8.406643357  3.542883680 8.266704055 CORE   6 Co Co  0.0000   6
Co    11.371330527 4.489897261 9.304472195 CORE   7 Co Co  0.0000   7
Co    6.712327345  1.648772360 6.192408108 CORE   8 Co Co  0.0000   8
Co    9.677306957  0.701674982 7.229479027 CORE   9 Co Co  0.0000   9
Co    7.551622523  4.489918211 9.813652366 CORE  30 0 0  0.0000  30
Co    4.587096585  3.542863833 8.776603246 CORE  31 0 0  0.0000  31
Co    0.774875744  4.490051678 1.513925691 CORE  32 0 0  0.0000  32
Co    4.163232059  4.490009218 5.663787177 CORE  33 0 0  0.0000  33
Co    2.469420179  0.701713885 3.589233130 CORE  34 0 0  0.0000  34
Co    5.857769253  0.701651398 7.739090701 CORE  35 0 0  0.0000  35
Co    6.704545848  4.490007896 3.588772315 CORE  36 0 0  0.0000  36
Co    6.281438426  3.542973686 0.476852063 CORE  37 0 0  0.0000  37
end
end

```

**.arc file is a format utilized by Material Studio
(export as .car)**

follow .arc file format , but much simplified (e.g: only read the first 4 columns of atomic information)

Line 1-4: information lines

Line 5 : Heading by “PBC”, followed by lattice parameters:
“ a b c α β γ ”

Line 6---end:
Element name Cartesian coordinate x , y z

End lines:
End
End

lasp.str: input coordinate

This is also a legal lasp.str !

```
[shang@storage3 opt.3]$ cat lasp.str
11---casda@!~
aasdfsd fdsfasdfadf
.....
xxxxxxxxx333333333333
PBC    26.71   19.7   23.0   90.0   90.0   90
O31      24.934560   0.685651757   2.300
          022      13.060960   13.824584019   2.300
O  21.966160   13.824584019   2.300
```

1. Everything before the “PBC” line will be ignored.
2. The coordinates lines must follow the PBC line without any space in between !!!
3. There can be numbers behind the element name, which will be ignored.
4. Everything within the line and behind the xyz coordinates will be ignored.
5. Aligning is unnecessary (free format)
6. The word “End” notes the end of the structure input, which can be replaced by spaces.

How to run

LaSpHUB Large-scale Atomic Simulation with neural network Potential

lasp.out

```
+-----+  
| Welcome to Fantastic Atomic World in Neural Network  
| -----  
| L A S P  
| -----  
| Large-scale Atomic Simulation with neural network Potential  
| Version 2.1.7 JUL. 2019  
| Shanghai, China  
| More on Website: www.lasphub.com  
|  
| Please cite: S.D. Huang, C. Shang, P.L. Kang, X.J. Zhang and Z.P. Liu ,  
| "LASP: Fast Global Potential Energy Surface Exploration" ,  
| WIREs. Comp. Mole. Sci. 2019;e1415 (DOI: 10.1002/wcms.1415)  
|  
| Compiled by IntelFortran version: ifort 18  
| CPU Parallel Run on 14 cores  
+-----+
```

```
+-----+  
+ The PES is calculated by  
+ nn  
+-----+  
|*****|  
| Welcome to SSW package  
| Stochastic Surface Walking Global Optimization 2.0  
| ----- now as a part of LASP program  
| Fudan University Liu group 2013-2019  
| Developers: Cheng Shang, Xiaojie Zhang, Sida Huang, Zhipan Liu |  
|*****|
```

```
SSW Job Start at 17:32:33 27th August 2019  
--Key Parameter from input: lasp.in  
-->System size: Number of atoms 14  
-->Fix Cell Structure Search: Run_type=5  
-->SSW quick_setting Level 1  
-->Number of Gaussian for atomic move 9  
-->Monte Carlo Temperature 100.00  
-->Gaussian ds for each atomic move 0.60  
-->Maximum Localmode-to-Globalmode ratio 50  
-->Maximum SSW steps 3  
--Please also check allkeys.log for SSW Parameters
```

```
SSW cycles start now
```

```
|*****|  
Str symm and Q 0 0 longQ T -78.119506 C1 1.2412 0.974  
adding internal LJcore Repulsion  
Str symm and Q 0 0 longQ T -79.252935 C1 1.1965 0.913  
Minimum found 0 0 -78.119506 -79.252935 C1 1.1965 0.913  
Str symm and Q 1 0 longQ T -79.252935 C1 1.1965 0.913  
Str symm and Q 1 0 longQ T -79.100590 Cs 0.9377 0.827  
Minimum found 1 0 -79.252935 -79.100590 Cs 1.1965 0.913  
Str symm and Q 2 0 longQ T -79.252935 C1 1.1965 0.913  
Str symm and Q 2 0 longQ T -79.257418 C1 1.2107 0.909  
Minimum found 2 0 -79.252935 -79.257418 C1 1.2107 0.909  
SSW all done !  
SSW Job Stop at 17:32:34 27th August 2019  
elapsed time 1.102000000000000 seconds
```

allkeys.log

```
explore_type ssw 3  
SSW.SSWsteps 5  
Run_type MD.dt 1.000000000000000  
SSW.mdfreq 0  
potential nn  
Symm_lattice F  
Bravice_cell F  
D3.func pbe  
Run_type 5  
SSW.mdfreq 0  
potential nn 1  
SSW.quick_setting DESW.quick_setting 1  
SSW.LSSWandPath F  
SSW.NG 9  
SSW.NG_cell 7  
SSW.Temp 100.0000  
SSW.LowTemp 100.0000  
SSW.HighTemp 100.0000  
SSW.TempCycle 500  
SSW.inirot 0  
SSW.iniopt T 150.0000  
SSW.Safe_hardcurv  
SSW.internal_LJ T  
SSW.linitBravis T  
SSW.Crystal_extra_step F  
SSW.TS_extra F  
SSW.printdelay -1  
SSW.printevery F 1  
SSW.printselect 0  
SSW.printformat  
SSW.output F  
SSW.SSWsteps 3  
SSW.ImaintainXYZ F  
SSW.ds_atom 0.6000  
SSW.lrandom_ds F  
SSW.ds_cell 0.5339  
SSW.DimerdR 0.0100  
SSW.ftol 0.1000  
SSW.strtol 0.1000  
SSW.pressure 0.0000  
SSW.cellmode_factor 0.1250  
SSW.cell_allowexpand 0.5000  
SSW.globalcompress 0.5000  
SSW.lmode_Q T  
SSW.mode_Q_pro 0.5000  
SSW.Ratio_local 50  
SSW.Ratio_atomcell 2  
SSW.Modelevel 0 0  
SSW.Modelevel_cell 15  
SSW.CBD_optimizer broyden
```

Version

Potential type

SSW main parameters

Results

Elapse time

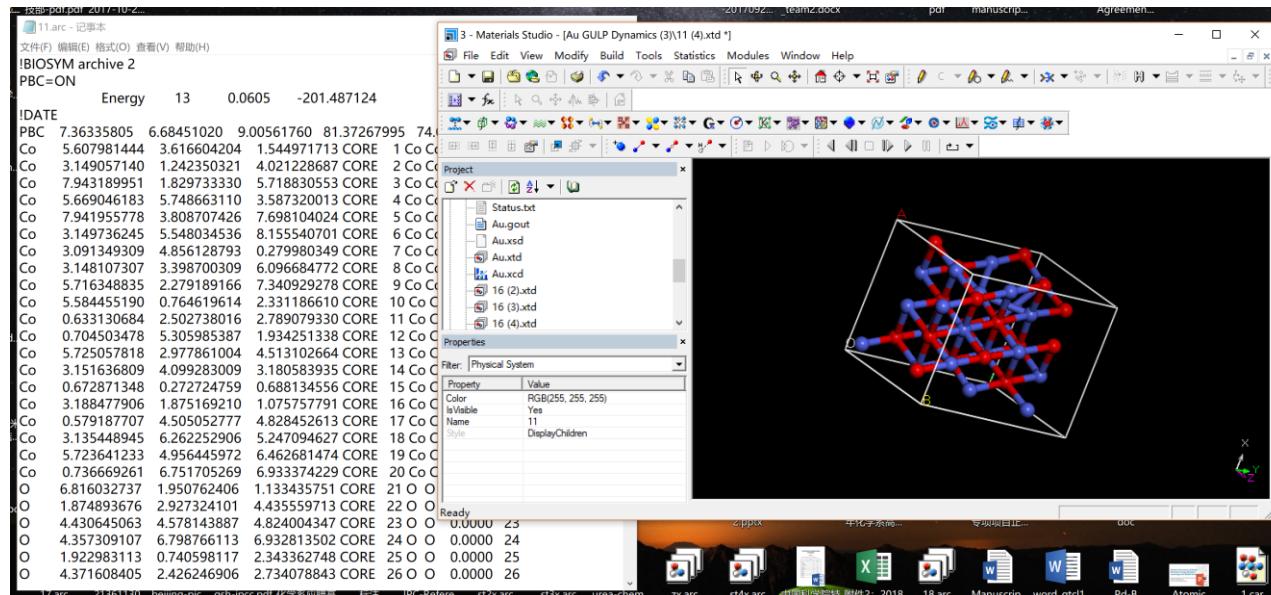
How to run

LaSpHUB Large-scale Atomic Simulation with neural network Potential

View structures, including all.arc, best.arc, allstr.arc

Use Material Studio (WINDOWS system) to view all .arc file (animation)

Edit the file using notepad in windows



A home made visualizer is on the way!

<http://www.lasphub.com/Examples.asp>

LASP_Example

Different Energy Function

```
[shang@storage3 LASP_Example]$ ll
total 36
drwxr-xr-x  9 shang shang 4096 Aug 24 09:51 VASP
drwxr-xr-x  6 shang shang 4096 Aug 24 09:51 SIESTA
-rw-r--r--  1 shang shang 5408 Aug 24 09:51 README
drwxr-xr-x  4 shang shang 4096 Aug 24 09:51 LAMMPS
drwxr-xr-x  4 shang shang 4096 Aug 24 09:51 CP2K
drwxr-xr-x  2 shang shang 4096 Aug 24 09:51 allexamples
drwxr-xr-x 21 shang shang 4096 Aug 24 09:51 NN
drwxr-xr-x  5 shang shang 4096 Aug 24 09:51 lasp-external
```

Different purpose

```
[shang@storage3 NN]$ ls
constrained_SSW                               NN-12  NN-8
molecule-ConfirmPathway-ExtrapolatefromTS    NN-13  NN-9
molecule-fixcell-SSW                           NN-14  NNpot_lib
molecule-MD-verlet                            NN-15  rigidssw-urea
molecule-PathwaySampling-SSWRS                NN-16  slab-global-search
Molecule-ReactPattern                         NN-17  solid-MD-NPT
molecule-TS-CBD                                NN-2   solid-MD-NVT
molecule-TS-DESW                               NN-3   solid-PathwaySampling-SSWRS
molecule-varcell-SSW                           NN-4   solid-TS-VCDESW
NN-1                                         NN-5   solid_umbrella_sampling
NN-10                                         NN-6   surface_constrained_MD
```

sourcedir: The input files to run the job

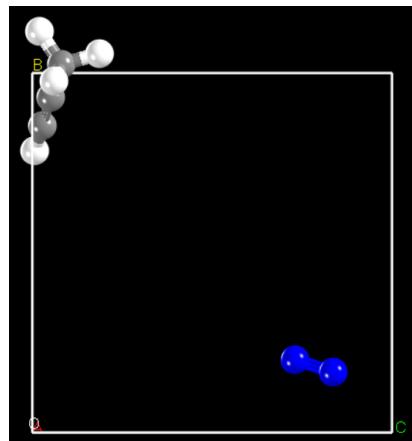
```
[shang@storage3 NN-2]$ ll
total 4
drwxr-xr-x 2 shang shang 4096 Jul  1 13:09 sourcedir
[shang@storage3 NN-2]$ cd sourcedir/
[shang@storage3 sourcedir]$ ll
total 12
-rw-r--r-- 1 shang shang 1344 Dec  3  2018 input.arc
-rw-r--r-- 1 shang shang   974 Dec  3  2018 input
lrwxrwxrwx 1 shang shang     29 Dec 17  2018 CHON.pot -> ../../NN-1/sourcedir/CHON.pot
-rw-r--r-- 1 shang shang   513 Jul  1 13:09 lasp.in
```

To run an example, such as NN-1, please follow the steps below

1. cd allexamples/NN-1
2. cp sourcedir/* .
3. [DIR]/Src/lasp or mpirun -np 4 [DIR]/Src/lasp

Example1: VASP-2 -> fixcell-opt

Find the local minimum with BFGS and VASP



BFGS

| | | |
|---------------|---------------|-------------|
| Energy, force | -56.393357815 | 0.432901292 |
| Energy, force | -56.399692196 | 0.347740222 |
| Energy, force | -56.402598449 | 0.304194360 |
| Energy, force | -56.406247426 | 0.246017400 |
| Energy, force | -56.409357745 | 0.192021943 |
| Energy, force | -56.411604912 | 0.146839312 |
| Energy, force | -56.413068563 | 0.125130002 |
| Energy, force | -56.414198784 | 0.081241782 |

lasp.in

Key words: Explore_type ssw

Run_type 5

Fix lattice

Slabs and nonperiodic molecules / clusters

Run_type 15

Variable lattice

Crystals and periodic materials / polymols

Main Keywords in lasp.in

SSW.SSWsteps 1

Local optimization

(0 for single point energy calculation)

%block fixatom
%endblock fixatom

The fixed Atoms should be listed here,
replace the “T” and “F” set in POSCAR.

SSW.output T/F

Whether to print out the detail information

SSW.printevery T/F

Whether to print out the structure and corresponding
energy and forces at every step

Notice: For the first principle computation, we recommend “SSW.printevery T”
to obtain extra information that could be used for NN training later.

Keywords related to optimization

| | | |
|----------------------|------|---|
| SSW.ftol | 0.10 | Maximum atomic force criterion (eV/Å) |
| SSW.strtol | 0.10 | Maximum lattice stress criterion (GPa) |
| SSW.Bfgs_maxstepsize | 0.20 | Maximum stepsize of each BFGS step (Å) |
| SSW.MaxOptstep | 300 | Maximum number of steps of BFGS optimization |

Auto modification in INCAR

```
[shang@storage3 fixcell-opt]$ cat INCAR
SYSTEM = default
ISTART = 0
ICHARG = 2
ENCUT = 450
IALGO = 38
NELMIN = 4
EDIFF = 5E-06

LWAVE = .FALSE.
LCHARG = .FALSE.
IBRION = 22
ISYM = 0
NSW = 2000000
ISIF = 3
```

IBRION=22: Interface with lasp

ISYM=0: No symmetry computation

NSW=20000: Random number to avoid VASP convergance

ISIF=3: Stress computation

Those are done by LASP automatically.

POSCAR is generated from input.arc.

The sequence of the element is determined by the POTCAR.

```
[shang@storage3 fixcell-opt]$ grep TITEL POTCAR
TITEL = PAW_PBE H 15Jun2001
TITEL = PAW_PBE C 08Apr2002
TITEL = PAW_PBE N 08Apr2002
```

```
!BIOSYM archive 2
PBC=ON
          Energy      0      0.09
!DATE
PBC  10.00000000  10.00000000  10.00000000  90.00
H    2.271429996   7.841545730   0.069098053  0
N    1.737295586   2.026081333   7.300695534  0
H    9.525806043   0.542566739   1.869899556  0
C    1.487651089   8.538733595   0.283346108  0
C    0.599514280   9.328622931   0.513643812  0
H    8.537793076   9.778230322   0.603689836  0
C    9.513523265   0.247385841   0.809741234  0
N    1.694675668   1.682047792   8.360845219  0
H    9.681099987   1.163187738   0.203403658  0
end
end
```

```
From lsasp
1.00000000
 10.000000   0.000000   0.000000
  0.000000  10.000000   0.000000
  0.000000   0.000000  10.000000
        4     3     2
D
  0.968110   0.116319   0.020340
  0.227143   0.784155   0.006910
  0.952581   0.054257   0.186990
  0.853779   0.977823   0.060369
  0.148765   0.853873   0.028335
  0.059951   0.932862   0.051364
  0.951352   0.024739   0.080974
  0.169468   0.168205   0.836085
  0.173730   0.202608   0.730070
```

The sequence of atoms in the input.arc can be random

Lasp.out (SSW.output=T)

```

Allopt          bfgs_opt
-----
Energy,force   -56.402598449  0.304194360
LBFGS opt      0.304194     -56.402598    2
bfgs--stepsize, force 8.434216837374255E-003 0.304194359693863
-----
Allopt          bfgs_opt
-----
Energy,force   -56.406247426 ← 0.246017400
LBFGS opt      0.246017     -56.406247    3
bfgs--stepsize, force 8.701915145610946E-003 0.246017400432605
-----
Allopt          bfgs_opt
-----
Energy,force   -56.409357745  0.192021943 ←
LBFGS opt      0.192022     -56.409358    4
bfgs--stepsize, force 7.543639980110923E-003 0.192021943058029
-----
Allopt          bfgs_opt
-----
Energy,force   -56.411604912  0.146839312
LBFGS opt      0.146839     -56.411605    5
bfgs--stepsize, force 6.344474960473139E-003 0.146839311961290
-----
Allopt          bfgs_opt
-----
Energy,force   -56.413068563  0.125130002
LBFGS opt      0.125130     -56.413069    6
bfgs--stepsize, force 5.923989198667805E-003 ← 0.125130001949250
-----
Allopt          bfgs_opt
-----
Energy,force   -56.414198784  0.081241782
finish bfgs allopt 7 -56.4141987837861      8.124178244722251E-002
atoms away from least move atoms       6           5
-----
Allopt          bfgs_opt
-----
Str symm and Q   0   0 longQ T      -56.414199  C1   1.0450  1.0105  0.9975
Minimum found    0   0                  -56.393358  1
Next local atom pair 6   1           1
SSW all done !
SSW Job Stop at 10:38:12 3rd September 2019

```

Energy

Maximum force component

Stepsize

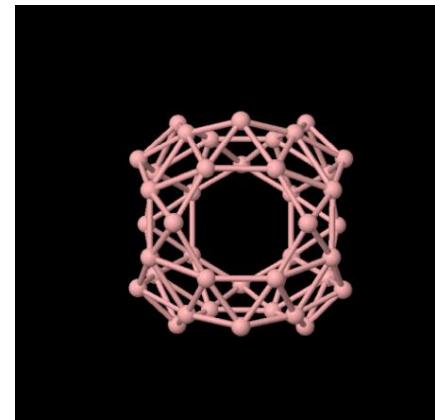
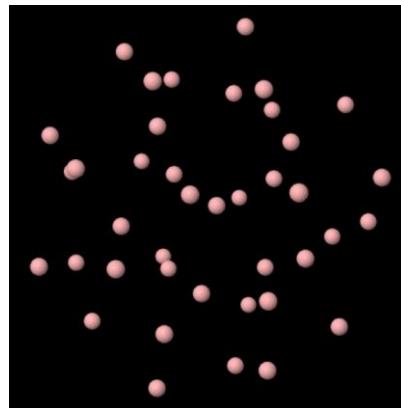
Order parameter

Converged energy

Symmetry

Example2: NN-2 -> molecule-varcell-SSW

Find the GM using stochastic surface walking method



lasp.in

Key words: `Explore_type ssw`

Run_type 5

Fix lattice

Slabs and nonperiodic molecules / clusters

Run_type 15

Variable lattice

Crystals and periodic materials / polymols

Main Keywords in lasp.in

SSW.SSWsteps 10000 Set the number of SSW steps

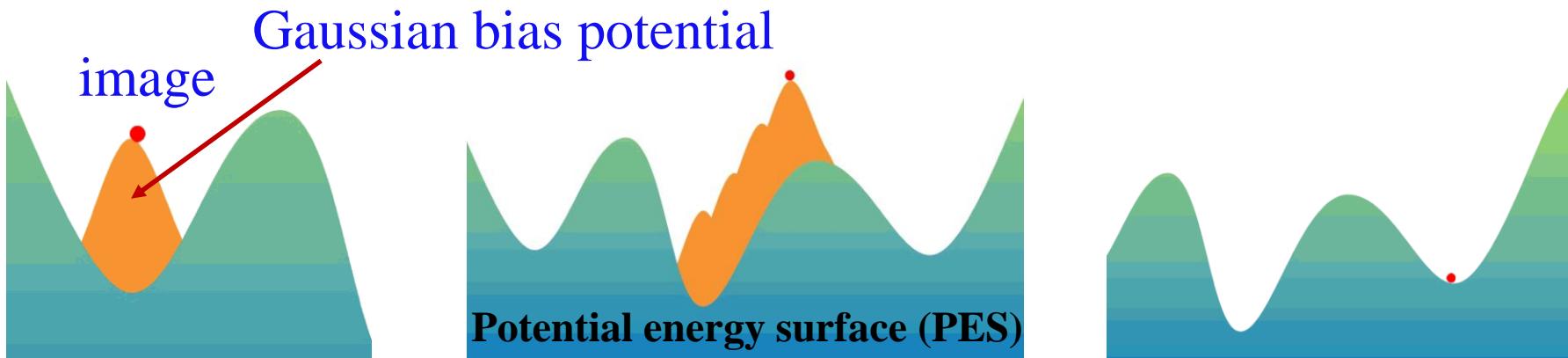
Note: global optimization is an NP-Hard problem. Massive SSW steps are required for large and complex systems to find the global minimum.

SSW.output T/F Whether to print out the detail information

SSW.printevery T/F Whether to print out the structure and corresponding energy and forces at every step

SSW.printselect 0--6 Set the print level if SSW.printevery is T. Large number generally corresponds to less output. The default is 6.

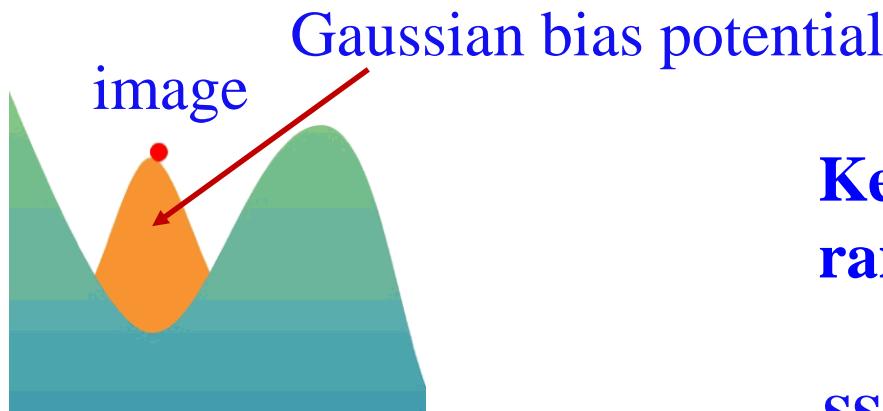
The mechanism of SSW method



Random reaction coordinate (mode)

Continuously add bias potential and local optimization to overcome barriers on the PES

Remove all the bias potential and optimize to find a local minimum. Monte Carlo is used to decide the acceptance.



Keywords related to the mode refinement:

```
SSW.DimerdR      0.0050
SSW.globalcompress 0.1000
SSW.Lmode_Q      T
SSW.Ratio_local    50
SSW.Rotftol_preRot 1.0000
SSW.Rotftol      0.1000
SSW.Rotftol_ini   1.0000
```

Control the accuracy of mode refinement

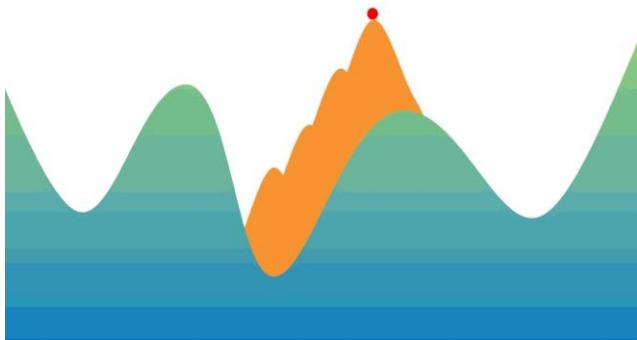
Keywords related to random mode generation:

SSW.globalcompress 0~1
Compress the configuration

SSW.Lmode_Q T/F
Order parameter component

SSW.Ratio_local 15~100
Local atomic movement

SSW.cell_allowexpand 0~1
Lattice expansion probability



Keywords related to climbing behavior of the image

| | |
|-------------|--------|
| SSW.NG | 10 |
| SSW.NG_cell | 8 |
| SSW.ds_atom | 0.6000 |
| SSW.ds_cell | 0.5000 |

SSW.ds_atom 0.3 ~1

Atomic movement of each gaussian

SSW.ds_cell 0.3 ~1.5

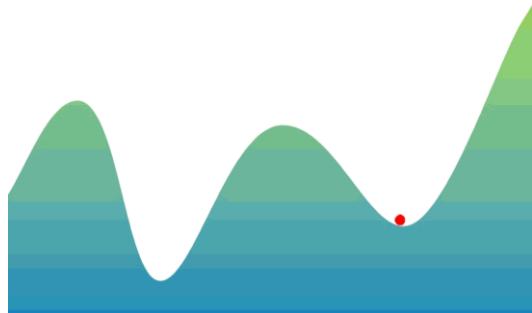
Lattice distortion of each gaussian

SSW.NG 6-15

Number of gaussian for atomic movement

SSW.NG_cell 4-11

Number of gaussian for lattice movement



Opt. accuracy and Accept Probability

SSW.ftol 0.01~0.1 eV/A

Maximum atomic force criterion

SSW.strtol 0.1~1 GPa

Maximum lattice stress criterion

SSW.Temp 10~500 K

Monte Carlo temperature

(Adjust Accept ratio 30%~50%)

SSW.LowTemp

SSW.HighTemp

SSW.TempCycle

Allow variable
temperature set

Keywords related to local optimization and MC

```
SSW.MaxOptstep    300
SSW.Bfgs_maxstepsize  0.2
SSW.ftol          0.05
SSW.strtol         0.05
```

```
SSW.Temp        300.0000
SSW.LowTemp     300.0000
SSW.HighTemp    300.0000
SSW.TempCycle   500
```

Output example for SSW crystal structure exploration

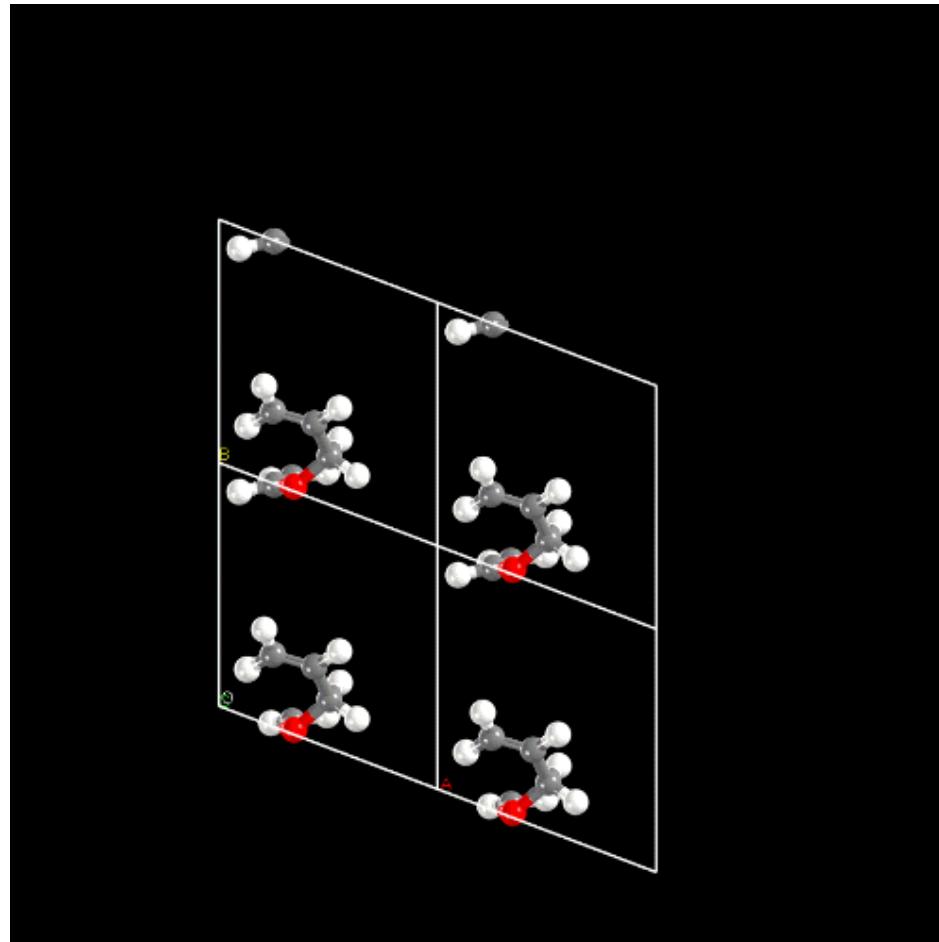
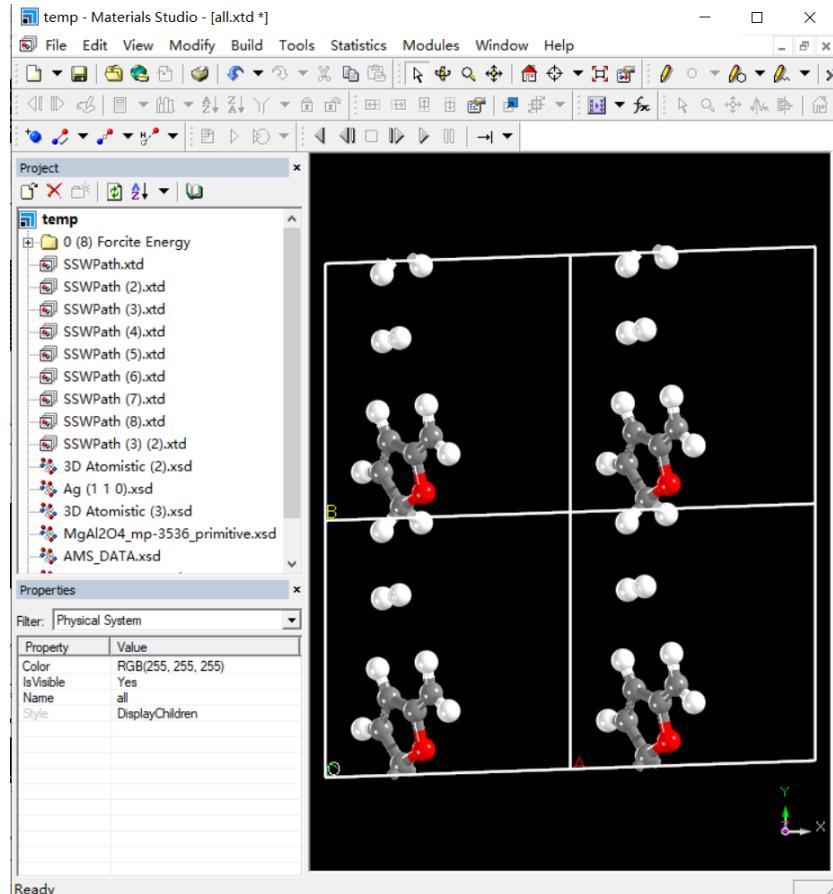
Lasp.out:

| minimum found | 883 | 886 | -79.79004 | | -79.26639 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 0.88 | F |
|-----------------|-----|-----------|-----------|---------|------------|---------|----|---|----|--|--------|------|---|
| Stru symm and Q | 889 | 0 longQ T | 1.38425 | 1.11119 | 0.92409 | 1 | P1 | | | | 5.7000 | 2.58 | F |
| Stru symm and Q | 889 | 0 longQ T | 1.58720 | 1.14710 | 1.06882 | 1 | P1 | | | | 5.7000 | 0.09 | |
| minimum found | 889 | 886 | -79.79004 | | -79.30914 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 3.13 | F |
| Stru symm and Q | 890 | 0 longQ T | 1.38425 | 1.11119 | 0.92409 | 1 | P1 | | | | 5.7000 | 0.10 | |
| Stru symm and Q | 890 | 0 longQ T | 1.37868 | 1.03249 | 0.92806 | 1 | P1 | | | | 3.4701 | 0.29 | T |
| minimum found | 890 | 886 | -79.79004 | | -79.65455 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 0.11 | |
| Stru symm and Q | 891 | 0 longQ T | 1.38425 | 1.11119 | 0.92409 | 1 | P1 | | | | 5.7000 | 0.82 | T |
| Stru symm and Q | 891 | 0 longQ T | 1.39853 | 1.09050 | 0.91209 | 1 | P1 | | | | 5.7000 | 0.09 | |
| minimum found | 891 | 886 | -79.79004 | | -79.80976 | 0.1E+03 | K | 1 | P1 | | 3.4701 | 0.68 | F |
| Stru symm and Q | 892 | 0 longQ T | 1.39853 | 1.09050 | 0.91209 | 1 | P1 | | | | 5.7000 | 2.18 | T |
| Stru symm and Q | 892 | 0 longQ T | 1.36720 | 1.11851 | 0.92542 | 1 | P1 | | | | 5.7000 | 0.08 | |
| minimum found | 892 | 891 | -79.80978 | | -79.79447 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 0.12 | F |
| Stru symm and Q | 893 | 0 longQ T | 1.36720 | 1.11851 | 0.92542 | 1 | P1 | | | | 5.7000 | 0.10 | |
| Stru symm and Q | 893 | 0 longQ T | 1.39390 | 0.97805 | 0.94178 | 1 | P1 | | | | 3.4701 | 0.12 | T |
| minimum found | 893 | 892 | -79.79449 | | -79.59574 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 0.09 | |
| Stru symm and Q | 894 | 0 longQ T | 1.36720 | 1.11851 | 0.92542 | 1 | P1 | | | | 5.7000 | 0.79 | T |
| Stru symm and Q | 894 | 0 longQ T | 1.37821 | 1.06183 | 0.95858 | 1 | P1 | | | | 5.7000 | 0.12 | F |
| minimum found | 894 | 892 | -79.79449 | | -79.74715 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 2.18 | T |
| Stru symm and Q | 895 | 0 longQ T | 1.37821 | 1.06183 | 0.95858 | 1 | P1 | | | | 5.7000 | 0.10 | |
| Stru symm and Q | 895 | 0 longQ T | 1.40718 | 1.08424 | 1.00032 | 1 | P1 | | | | 3.4701 | 0.12 | T |
| minimum found | 895 | 894 | -79.74717 | | -79.777055 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 0.12 | F |
| Stru symm and Q | 896 | 0 longQ T | 1.40718 | 1.08424 | 1.00032 | 1 | P1 | | | | 5.7000 | 0.09 | |
| Stru symm and Q | 896 | 0 longQ T | 1.39062 | 1.10440 | 0.96950 | 1 | P1 | | | | 3.4701 | 0.12 | T |
| minimum found | 896 | 895 | -79.77058 | | -79.78947 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 0.09 | |
| Stru symm and Q | 897 | 0 longQ T | 1.39062 | 1.10440 | 0.96950 | 1 | P1 | | | | 5.7000 | 1.57 | F |
| Stru symm and Q | 897 | 0 longQ T | 1.37974 | 1.08200 | 0.98986 | 1 | P1 | | | | 5.7000 | 0.09 | |
| minimum found | 897 | 896 | -79.78949 | | -79.75701 | 0.1E+03 | K | 1 | P1 | | 5.7000 | 0.12 | T |

Order Parameter Q_2 , Q_4 and Q_6

Distance in SSW
Maximum force eV/Ang₃₁
for new minimum

Display the animation in all.arc file using Material Studio



Example3: NNtrain

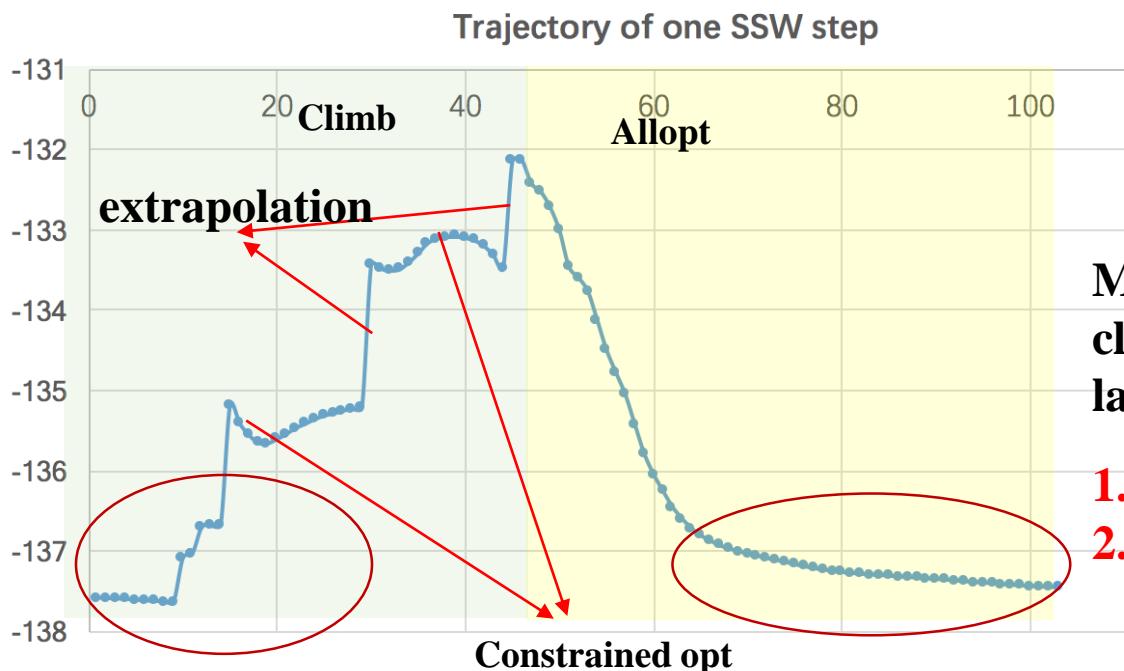
3.1 Prepare data

Notice: Set “SSW.printevery T” when doing DFT computation



| | |
|------------|--------------|
| allstr.arc | xyz/energy |
| allfor.arc | Force/stress |

\$grep Energy allstr.arc
To draw a pathway energy profile



| |
|-------------------|
| SSW.printevery T |
| SSW.printselect 1 |
| SSW.printdelay -1 |

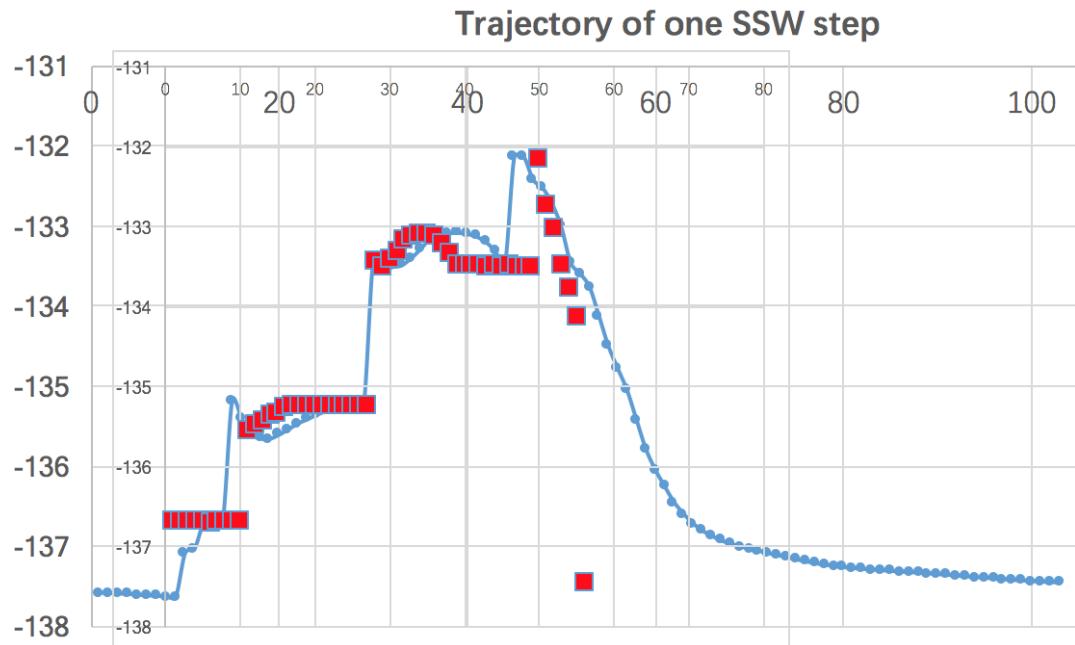
Most point in SSW trajectories are close to the minima (red circle) and lacking of points corresponding to

1. High energy region
2. Extrapolation after adding bias potential.

Comprehensive sampling on the PES is essential to guarantee the robustness and the predictive ability of reactions.

SSW.printselect 6 is recommended

```
SSW.printevery T
SSW.printselect 6
SSW.printdelay -1
```



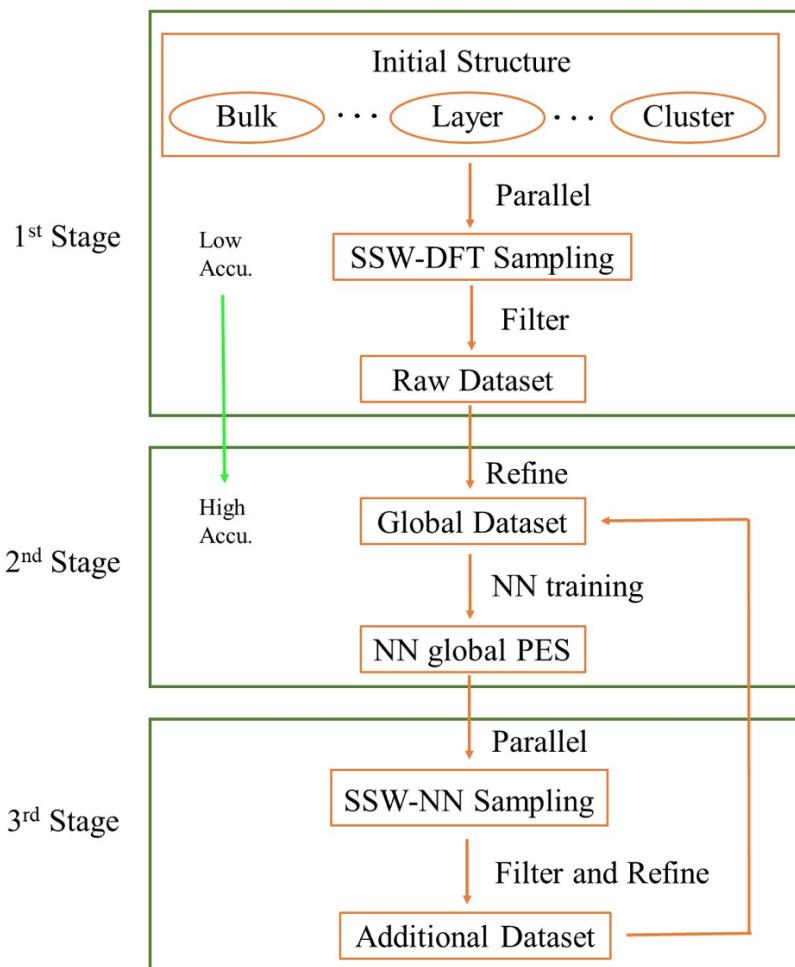
SSW.printselect can greatly affect the number of selected points.

As shown in the left figure, compare to the blue points, the red points

1. Remove most of the points that close to the minimum
2. Include points close to the extrapolation point.

Notice2: Increase the diverseness is always helpful. For example, starting from different random configuration.

Flowchart of NN training



The 1st stage is SSW sampling using DFT computation, which is the main cost.

- **High precision is not necessary for SSW sampling**
The cutoff and K-mesh should be as low as possible but can still generate reasonable configurations
- **The Num. of atoms should be around 16~32**
Large number samplings is indispensable to generate amorphose candidates.
- **Run parallel jobs with different initial configurations**
Some of the jobs should be similar to the target system.
- **More elements requires more training set**
Under the current architechture, we recommend no more than three elements.³⁵

Notice3: Try to reduce the size of the training set (high representative)

Keep only 10%-1% data from the SSW trajectories.

Suppose 20 SSW jobs, with 100 SSWsteps each (SSW.SSWsteps 100)

There will be $20*100*200*0.4= 160k$ data

Randomly pick 5000-10000 points and do (proper) high accurate SPE computation (with SSW.printevery), should be enough to generate a proper NN potential.

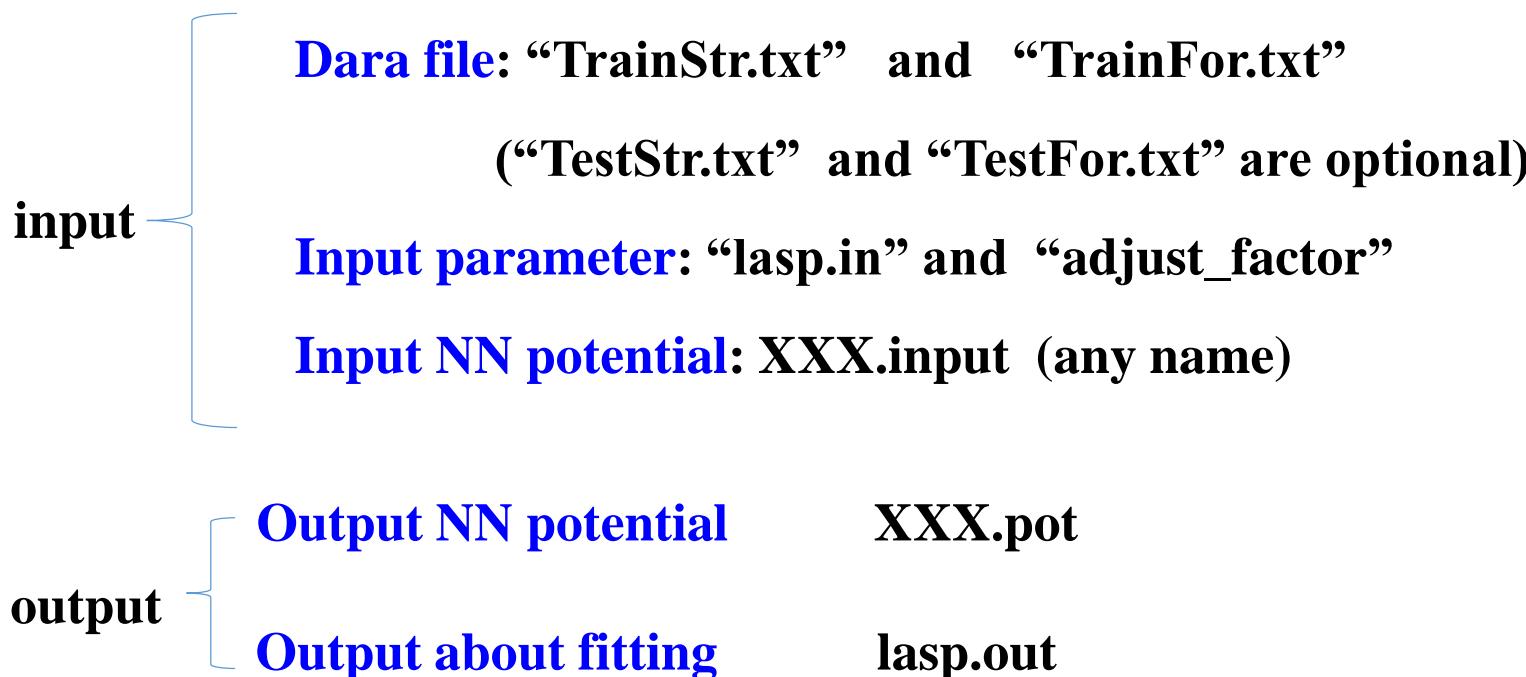
The generated NN potential still needs to be refined by iterative-self learning, i.e. the 2nd and the 3rd stages, which are less time consuming compare to the 1st stage.

Example3: NNtrain

3.2 Training

LASP 2.0

Target: Obtain NN potential by machine learning
(parameter fitting) based on DFT data set.



Data files: TrainStr.txt (structure) and TrainFor.txt (force)

1. Obtain **allstr.arc** and **allfor.arc** from SSW or MD simulation.

You can pastes multiple **allstr.arc** and **allfor.arc** into one **allstr.arc** and one **allfor.arc**.

Just make sure that the pasting sequence of both files are the same.

2. Using python script “**arc2train.py**” to transform “**allstr.arc**” and “**allfor.arc**” into “**TrainStr.txt**” and “**TrainFor.txt**”

(See Example/NNtrain/data-treat)

arc2train.py requires **python 2.7**

- TrainStr.txt

Structure & energy

```
[root@storage3 train-1]# head -100 TrainStr.txt
Start one structure
Energy is -90.311342 eV
total number of element is 21
element in structure:
symbol H O
No. 1 8
number 14 7
weight 1.000 1.000
lat 3.35380000 0.00000000 0.00000000
lat -0.07946898 6.22929312 0.00000000
lat 0.02404341 0.15952898 3.76974942
ele 1 0.14484300 1.44770700 3.03983500 0.00000000
ele 1 1.85554600 6.14001300 3.26371600 0.00000000
ele 1 3.11433900 5.02752500 3.23655200 0.00000000
ele 1 1.78860000 4.76133200 2.51173200 0.00000000
ele 1 0.86479900 1.99063900 0.72217300 0.00000000
```

- TrainFor.txt

Stress & Force

```
[root@storage3 train-1]# head -100 TrainFor.txt
Start one structure
stress -0.42497900 -0.02932100 -0.13619800 -0.47462300 0.04716500 -0.66523600
force 1 -0.05439800 0.24378700 0.02063000
force 1 0.28527200 -0.12513800 -0.25176500
force 1 -0.28888800 -0.00618900 0.16100700
force 1 -0.46466100 0.08383700 -0.17050500
force 1 0.07224600 -0.39832200 -0.01284800
force 1 -0.11602100 -0.79190800 0.00023400
force 1 0.36663400 0.17577500 -0.58945900
force 1 0.55632000 0.45096100 0.42107600
force 1 -0.00741000 -0.36140500 0.11907300
force 1 -0.15219600 0.04598200 0.49314600
force 1 -0.18696600 -0.12957600 0.55091900
```

lasp.in

```
[root@storage3 train-1]# cat lasp.in
#
explore_type train
Ntrain 100
# Input of training
#
Jobname      H2O
Ntest         0
NNepochs     200
readScale    0 # decide read Gmin/Gmax
analyzesym   0
#
# Specify network information
#
%block netinfo
H  H2O.input
O  H2O.input
%endblock netinfo
```

Keywords:

explore_type train

Ntrain 100

Training set includes
100 structures

NNepochs 200 200 epochs

Notice: In real system, the training set generally contains more than 10000 structures and requires at least 10000 epochs.

%block netinfo
the input file of each element

“adjust_factor” file:

```
[root@storage3 train-1]# cat adjust_factor
calfact 200 1000.0 10 0
outfact 1.0 0.0 0.0
train_RMS 1
test_RMS 0
Time_step 1
BFGS_loop 1
```

- In general, you don't have to modify this file.
- The first line is the most important.
- “200 1000 10 0” determining the weight of the energy the force and the stress in the cost function

“XXX.input” file:

```
[root@storage3 train-1]# head -100 H2O.input  
  
shiftElement 0 2 1 8  
  
Element 0 —————→  
Neural Network Architecture  
# Note : typically, fully layer is putting in  
%block 0_netinfo  
# fully layer -> full, size, active_type, fl  
# input 106 2 float newrun  
full 50 2 float newrun  
full 50 2 float newrun  
full 1 0 float newrun  
%endblock 0_netinfo
```

The interpretation of 0th element.

The 0th element contains 2 elements, H(1) and O(8)

NN architechture 106-50-50-1
50-50 are hidden layer

For the first train, write “newrun” here
For restart, changes to XXX.input (e.g. old pot)

continue: XXX.input

PowerType Structure Descriptors(PTSD)

All the fundamental PTSD paramters, namely S1, S2, S3, S4, S5 and S6

```

S1 : sum(rij**n * fc(rij))
      : radical distribution, large n means considering more long-distance neigb
%block 0_S1
#-----#
# cutoff    neigb_atom    n          Gmin          Gmax      #
#-----#
#-----#
 1.20        1         8  0.00000000000000E+00  0.183035038932026E-01 # 1
 1.40        1         2  0.00000000000000E+00  0.103357295876206E+00 # 2
 1.80        1         4  0.00000000000000E+00  0.426574772070577E+00 # 3
 1.80        8         8  0.00000000000000E+00  0.184514583370168E+00 # 4
 1.90        8         2  0.00000000000000E+00  0.955976164742935E-01 # 5
 1.80        8         4  0.00000000000000E+00  0.836717650519582E-01 # 6

```

NN inpute parameters

1: H; 8: O elements

0: the 0th elements refers to multiple elements, which is defined by the “shiftElement” keyword.

XXX.pot: The final potential file that can be used by lasp

The preamble includes elements, date, size of the data set and the accuracy

```
Neural Network Potential For LASP
Reference      : www.lasphub.com
Name          :
Species       : Mn H O
Version        : 20190224063759
Dataset size  : 14239
```

Fitting Accuracy

| | Energy/(meV/atom) | Force/(eV/A) | Stress/(GPa) |
|-----------|-------------------|--------------|--------------|
| RMS | 2.903 | 0.129 | 1.841 |
| MAX | 34.982 | 4.455 | 17.079 |
| scale_RMS | 2.903 | 0.129 | 1.841 |
| scale_MAX | 34.982 | 4.455 | 17.079 |

lasp.out:

```
+-----+
+      train neural network potential
+-----+
NN Started at 22:45:35 23rd February 2019
Time consuming until read train data is 0.0472 s
Time consuming for build network is 0.0476 s
Number of parameters in system is 15902
H : layer 1 is inpu with 1 channel, each 106 nodes, total 106 nodes.
H : layer 2 is full with 1 channel, each 50 nodes, total 50 nodes.
H : layer 3 is full with 1 channel, each 50 nodes, total 50 nodes.
H : layer 4 is full with 1 channel, each 1 nodes, total 1 nodes.
O : layer 1 is inpu with 1 channel, each 106 nodes, total 106 nodes.
O : layer 2 is full with 1 channel, each 50 nodes, total 50 nodes.
O : layer 3 is full with 1 channel, each 50 nodes, total 50 nodes.
O : layer 4 is full with 1 channel, each 1 nodes, total 1 nodes.

YWarning : 0 S1 No. 4 Gmax is too small in dataset.
YWarning : 0 S1 No. 6 Gmax is too small in dataset.

Time consuming for calculating symf is 10.7626 s
-----+
step      0 C =      3414177.565629 BFGS-step = 0.000000
```

Number of parameters in this NN potential

More parameters can provide better accuracy but lead to over fitting more easily.

Indicate the training set is not large enough, some PTSD has an empty value.

**The value of the cost function
(Smaller means better)**

Notice:

1. A good NN potential requires bid data. Each point should not contains too many atoms (<200)

Small data set, such as data from MD can give very high accuracy, but also very poor generality

2. To have a robust potential, the data sets should be within a proper range, e.g. the energy should not be to high (<0) and the force should not be larger than 100 eV/Ang

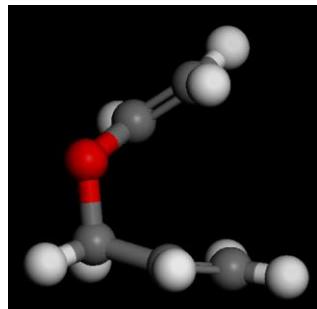
The accuracy of DFT computation should be high and self-consistent. The released NN pot were trained based on VASP results at PBE (or PBE+U) level with 450 eV cutoff and K-point 25 Ang-1.

3. NN training generally requires huge computational resources

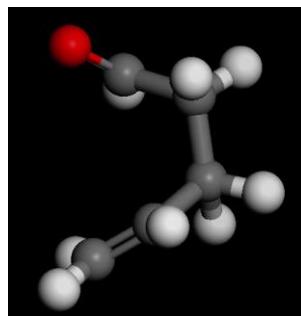
Two weaks with 1000 cores.

Example4: NN-5 -> molecule-TS-DESW

Find the transition states that connecting two known minima

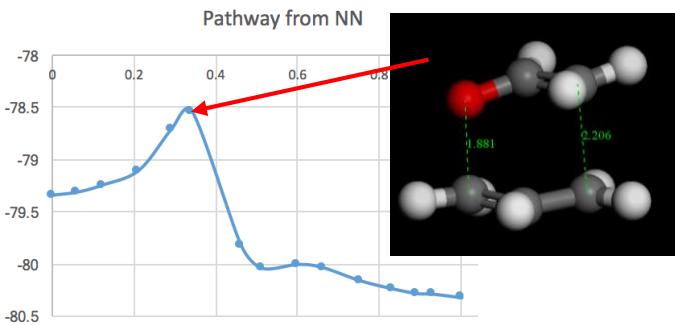


Initial state IS



Final state FS

DESW



Transition state TS

lasp.in

Explore_type ssw

Run_type 2

Fix lattice

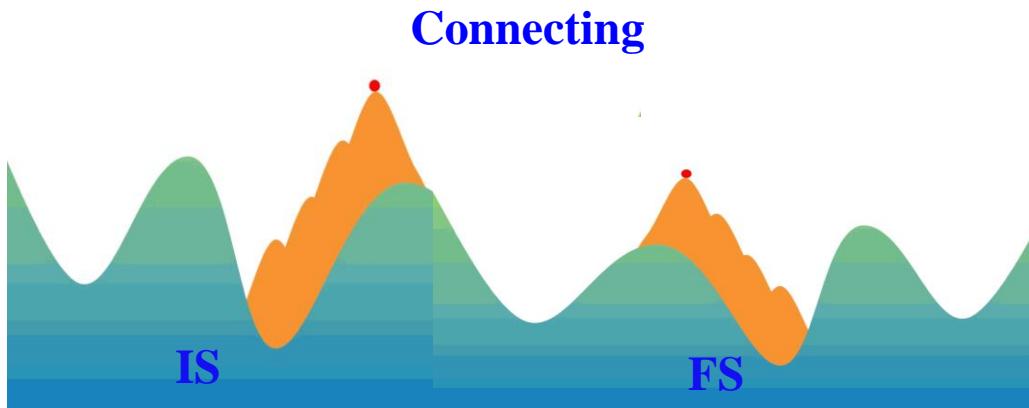
Slabs and nonperiodic molecules / clusters

Run_type 12

Variable lattice

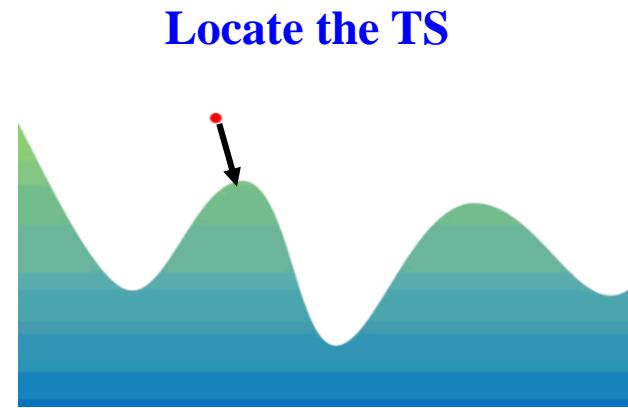
Crystals and periodic materials / polymols

Mechanism of DESW method



Adding bias potential from both IS and FS unit two images reach each other.

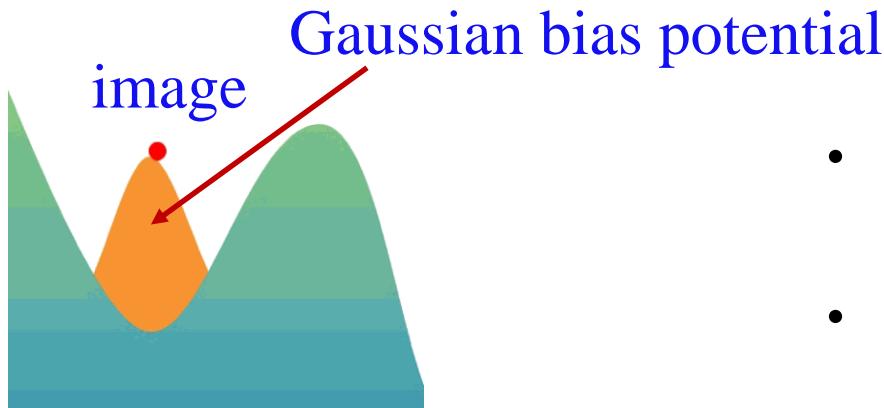
| | | | |
|------------------|----------------|---|------------|
| DESW.task | string | Connecting only | IS and FS. |
| DESW.task | TS | Connecting + TS locating | |
| DESW.task | optpath | Connecting + pathway optimization + TS locating | |



Locate the TS exactly from the highest point of the pathway connecting JS and FS.

Main keywords

| | | | |
|---------------------------|------------|------------|--|
| DESW.ds | 0.2 | Ang | Set DESW stepsize For short pathway, reduce DESW.ds |
| DESW.optpath_cycle | | 4 | Set number of cycles to optimize the pathway |
| CBD.maxcycle | | 20 | Set number of cycles for CBD method to locate TS |
| CBD.TSftol | | 0.1 | Set converge criterion for TS search (maximum atomic force in eV/Ang) |
| CBD.strtol | | 0.1 | Set converge criterion for TS search (maximum stress in Gpa) |
| SSW.output | | T | |
| SSW.printevery | | F | |



- Mode is updated based on the current position of IS / FS images.
- Biased rotation is used to optimize the direction of mode

Keywords

| | |
|-----------------------|--------|
| SSW.DimerdR | 0.0050 |
| SSW.RotMaxStep_preRot | 8 |
| SSW.RotMaxStep | 18 |
| SSW.Rotftol_preRot | 1.0000 |
| SSW.Rotftol | 0.1000 |

SSW.Rotftol **0.01 ~ 0.1**
SSW.Rotftol_preRot **0.5~1.0**

For TS with small negative curvature
(soft TS) , reduce SSW.Rotftol

Extra input file: uncm.arc

```
17:02:30:[zpliu@storage3 NN-5]$ ll
total 112K
lwxrwxrwx 1 zpliu zpliu 18 Jun 4 17:46 uncm.arc -> sourcedir/uncm.arc
lwxrwxrwx 1 zpliu zpliu 17 Jun 4 17:46 lasp.in -> sourcedir/lasp.in
lwxrwxrwx 1 zpliu zpliu 19 Jun 4 17:46 input.arc -> sourcedir/input.arc
lwxrwxrwx 1 zpliu zpliu 18 Jun 4 17:46 CHON.pot -> sourcedir/CHON.pot
-rw----- 1 zpliu zpliu 0 Jun 4 17:46 nohup.out
-rw-rw-r-- 1 zpliu zpliu 0 Jun 9 15:57 z
```

**Notice: the structure in input.arc is not used
(better use IS as input.arc)**

```
17:13:07:[zpliu@storage3 NN-5]$ cat uncm.arc
!BIOSYM archive 2
PBC=ON
```

| | React | 1 | 1 | -1335.4875956 | -1335.4875956 |
|-----|-------------|-------------|-------------|---------------|------------------|
| PBC | 15.0000 | 15.0000 | 15.0000 | 90.0000 | 90.0000 |
| H | 3.144160760 | 4.517488769 | 6.756447451 | CORE | 1 H H 0.0000 1 |
| H | 3.283939812 | 2.651288474 | 6.678731676 | CORE | 2 H H 0.0000 2 |
| H | 5.596489157 | 2.869418544 | 5.804940015 | CORE | 3 H H 0.0000 3 |
| H | 6.663678005 | 5.186591815 | 5.762493313 | CORE | 4 H H 0.0000 4 |
| H | 5.125262368 | 5.906143482 | 6.339914284 | CORE | 5 H H 0.0000 5 |
| H | 2.125315930 | 4.905870300 | 3.098265313 | CORE | 6 H H 0.0000 6 |
| H | 3.575335246 | 3.710735357 | 3.102529396 | CORE | 7 H H 0.0000 7 |
| H | 3.523704793 | 6.570586555 | 4.359380806 | CORE | 8 H H 0.0000 8 |
| C | 5.018270423 | 3.760641652 | 6.078727728 | CORE | 9 C C 0.0000 9 |
| C | 3.915436135 | 5.595886110 | 4.027892549 | CORE | 10 C C 0.0000 10 |
| C | 3.157740732 | 4.682453702 | 3.375488562 | CORE | 11 C C 0.0000 11 |
| C | 3.750282281 | 3.630455282 | 6.543086063 | CORE | 12 C C 0.0000 12 |
| C | 5.566495745 | 5.117793255 | 5.703135701 | CORE | 13 C C 0.0000 13 |
| O | 5.266654611 | 5.386893703 | 4.285289141 | CORE | 14 O O 0.0000 14 |
| end | | | | | |
| end | | | | | |

| | React | 1 | 2 | -1335.4875956 | -1336.5710057 |
|-----|-------------|-------------|-------------|---------------|------------------|
| PBC | 15.0000 | 15.0000 | 15.0000 | 90.0000 | 90.0000 |
| H | 3.144160760 | 4.517488769 | 6.756447451 | CORE | 1 H H 0.0000 1 |
| H | 3.010391122 | 3.101706762 | 5.934107924 | CORE | 2 H H 0.0000 2 |
| H | 5.463107657 | 2.875164776 | 5.735325439 | CORE | 3 H H 0.0000 3 |
| H | 6.873034983 | 4.546433673 | 6.800129624 | CORE | 4 H H 0.0000 4 |
| H | 5.451696901 | 5.762263561 | 6.872310103 | CORE | 5 H H 0.0000 5 |
| H | 2.234067776 | 4.235094530 | 3.947014943 | CORE | 6 H H 0.0000 6 |
| H | 3.860660375 | 3.615131427 | 3.505464637 | CORE | 7 H H 0.0000 7 |
| H | 3.216976521 | 6.522536252 | 4.451976265 | CORE | 8 H H 0.0000 8 |
| C | 5.023626124 | 3.841727937 | 6.014177473 | CORE | 9 C C 0.0000 9 |
| C | 3.759024520 | 5.740272560 | 3.850574344 | CORE | 10 C C 0.0000 10 |
| C | 3.318979905 | 4.317934153 | 4.156960399 | CORE | 11 C C 0.0000 11 |
| C | 3.556447807 | 4.024611797 | 5.673831366 | CORE | 12 C C 0.0000 12 |
| C | 5.820569016 | 4.768449809 | 6.599616539 | CORE | 13 C C 0.0000 13 |
| O | 4.651214584 | 6.062278980 | 3.059462504 | CORE | 14 O O 0.0000 14 |
| end | | | | | |
| end | | | | | |

Contains two structures, i.e. IS and FS
(not necessarily pre-optimized)

Notice: Atoms should have the same sequence in both structures

lasp.out grep “List” to obtain the pathway

odd: from IS
even: from FS

Meeting image don't
compute frequency

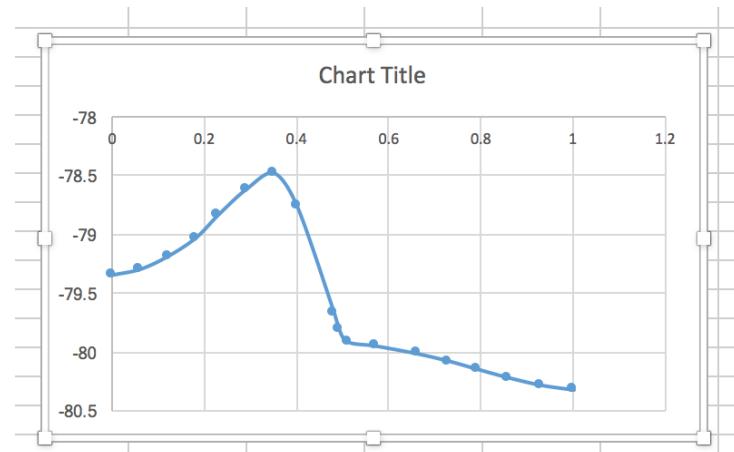
```
16:45:06:[zpliu@storage3 NN-5]$ grep List lasp.out
|   List 1  0.00    2.38      -79.3424   0.008   0.002
|   List 3  0.06    2.74      -79.2927   0.195  -0.274
|   List 5  0.12    3.23      -79.1824   0.304  -0.488
|   List 7  0.18    3.61      -79.0263   0.395  -0.555
|   List 9  0.23    3.20      -78.8259   0.515  -0.814
|   List 11 0.29    3.42      -78.6113   0.723  -0.392
|   List 13 0.35    0.78      -78.4775   0.473  -0.721
|   List 15 0.40    0.65      -78.7624   2.418   0.080
|   List 17 0.48    999.00    -79.6661   1.090  999.000
|   List 18 0.49    999.00    -79.7995   0.558  999.000
|   List 16 0.51    2.17      -79.9127   0.369  -0.393
|   List 14 0.57    0.48      -79.9470   0.597  -0.103
|   List 12 0.66    0.07      -80.0126   0.445  -0.197
|   List 10 0.73    0.36      -80.0771   0.388  -0.244
|   List 8  0.79    0.77      -80.1437   0.544  -0.230
|   List 6  0.86    0.67      -80.2185   0.362  -0.221
|   List 4  0.93    0.65      -80.2811   0.323  -0.174
|   List 2  1.00    0.75      -80.3207   0.008  0.001
16:45:13:[zpliu@storage3 NN-5]$
```

Reaction coordinate
0~1

Frequency (Curvature)

Energy
Maximum atomic force component

Plot Column 3 against 5



\$ grep MaxP lasp.out

```
16:45:13:[zpliu@storage3 NN-5]$ grep MaxP lasp.out
 1 IS/MaxP/FS      -79.34235      -78.47753      -80.32070      0.865
16:56:33:[zpliu@storage3 NN-5]$
```

\$ grep TS lasp.out

```
16:56:33:[zpliu@storage3 NN-5]$ grep TS lasp.out
Start to Construct TS class
begin TSsearch
TS information      -78.555444   -7.400326   0.009516   0.000000   0.000000
 1 IS/TS/FS      -79.34235      -78.55544      -80.32070      0.787
```

Maximum atomic
force component

Curvature of TS

Energy of TS

Barrier

SSWpath.arc

TSstr.arc

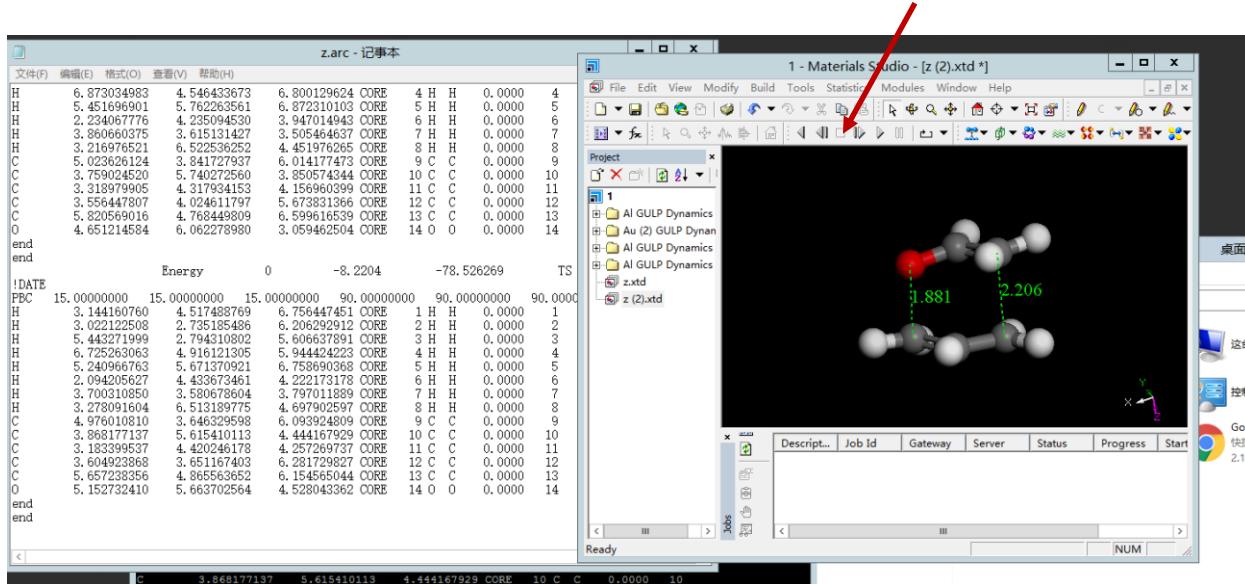
TSmode.arc

DESW pathway

configuration of TS

snapshots correspond to TS vibration

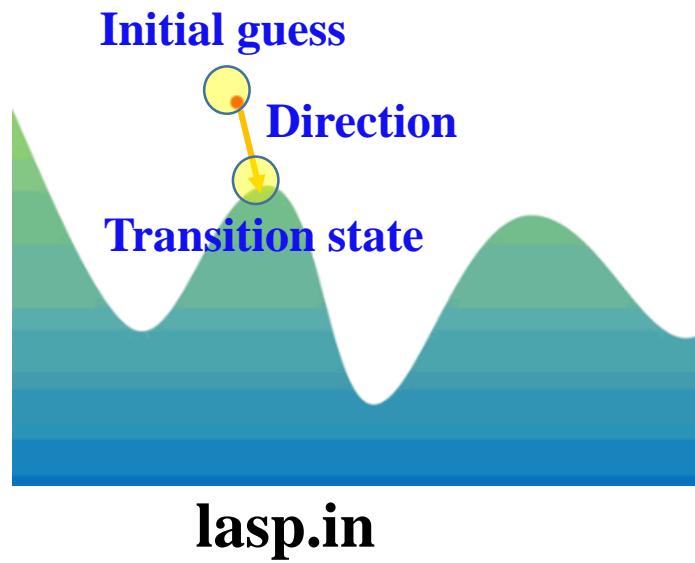
Display the animation in .arc file using Material Studio



Example5: NN-4 -> molecule-TS-CBD

Find the transition state using the pre-guessed initial structure and searching direction

Constrained Broyden
Dimer Method



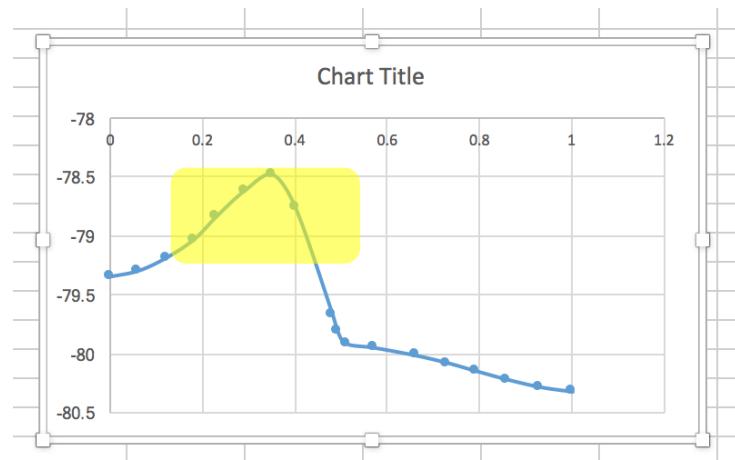
Input files:
lasp.in
input.arc
uncm.arc

Run_type 1 Fix lattice Slabs and nonperiodic molecules / clusters

Run_type 11 Variable lattice Crystals and periodic materials / polymols

Three method to construct the pre-guessed initial structure

1. From the DESW pathway (DESW.task string). The images falling into the yellow region in the right figure are all good candidate.
2. Manually modify the IS / FS structure based on chemical intuition
3. Other jobs has print the allstr.arc file (SSW.printevery T). Find the candidate from it.

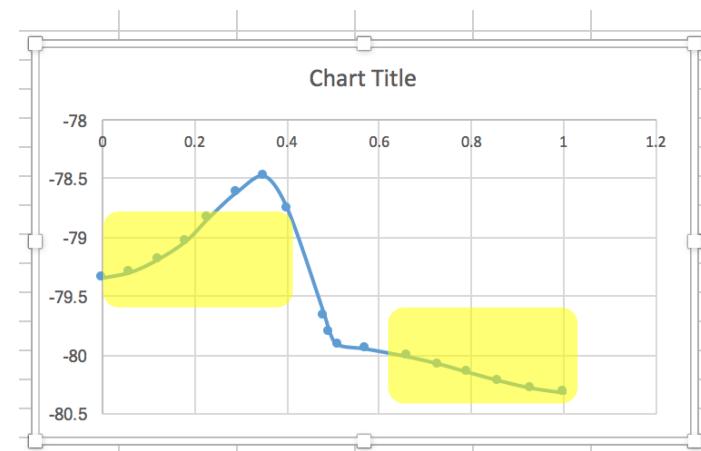


Generally speaking, for the TS that involves breaking/forming bonds. The bond length at the TS is about 0.5 Å longer than at the minimum. For example, the bond length of H₂ is 0.7 Å at the minimum and 1.2 at the TS.

Construct the pre-guessed initial searching direction

The searching direction is generated based on two configurations in uncm.arc file

1. From the DESW pathway (DESW.task string). The images falling into the yellow region in the right figure are all good candidate for direction generation.
2. Modify the IS / FS configuration based on chemical intuition



Keywords:

| | | |
|-----------------------|-------------|---|
| CBD.maxcycle | 20 | Set number of cycles for CBD method to locate TS |
| CBD.maxdist | 0.25 | Set the maximum moving distance in each cycle. |
| CBD.TSftol | 0.1 | Set converge criterion for TS search (maximum atomic force in eV/Ang) |
| CBD.strftol | 0.1 | Set converge criterion for TS search (maximum stress in Gpa) |
| SSW.output | T | |
| SSW.printevery | T | |

Keywords:

- Rotation is used to optimize the direction of mode

```
SSW.DimerdR      0.0050
SSW.RotMaxStep_preRot      8
SSW.RotMaxStep      18
SSW.Rotftol_preRot    1.0000
SSW.Rotftol      0.1000
```

SSW.Rotftol **0.01 ~ 0.1**

SSW.Rotftol_preRot **0.5~1.0**

For TS with small negative curvature (soft TS), reduce SSW.Rotftol

Extra input file: uncm.arc

```
7:47:24:[zpliu@console NN-4]$ ll
total 52K
rwxrwxrwx 1 zpliu zpliu 18 Jun 4 17:39 uncm.arc -> sourcedir/uncm.arc
rwxrwxrwx 1 zpliu zpliu 17 Jun 4 17:39 lasp.in -> sourcedir/lasp.in
rwxrwxrwx 1 zpliu zpliu 19 Jun 4 17:39 input.arc -> sourcedir/input.arc
rw-r--r-- 1 zpliu zpliu 938 Jun 4 17:39 sge.script
rw-rw-r-- 1 zpliu zpliu 0 Jun 4 17:39 output
rwxrwxrwx 1 zpliu zpliu 18 Jun 4 17:41 CHON.pot -> sourcedir/CHON.pot
rwxr-xr-x 2 zpliu zpliu 4.0K Jun 4 17:42 sourcedir/
rw----- 1 zpliu zpliu 0 Jun 4 21:33 nohup.out
rw-r--r-- 1 zpliu zpliu 3.1K Jun 4 21:33 allkeys.log
rw-r--r-- 1 zpliu zpliu 1.4K Jun 4 21:33 TSstr.arc
rw-r--r-- 1 zpliu zpliu 22K Jun 4 21:33 TSmode.arc
rw-r--r-- 1 zpliu zpliu 12K Jun 4 21:33 lasp.out
```

Contains two structures, i.e. IS and FS
(not necessarily pre-optimized)

Notice: Atoms should have the same sequence in both structures

input.arc contains the pre-guessed initial structure

```
17:48:23:[zpliu@console NN-4]$ cat uncm.arc
!BIOSYM archive 2
PBC=ON
      React    1        1      -1335.4875956      -1335.4875956
!DATE
PBC  15.0000  15.0000  15.0000  90.0000  90.0000  90.0000
H    3.144160760  4.517488769  6.756447451 CORE  1 H H  0.0000  1
H    3.283939812  2.651288474  6.678731676 CORE  2 H H  0.0000  2
H    5.596489157  2.869418544  5.804940015 CORE  3 H H  0.0000  3
H    6.663678005  5.186591815  5.762493313 CORE  4 H H  0.0000  4
H    5.125262368  5.906143482  6.339914284 CORE  5 H H  0.0000  5
H    2.125315930  4.905870300  3.098265313 CORE  6 H H  0.0000  6
H    3.575335246  3.710735357  3.102529396 CORE  7 H H  0.0000  7
H    3.523704793  6.570586555  4.359380806 CORE  8 H H  0.0000  8
C    5.018270423  3.760641652  6.078727728 CORE  9 C C  0.0000  9
C    3.915436135  5.595886110  4.027892549 CORE 10 C C  0.0000  10
C   3.157740732  4.682453702  3.375488562 CORE 11 C C  0.0000  11
C   3.750282281  3.630455282  6.543086063 CORE 12 C C  0.0000  12
C   5.566495745  5.117793255  5.703135701 CORE 13 C C  0.0000  13
O   5.266654611  5.386893703  4.285289141 CORE 14 O O  0.0000  14
end
end
      React    1        2      -1335.4875956      -1336.5710057
!DATE
PBC  15.0000  15.0000  15.0000  90.0000  90.0000  90.0000
H    3.144160760  4.517488769  6.756447451 CORE  1 H H  0.0000  1
H    3.010391122  3.101706762  5.934107924 CORE  2 H H  0.0000  2
H    5.463107657  2.875164776  5.735325439 CORE  3 H H  0.0000  3
H    6.873034983  4.546433673  6.800129624 CORE  4 H H  0.0000  4
H    5.451696901  5.762263561  6.872310103 CORE  5 H H  0.0000  5
H    2.234067776  4.235094530  3.947014943 CORE  6 H H  0.0000  6
H    3.860660375  3.615131427  3.505464637 CORE  7 H H  0.0000  7
H    3.216976521  6.522536252  4.451976265 CORE  8 H H  0.0000  8
C    5.023626124  3.841727937  6.014177473 CORE  9 C C  0.0000  9
C    3.759024520  5.740272560  3.850574344 CORE 10 C C  0.0000  10
C    3.318979905  4.317934153  4.156960399 CORE 11 C C  0.0000  11
C    3.556447807  4.024611797  5.673831366 CORE 12 C C  0.0000  12
C    5.820569016  4.768449809  6.599616539 CORE 13 C C  0.0000  13
O   4.651214584  6.062278980  3.059462504 CORE 14 O O  0.0000  14
end
end
```

grep TS lasp.out get the searching trajectory

```
17:49:54:[zpliu@console NN-4]$ grep TS lasp.out
Start to Construct TS class
TS-seach-Energy,force      -78.561114834    0.65
TS-seach-Energy,force      -78.557231575    0.64
TS-seach-Energy,force      -78.556735604    0.64
TS-seach-Energy,force      -78.556562902    0.64
TS-seach-Energy,force      -78.555932690    0.63
TS-seach-Energy,force      -78.555692989    0.63
TS-seach-Energy,force      -78.555700400    0.63
TS-seach-Energy,force      -78.555648234    0.63
TS-seach-Energy,force      -78.561114834    0.65
TS-seach-Energy,force      -78.533794316    0.58
TS-seach-Energy,force      -78.538238739    0.64
TS information      -78.571856   -5.789581   0.095476   0.000000
TS-seach-Energy,force      -78.571856080    0.10
```

Curvature at TS

Energy at TS

Optimize the current mode
(energy/force almost unchanged)

Searching TS
(energy and force tend to reduce)

Maximum atomic
force at TS

\$grep Curv lasp.out (obtain Curvature information)

```
17:51:33:[zpliu@console NN-4]$ grep Curv lasp.out
Curv real -4.50485356102658
CBD_Rotate: Fv,ftol,Curv 1 0.537169 1.000000 -4.504854 -4.504854
Curv real -7.31724313999570
CBD_Rotate: Fv,ftol,Curv 2 0.449970 0.100000 -7.317243 -7.317243
Curv real -8.47138507457987
CBD_Rotate: Fv,ftol,Curv 3 0.385950 0.100000 -8.471385 -8.471385
Curv real -11.8552455379357
CBD_Rotate: Fv,ftol,Curv 4 0.241695 0.100000 -11.855246 -11.855246
Curv real -13.1117479994517
CBD_Rotate: Fv,ftol,Curv 5 0.109170 0.100000 -13.111748 -13.111748
Curv real -13.0737426975446
CBD_Rotate: Fv,ftol,Curv 6 0.115464 0.100000 -13.073743 -13.073743
Curv real -13.3672977991485
CBD_Rotate: Fv,ftol,Curv 7 0.030203 0.100000 -13.367298 -13.367298
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 1.1268 0.8618
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.5539 1.3398
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.5224 1.0654
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.3873 1.0405
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.2929 0.6668
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.2131 0.8398
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.1755 0.7000
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.1686 0.4456
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.1353 1.6320
Translation at Curv < 0 -13.36730, F=Fv-Lamda*FN, FN 1.0000 0.1142 0.8051
Curv real -3.94441884178487
CBD_Rotate: Fv,ftol,Curv 1 0.269116 1.000000 -3.944419 -3.944419
Curv real -4.47941619032063
CBD_Rotate: Fv,ftol,Curv 2 0.383028 0.100000 -4.479416 -4.479416
Curv real -5.07743541571001
CBD_Rotate: Fv,ftol,Curv 3 0.212366 0.100000 -5.077435 -5.077435
Curv real -5.42239192974899
CBD_Rotate: Fv,ftol,Curv 4 0.141156 0.100000 -5.422392 -5.422392
Curv real -5.69947324410195
CBD_Rotate: Fv,ftol,Curv 5 0.138501 0.100000 -5.699473 -5.699473
Curv real -5.78958057278538
CBD_Rotate: Fv,ftol,Curv 6 0.067984 0.100000 -5.789581 -5.789581
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.1178 0.8046
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.0802 3.2411
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.0855 0.5050
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.0687 1.8627
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.0593 0.3710
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.0404 1.0046
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.0418 0.4582
Translation at Curv < 0 -5.78958, F=Fv-Lamda*FN, FN 1.0000 0.0294 1.0531
```

Optimize mode

Cycle-1

Locate the TS
structure

Optimize mode Cycle-2

Locate the TS
structure

SSWpath.arc

TSstr.arc

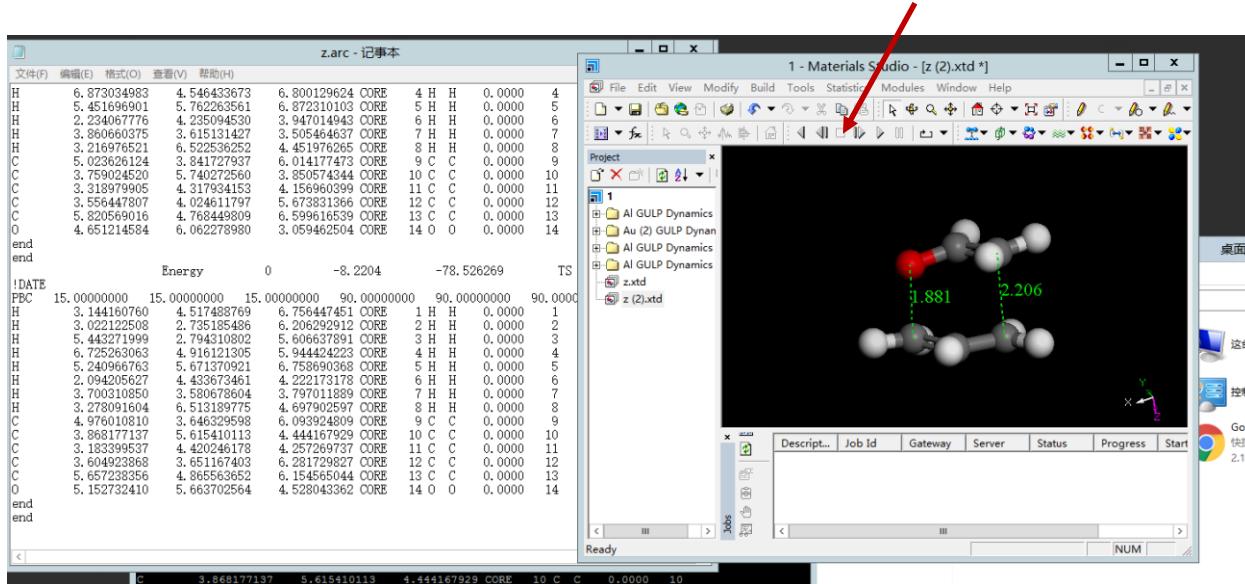
TSmode.arc

DESW pathway

configuration of TS

snapshots correspond to TS vibration

Display the animation in .arc file using Material Studio



Example6: lasp-external/external-gaussian

Lasp.in

```

potential external
External.script    lasp.external.sh

explore_type ssw

Run_type 15      #      15 for SSW with
SSW.Crystal_extra_step T

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

```

- potential external
the energy is computed using external program
- External.script lasp.external.sh
the file name of the script, which should be put in the working directory
(lasp.external.sh by default)

- **to use, the lasp must be run in serial**
\$/xxx/xxx/xx/lasp
- **The external program can be run in parallel by modify the lasp.external.sh script**

When using “external” , the lasp program will give out external.coord, which record the configuration, as shown in below, in the unit of Å

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 |

Element
coordinate x y z

sequence

} lattice
a(x, y, z)
b(x, y, z)
c(x, y, z)

} atom

After running the external program, an “external.ene” file should be generated by the script, which includes energy, atomic force and stress.

As shown below, the unit of energy, force and stress are eV, eV/Å and eV/Å³, respectively.

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
```

- Energy
- Force of each atom, 3×n in total (“n” is the number of atom, read in the sequence of fx(1), fy(1), fz(1), fx(2), fy(2), fz(2)

stress

An example of the Gaussian input

```
[root@node200 external gaussian]# cat gaussian.in
%chk=gaussian.chk
%nprocshared=8
%mem=2GB
#p B3LYP/3-21G
FORCE
FORMCHECK
SCF=(Conver=5)
SCRF=(IEFPCM,solvent=Chloroform,READ)
IOP(7/33=1)
Nosymm

lasp external

      0      1
H 3.0024124390 4.4238420180 6.5231995000
H 3.1916959120 2.6175933250 6.2249342440
H 5.5876013740 2.8563338610 5.9616503120
H 6.7094106270 5.0352306850 6.0237750080
H 5.1113910420 5.8589311850 6.4884285450
H 2.1523749040 4.8225740130 3.6983363170
H 3.5710804780 3.7666639350 3.2065538090
H 3.5567316940 6.4449361500 4.7552692020
C 4.9621932200 3.7151733420 6.1319024860
C 4.0147538040 5.5557413100 4.3004203360
C 3.1949144280 4.5514164330 3.8637003020
C 3.6188972840 3.6296983500 6.1820901100
C 5.6476945460 5.0000885510 6.1134912650
O 5.2739138120 5.4343260050 4.3694133670

OFac=0.8
RMin=0.5
Radii=UAKS

[root@node200 external gaussian]#
```

```
#!/bin/bash

#---Provide any necessary defination 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
#-----
```

- Define charge and multiplicity here
- **Devide the gaussian input file into three parts:**
 1. The first part is “gaussian.inp.pre”, which should be prepared in advance
 2. The second part include charge, multiplicity and coordinates, which is generated during running.
 3. The third part is “gaussian.inp.after”, which include the rest information, such as the solvent model

```
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 -m1 -n 'Cartesian Gradient' Test.FChk|sed 's/:/ /'|awk '{print $1}'`  

c=`grep -m1 -n 'Cartesian Force Constants' Test.FChk|sed 's/:/ /'|awk '{print $1}'`  

let "d=$c-$b-1"  

grep -m1 -A$d 'Cartesian Gradient' Test.FChk|sed 1d|awk -v m=-51.422057 '{printf ("% .15f\n", $m*27.211383)}' >external.frc

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

Notice: Beware of the unit.

How to discuss LaSpHUB Large-scale Atomic Simulation with neural network Potential

LaSpHUB

Large-scale Atomic Simulation with neural network Potential

English | 中文 | Register | Login

HomePage

Tutorial @2018

Software Introduction

User Manual and

Examples

NN Library

LASP download

LASP theoretical
background

LASP applications

LASP tutorial

Lesson 1: input and output

Lesson 2: global optimization

Lesson 3: transition state search (1)

Lesson 4: transition state search (2)

Lesson 5: global NN potential--prepare dataset

Lesson 6: Reaction pathway sampling

Lesson 7: Homogeneous solid-solid phase transition

Lesson 8: interfaces to other programs

Lesson 9: molecular dynamics (1)

Lesson 10: molecular dynamics (2) enhanced sampling

Lesson 11: Constructing NN potential

Lessons On-site course @2018

1-4 lessons

加入 QQ LASPhub group/for other questions and queries, please do not hesitate to contact:

laspmaster@lasphub.com



群名称: LASPhub

群号: 699371738

www.lasphub.com

LaSpHUB

Large-scale Atomic Simulation with neural network Potential

English | 中文 | Register | Login

Thank you!

background

LASP applications

New design of organic reactions and drugs using
machine learning

New design of solid materials using
machine learning

LASP tutorial