

ABACUS + NEB

ABACUS目前没有自带NEB计算，只能依靠ASE（全名Atomic Simulation Environment，一个python库）调用ABACUS来算NEB。

1. 安装 ASE-ABAUCS接口

原版ASE不支持ABACUS，所以我们手动安装一下。（注：建议在conda环境中安装）

```
git clone https://gitlab.com/1041176461/ase-abacus.git
cd ase-abacus
pip install .

# 安装其他必要python库
pip install pymatgen
pip install pymatgen-analysis-diffusion
```

2. 生成NEB初猜轨迹

2.1 生成初猜

准备好初、末态的文件（支持格式：VASP的OUTCAR、ABACUS的running_scf.log/running_relax.log）

假设我们有（这里给的example是极化翻转）：

- 1) 初态输出文件：`up_running_scf.log`
- 2) 末态输入文件：`down_running_scf.log`

```
# 插20个点，输出为neb-init.traj
python 01-neb-make.py -i up_running_scf.log down_running_scf.log -n 20 -o neb-init.traj
```

运行后文件夹下会生成：

- 1) neb-init.traj（ASE格式轨迹，二进制格式）
- 2) POSCAR-xx（对应的POSCAR，方便拖到VESTA里面可视化）

2.2 检查初猜

强烈建议跑之前看一下生成的初猜，确保没有明显不合理的地方。用ASE可以直接可视化`neb-init.traj`：

```
ase gui neb-init.traj
```

2.3 帮助信息

输入：`python3 01-neb-make.py -h` 可查看help message：

```
usage: 01-neb-make.py [-h] -n N [-f FORMAT] -i INPUT INPUT [-m {IDPP,linear}] [-o
0]
                        [-sort_tol SORT_TOL] [--fix FIX] [--mag MAG]

Make input files for NEB calculation

options:
  -h, --help            show this help message and exit
  -n N                  Number of images
  -f FORMAT, --format FORMAT
                        Format of the input files, default is abacus-out
  -i INPUT INPUT, --input INPUT INPUT
                        IS and FS file
  -m {IDPP,linear}, --method {IDPP,linear}
                        Method to generate images
  -o O                  Output file
  -sort_tol SORT_TOL    Sort tolerance for matching the initial and final
structures, default is 1.0
  --fix FIX             [height]:[direction] : fix atom below height (fractional)
in direction
                        (0,1,2 for x,y,z), default None
  --mag MAG             [element1]:[magmom1],[element2]:[magmom2],... : set
initial magmom for atoms
                        of element, default None
```

3. 提交任务

3.1 python脚本

修改`02-neb-run.py`，里面每一项都写了注释，按照需要修改：

```

from ase.optimize import FIRE, BFGS
from ase.io import read, write
from abacus_neb import AbacusNEB

# 1. ABACUS Setup
mpi = 8          # MPI, 等价于: mpirun -np xxx
omp = 4          # OpenMP, 等价于: export OMP_NUM_THREADS=xxx
abacus = "abacus" # abacus可执行文件路径

# 2. ASE 优化器设置
fmax = 0.05      # 力收敛判据, 单位: eV/Å
neb_optimizer = FIRE      # 结构优化器
neb_directory = "NEBrun"  # NEB计算目录
algorithm = "improvedtangent" # NEB算法
climb = True
dyneb = False
parallel = False    # 是否并行。默认是串行计算 (即挨个算完每个NEB image)
k = 0.10            # Spring Constant
init_chain = "neb-init.traj" # 初始NEB初猜

# 3. 赝势基组
pseudo_dir = f"./PP" # 赝势路径
basis_dir = f"./ORB" # 基组路径
pp = {
    'C': 'C_ONCV_PBE-1.0.upf',
    'H': 'H_ONCV_PBE-1.0.upf',
    'Pt': 'Pt_ONCV_PBE-1.0.upf',
}
basis = {
    'C': 'C_gga_7au_100Ry_2s2p1d.orb',
    'H': 'H_gga_6au_100Ry_2s1p.orb',
    'Pt': 'Pt_gga_7au_100Ry_4s2p2d1f.orb',
}

```

3.2 Runscript

提交脚本runscript (注意: 需要激活安装了ASE-ABACUS的conda环境) :

```

#!/bin/bash
#SBATCH --partition=liushi,intel-sc3-32c,intel-sc3
#SBATCH -N 1 # Should be defined manually
#SBATCH --cpus-per-task=4 # Should be defined manually. Equal to
$OMP_NUM_THREADS
#SBATCH --ntasks-per-node=8 # Should be defined manually
#SBATCH --mem=100G
#SBATCH --qos=huge

module purge
module load abacus/3.7.1-zen3
source /home/liushiLab/lidenan/software/miniconda3/bin/activate
export OMP_NUM_THREADS=4

python3 02-neb-run.py

```

4. 任务监控

提交任务后，slurm输出文件长这样；同时会输出neb.traj文件，包含每一步的NEB结果，可以用ASE读取

	Step	Time	Energy	fmax # 最大受力
BFGS:	0	13:08:33	-27773.721348	0.084962
BFGS:	1	13:11:17	-27773.723322	0.065365
BFGS:	2	13:14:02	-27773.727231	0.052534
BFGS:	3	13:16:45	-27773.728308	0.053460
BFGS:	4	13:19:29	-27773.729948	0.049554
BFGS:	5	13:22:13	-27773.731250	0.056479
BFGS:	6	13:24:56	-27773.732594	0.056046
BFGS:	7	13:27:42	-27773.733596	0.045237
BFGS:	8	13:30:27	-27773.734615	0.044459
BFGS:	9	13:33:12	-27773.735753	0.057208
BFGS:	10	13:35:56	-27773.737030	0.066975
BFGS:	11	13:38:39	-27773.738280	0.067569
BFGS:	12	13:41:24	-27773.739483	0.062210
BFGS:	13	13:44:08	-27773.740690	0.057106
BFGS:	14	13:46:53	-27773.741956	0.056265
BFGS:	15	13:49:38	-27773.743143	0.070447
BFGS:	16	13:52:22	-27773.744168	0.069835
BFGS:	17	13:55:05	-27773.745026	0.047235
BFGS:	18	13:57:49	-27773.745843	0.048347
BFGS:	19	14:00:33	-27773.746733	0.065234
BFGS:	20	14:03:17	-27773.747711	0.071390
BFGS:	21	14:06:01	-27773.748747	0.063422
BFGS:	22	14:08:46	-27773.749800	0.050066
	...			

5. 数据提取

```
python3 03-neb-post.py neb.traj 20 # 20是nimages
```