



# **VASPMATE User Manual**

Version 1.0, July 16, 2022

Edit by

Zhaocheng Pan

Ruifeng Zhang

## **M I C I D    Lab**

Materials Integrated Computation and Intelligent Design

## Catalog

1 Introduction to VASPMATE .....	4
2 Configuration and use of VASPMATE .....	5
3 VASPMATE Function Introduction .....	8
3.1 VASP input file .....	8
3.1.1 POSCAR.....	8
3.1.2 INCAR .....	20
3.1.3 KPOINTS.....	23
3.1.4 POTCAR.....	25
3.1.5 Input file checking .....	27
3.2 Energy band calculation.....	28
3.2.1 PBE generalized function calculation energy band steps .....	29
3.2.2 HSE generalized function calculation energy band step.....	40
3.3 Density of states calculation .....	43
3.4 Bader charge calculation.....	47
3.5 Transition state NEB calculation .....	48
3.6 Calculation of charge with edge .....	51
3.7 Charge density calculation.....	53
3.8 Visual Analysis of Real Space Wave Functions.....	56
3.9 Fermi surface calculation.....	57
3.10 3D energy band .....	60
3.11 Thermodynamic quantity correction .....	63

3.11.1 Thermodynamic correction of gas-phase molecules.....	66
3.11.2 Thermodynamic correction of adsorbed molecules .....	68
3.12 Calculation of enthalpy of formation of compounds .....	70
3.13 High throughput programs: ABAND and AEDOS .....	70
4 Examples.....	72
4.1 B <sub>6</sub> N energy band structure calculation (PBE generalized function).....	72
4.2 B <sub>6</sub> N energy band structure calculation (HSE generalized function).....	74
4.3 B <sub>6</sub> N density of states calculation .....	77
4.4 Calculation of energy bands and density of states of Fe (with spin) .....	79
References.....	84

## 1 Introduction to VASPMATE

VASP[1], known as Vienna Ab-initio Simulation Package, is a software package developed by the Hafner group at the University of Vienna to perform first-principles calculations and ab initio molecular dynamics simulations, and is one of the most popular commercial software for materials simulation and computation. However, VASP does not provide a simple graphical user interface, which is a big problem for beginners. In most cases, VASP users need to rely on third-party programs such as ASE, P4VASP, VASPView, VESTA, etc. for visualization, or use homemade scripts or applets such as VTST, VASPMO, etc. Nevertheless, it is still difficult for new users to find the ideal processing script and use it successfully. In recent years, the vaspkit [2] program has partially solved this problem, and its powerful pre- and post-processing features are very user-friendly and convenient for beginners. However, the vaspkit program is more focused on simple and general user interaction control and does not provide enough support for high throughput computational flow design convenience. In this context, VASPMATE, a VASP partner program for simple and efficient high-throughput process design was created.

VASPMATE is very user-friendly and powerful for VASP pre and post processing, runs under linux, is written in C/C++, is small, portable and easy to operate. In addition, VASPMATE as a high-throughput control sub-module can be easily added to the integrated platform SPaMD [3] developed by the group, resulting in two GUI software ABAND and AEDOS for high-throughput calculations of energy band structure and density of states.

## 2 Configuration and use of VASPMATE

Get the VASPMATE installation package

1. Send registry to [zrfcms@buaa.edu.cn](mailto:zrfcms@buaa.edu.cn) for the Baidu.com link.
2. Directly from github download, the link is as follows:

<https://github.com/zrfcms/VASPMATE>

Please send bugs and suggestions to [zrfcms@buaa.edu.cn](mailto:zrfcms@buaa.edu.cn)

## VASPMATE installation steps

>>>>>>>>>> EXECUTABLE BINARY FILE <<<<<<<<<<<<

You can just use the binaries already compiled in the bin directory.

>>>>>>>>>>>>>>>>>> INSTALL <<<<<<<<<<<<<<<<<<

The following installation environment is required:

1. MacOS or Linux system (Centos or Ubuntu)
2. g++ version  $\geq 4.7.0$
3. Reliance on fftw library <http://www.fftw.org/install/windows.html> (suggest version: 3.3.8.)

# >>>>>>>>>>>>>> INSTALL FFTW <<<<<<<<<<<<<<

if your environment shows "can't find <fftw.h>", please try "module load fftw" or similar commands. (We suggest it)

or you can try to use the `fftw-x.x.x.tar.gz` package in your own computer.

>>>>>>>>>>WITH ROOT AUTHORITY<<<<<<<<<<<

- (1) `$tar -zxvf fftw-x.x.x.tar.gz`
- (2) `cd fftw-x.x.x`





## 3 VASPMATE Function Introduction

This manual will introduce the basic command line usage of VASPMATE, which currently includes the generation of input files required for VASP operation, various operations and analysis of the input structure, calculation of energy band structure and density of states using GGA-PBE and HSE-PBE methods, etc.

### 3.1 VASP input file

#### 3.1.1 POSCAR

!!!

ADDITION: VASPMATE CAN ONLY RECOGNIZED POSCAR FORMAT IN VASP 5.x VERSION RATHER THAN 4.x, THAT MEANS POSCAR NEEDS ELEMENTS LINE.

!!!

POSCAR is a file containing cell information, atomic coordinate information, atomic constraint information and atomic velocity information (MD). The following commands operate directly on POSCAR, note that the parameters in parentheses are the default parameters if no parameters are specified, e.g. VASPMATE --prim means that the INPOS file performs the generate progenitor operation and saves the newly generated progenitor to the PRIMPOS file by default.

*Mode:* **cif**



*Syntax:*

```
VASPMATE --cif2pos file1 file2
```

```
VASPMATE --pos2cif file1 file2
```

*Description:*

This module is used for swapping between cif and POSCAR formats. For cif2pos, file1 is the CIF file name and file2 is the POSCAR file name, for pos2cif the opposite is true.

*Examples:*

```
VASPMATE --cif2pos POSCAR POSCAR.cif
```

```
VASPMATE --cif2pos POSCAR.cif POSCAR
```

*Mode: **volume****Syntax:*

```
VASPMATE --vol file(INPOS)
```

*Description:*

This module is used to quickly obtain the crystal structure volume of POSCAR type files.

*Examples:*

```
VASPMATE --vol POSCAR
```

```
VASPMATE --vol
```

*Mode: **atom***

*Syntax:*

VASPMATE -- atom file(INPOS)

*Description:*

This module is used to quickly obtain the crystal structure atom of POSCAR type files.

*Examples:*

VASPMATE -- atom POSCAR

VASPMATE -- atom

*Mode: **primitive cell****Syntax:*

VASPMATE --prim file1(INPOS) file2(PRIMPOS)

*Description:*

A protocell is the smallest repeating unit of a crystal structure, which can completely reflect the chemical-structural characteristics of the distribution of atoms or ions inside the crystal in three dimensions. The default input file file1 is INPOS and the output file file2 is PRIMPOS.

*Examples:*

VASPMATE --prim

VASPMATE --prim POSCAR POSCAR1

*Mode: **convention/unit cell***

*Syntax:*

VASPMATE --unit file1(INPOS) file2(UNITPOS)

*Description:*

In many cases, it is necessary to reduce the original cell to a single cell to better reflect the crystal structure characteristics or symmetry. The default input file file1 is INPOS and the output file file2 is UNITPOS.

*Examples:*

VASPMATE --unit

VASPMATE --unit POSCAR POSCAR1

*Mode: **symm****Syntax:*

VASPMATE --symm file (INPOS)

*Description:*

Symmetry is the most basic characteristic of a crystal structure, and using this module will output information about the basic symmetry of the crystal, such as space group, point group, crystal system, etc. The output information will be saved in the SYMMCAR file by default.

```
*****Find Symmetry*****  
International Tables: 164  
International: P-3m1  
Hall symbol: -P 3 2"  
Holohedry: Trigonal  
Crystal System: -3m  
Number: 164  
Long Name: P -3 2/m 1  
Schoenflies Names: D3d  
*****
```

*Examples:*

VASPMATE--symm

VASPMATE--symm POSCAR

*Mode:* **supercell**

*Syntax:*

VASPMATE --super file1(INPOS) file2(SUPERPOS) sup\_1 sup\_2 sup\_3

*Description:*

This mode will construct a duplicate supercell of the current cell. The lattice vectors of the supercell are integer multiples of the lattice vectors in the original cell. sup\_1, sup\_2 and sup\_3 are multiples of the expanded cell of the lattice vectors. Note that sup\_1, sup\_2, and sup\_3 must be integers and Greater than or equal to 1. The default input file file1 is INPOS and the output file file2 is SUPERPOS.

*Examples:*

VASPMATE -super 4 4 1

VASPMATE --super POSCAR POSCAR1 4 4 1

*Mode:* **affine [4]**

*Syntax:*

VASPMATE --affine -simshear(-purshear,-tension) mode(xx yy zz xy xz yz yx zx zy)

init\_strain step\_length step\_num

*Description:*

This mode applies affine pure or simple deformation to the structure of 3D materials, a structure file with POSCAR format named AFFPOS0 is needed. And the AFFPOS\_\* file is generated, where \* indicates the current strain value.

-simshear, -purshear, -tension is the deformation mode, which is divided into simple shear, pure shear, and uniaxial tension, mode is the deformation direction, init\_strain is the initial strain value, step\_length is the strain increase step, and step\_num is the strain number of steps.

*Examples.*

```
VASPMATE --affine -simshear xy 0 0.1 10
```

```
VASPMATE --affine -purshear xy 0 0.1 10
```

```
VASPMATE --affine -tension xx 0 0.1 10
```

*Mode: alias [4]*

*Syntax:*

```
VASPMATE --alias -tension mode(xx, yy, zz) istart iend ispac [value]
```

```
VASPMATE --alias -shear mode(xy, xz, yz) istart1 iend1 ispac1 istart2 iend2 ispac2  
[zvalue]
```

*Description:*

This mode applies alias deformation to the structure of 3D materials. A structure file named ALIPOS0 with POSCAR format is needed, and then the ALIPOS\_\*(tensi) or ALIPOS\_\*(shear) file will be created after applied alias deformation, where \* indicates the current strain value.

For tension mode, mode is the deformation direction, istart is the starting strain, iend is the ending strain, ispac is the strain step for each step, and the total strain value for each step is  $\text{istart} + \text{ispac} \times k$ ,  $k=0,1,2, 3\ldots$  until  $\text{istart} + \text{ispac} \times k > \text{iend}$ . Similarly, for the shear mode, mode is the deformation plane, istart1 and istart2 are the starting strains in both directions, iend1 and iend2 are the ending strains in both directions, and ispac1 and ispac2 are the starting strains in both directions. ispac1 and ispac2 are the step lengths for each step in both directions. The total strain values for each step are  $\text{istart1} + i \times \text{ispac1}$  and  $\text{istart2} + j \times \text{ispac2}$ ,  $i=0,1,2,\ldots, j=0,1,2,\ldots$  until  $\text{istart1} + i \times \text{ispac1} > \text{iend1}$ ,  $\text{istart2} + j \times \text{ispac2} > \text{iend2}$ .

Tip: After deformation operations with alias, it is usually necessary to fix the atomic positions of the output file. For example, for shear, all atoms need to be relaxed perpendicular to the deformation plane and fixed in other directions, so you can use the fix function of VASPMATE to uniformly relax or fix the atomic coordinates.

*Examples:*

```
VASPMATE --alias -tension xx 1 10 1 0
```

```
VASPMATE --alias -shear xy 1 10 1 1 1 10 0.5 0
```

*Mode:* **proj**

*Syntax:*

```
VASPMATE --proj -rot file1(INPOS) file2(PROJPOS) rotx roty rotz
```

```
VASPMATE --proj -ind file1(INPOS) file2(PROJPOS) pvh pvk pvl uvu uvvv uvw
```

```
VASPMATE --proj -mat file1(INPOS) file2(PROJPOS) mat11 mat12 mat13 mat21
```

mat22 mat23 mat31 mat32 mat33

*Description:*

This mode projects a specific direction of a given cell (i.e., the direction of applied strain) as parallel to an axis vector of tensile deformation (e.g., y), thus facilitating the application of uniaxial tensile deformation; or projects the sliding surface normal to the cell as parallel to an axis vector (e.g., x) while projecting the shear direction as parallel to another axis vector (e.g., y), thus facilitating the application of shear deformation for a specific slip system.

-rot is the rotation method, which requires three rotation angles rotx, roty, and rotz to be provided. The unit of angles is degree.

-ind For the crystal index method, it is necessary to provide an inverse easy vector [pvh,pvk,pvl]\* perpendicular to the plane of the positive space (pvh,pvk,pvl) and a lattice vector [uvu uvv uvw], when it must be ensured that the lattice vector [uvu uvv uvw] lies on the plane (pvh,pvk,pvl), i.e., it is ensured that  $pvh \cdot uvu + pvk \cdot uvv + pvl \cdot uvw = 0$ . Note that pvh, pvk, pvl, uvu, uvv and uvw must be integers.

-mat is the generalized projection method, where mat is an orthogonal normalized  $3 \times 3$  matrix which satisfies with the initial atomic basis vector matrix  $R$  and the projection transformed atomic basis vector matrix  $R^{proj}$  satisfy  $R = R^{proj} \cdot mat_{ij}$ . This matrix can be obtained by the visualization program VESTA.

This mode defaults the input file file1 to INPOS and the output file file2 to PROJPOS.

*Examples:*

VASPMATE --proj -rot 30.00 0.00 30.00

```
VASPMATE --proj -ind 1 1 1 1 1 1-2
```

```
VASPMATE --proj -mat 0.5774 0.5774 0.5774 0.4082 0.4082 -0.8165 -0.7071 0.7071  
0.0000
```

*Mode:* **redefine**

*Syntax:*

```
VASPMATE --redef (INPOS) (REDPOS) vect11 vect12 vect13 vect21 vect22 vect23  
vect31 vect32 vect33 a1 [newaxis] a2 [newplane]
```

*Description:*

This mode will redefine the base vector of the current cell and redetermine the crystal orientation according to the selected parallel coordinate axes and the plane in which it is located. This mode involves redefining the base vector of the current cell and determining the crystal orientation anew based on the selected parallel coordinate axes and the corresponding plane. The last four parameters indicate that the new lattice satisfies where the a1 axis is parallel to the *newaxis*, and the a2 axis lies in the *newplane*. *newaxis* = [x,y,z], *newplane* = [xz,xy,yz] (note that all are lower case). For example a x b xy means that the new lattice vector a will be parallel to the x-axis and vector b will lie in the xy-plane. The default input file file1 is INPOS and the output file file2 is REDPOS.

*Examples:*

```
VASPMATE --redef 1 -1 0 1 1 -2 2 2 2 a x b xy
```

```
VASPMATE --redef POSCAR POSCAR1 1 -1 0 1 1 -2 2 2 2 c x a yz
```



*Mode:* **ieee** [5]

*Syntax:*

VASPMATE --ieee file1(INPOS) file2(IEEEPOS)

*Description:*

This mode converts a 2D or 3D material to the IEEE structure format requested in reference [5]. The default input file file1 is INPOS and the output file file2 is IEEEPOS.

*Examples:*

VASPMATE --ieee POSCAR

*Mode:* **fix**

*Syntax:*

VASPMATE --fixc file2(INPOS) file2(FIXPOS) axis m n F F F (fix m-n)

VASPMATE --fixa file2 file2 1 2 3-5... F F F F (fix m-n)

VASPMATE --fixe file2 file2 A B C... F F F F (fix m-n)

VASPMATE --ufix file (INPOS)

*Description:*

This mode is mainly used to fix the atomic coordinates according to the requirements, and is divided into three fixing methods. The first fixc fixes atoms according to their coordinate components. axis is the selected coordinate axis [x,y,z], m and n are the minimum and maximum values in the Cartesian coordinate system, respectively. for atoms whose coordinate components of the axis axis are between [m,n],

a fix in some direction is applied, e.g. F F F. The second fixa fixes atoms according to their atomic number, with the last three being T or F. The third fixe fixes atoms according to element type, with A,B,C being the element names and the last three being T or F. The last ufix is a clear fix, i.e., it releases all atoms for relaxation. fixc and ufix have INPOS as the default input file1 and FIXPOS as the output file2. fixa and fixe require manual specification of the input and output files .

*Examples:*

```
VASPMATE --fixc z 2 10 F F F
```

```
VASPMATE --fixa INPOS FIXPOS 2-5 F F T
```

```
VASPMATE --fixe INPOS FIXPOS B N F F T
```

```
VASPMATE --ufix
```

*Mode: to direct or cartes*

*Syntax:*

```
VASPMATE --cartes file1(INPOS) file2(NEWPOS)
```

```
VASPMATE --direct file1(INPOS) file2(NEWPOS)
```

*Description:*

This mode is mainly used to convert fractional coordinates and Cartesian coordinates of POSCAR. --cartes means convert the current structure to Cartesian coordinates, --direct means convert the current structure to fractional coordinates, this function does not need to specify the coordinate form of the initial structure, the program will judge and read it by itself. The default input file file1 is INPOS and the

output file file2 is NEWPOS.

*Examples:*

VASPMATE--cartes

VASPMATE --direct POSCAR POSCAR1

*Mode:* **sort**

*Syntax:*

VASPMATE --sortc file1(INPOS) file2(SORTPOS) axis

VASPMATE --sorte file1(INPOS) file2(SORTPOS) elem1 elem2

*Description:*

This module is mainly used for sorting atoms in two ways. sortc sorts the atoms according to the selected axis and rearranges the sequence of atoms in the structure file according to the size of the atomic coordinate components on that axis. sorte swaps the order of two different types of atoms in the structure file. The default input file file1 is INPOS and the output file file2 is SORTPOS.

*Examples:*

VASPMATE --sortc z

VASPMATE --sorte B N

*Mode:* **move**

*Syntax:*

VASPMATE --move (-c/-d) file1(INPOS) file2(MOVEPOS) axis min max dx dy dz

*Description:*

This module is used to move atoms in a directional manner. -c/-d indicates the distance moved relative to the Cartesian (carts) and fractional (direct) coordinates. If this parameter is missing, the movement is performed according to the coordinate form of the initial file, the default input file is INPOS and the output file is MOVEPOS. dx, dy, dz are the distances moved in the three directions, and the moved the atoms are selected as those whose coordinate components are between [min, max] on the axis (x, y, z), and note that the value of this coordinate component should be consistent with the -c/-d selection.

*Examples:*

```
VASPMATE --move z 0 1 0.1 0.2 0.3
```

```
VASPMATE --move -d z 0 1 0.1 0.2 0.3
```

```
VASPMATE --move -d POSCAR POSCAR1 z 0 1 0.1 0.2 0.3
```

### 3.1.2 INCAR

*Mode:* **incar input files**

*Syntax:*

```
VASPMATE --i mode1 mode2 ...
```

*Description:*

For ease of use, VASPMATE has embedded default incar files for calculating different tasks, with the mode option being the abbreviated command for the calculated

task. The generated file format is `incar_*`.... Refer to the following table for detailed information.

rlx/lar	Lattice & Atomic Relaxation
sst/stc	Static State Calculation
mds	Molecular Dynamic Simulation
sar	Standard Atomic Relaxation
mpc	Magnetic Property Calculation
soc	Spin-Orbit Coupling Calculation
hse	Hybrid function HSE06
vdw	DFT-D3 Correction
sic	Self-Interaction Correction
ecc	Elastic Constants Calculation
bca	Bader Charge Analysis
elf	Electron Localization Function
fpm	Frozen Phonon Method
dfp	Density functional Perturbation
neb	Nudged Elastic Band method
tsd	Transition State Dimer method
pbs	Projected Band Structure
bbs	Basic Band Structure
bcd	Basic Charge Density
scd	Spin Charge Density

pcd	Partial Charge Density
tlp	Total Local Potential
esp	Electrostatic Potential
wfn	Wave Function Calculation
tdm	Transition Dipole Moment
los	Linear Optical Spectrums

*Examples:*

VASPMATE --i rlx

VASPMATE --i stc sic hse

*Mode:* **incar parameters revision**

*Syntax:*

VASPMATE --i\_app keyword value (--i\_append)

VASPMATE --i\_del keyword (--i\_delete)

VASPMATE --i\_rem keyword (--i\_remove)

VASPMATE --i\_rep keyword (--i\_replace)

*Description:*

This function is mainly used to make dynamic changes to INCAR files. Note that the operation object must be an INCAR file. --i\_append and --i\_replace are to modify the value of the corresponding keyword and append the corresponding value at the end of the file, --i\_delete is to delete the line where the keyword is located, --i\_remove is to comment out the line where the keyword is located, keyword is each parameter

provided by INCAR, value is the value of the variable corresponding to the keyword, which can be single or multiple numbers and strings, separated by spaces. For `--i_app` and `--i_rep`, if you want to append the parameters consecutively, you need to separate them with - separator.

*Examples:*

```
VASPMATE --i_app EINT 2 5 - LORBIT 11
```

```
VASPMATE --i_del NSW ENCUT
```

```
VASPMATE --i_remove NSW ENCUT
```

```
VASPMATE --i_replace EINT 2 5 - LORBIT 11
```

### 3.1.3 KPOINTS

Only the commonly used KPOINTS method is introduced here. For high symmetry point energy band paths needed for energy band calculation, please refer to the subsequent energy band calculation details.

*Mode:* **kpt**

*Syntax:*

```
VASPMATE --ka (INPOS) [kppra] [kscheme]
```

```
VASPMATE --kv (INPOS) [kspac] [kscheme]
```

```
VASPMATE --km (INPOS) k_1 k_2 k_3 [kmesh] [kscheme]
```

*Description:*

This mode is used to create the KPOINTS files required for non-energy band self-consistent calculations. There are three methods of specifying KPOINTS generation in the VASPMATE code: KPPRA, KSPAC, and KMESH.

KPPRA is a method to automatically set the k-point lattice while keeping the density of k points constant. VASPMATE calculates the inverse easy lattice vector ratio by rigorously computing the number of k points on the a-axis of each inverse easy lattice vector proportional to  $|\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)|/|\mathbf{a}_2 \times \mathbf{a}_3|$ , the  $\mathbf{a}_2$  and  $\mathbf{a}_3$ . The KSPAC method is another automatic method to make the lattice as homogeneous as possible. The number of k points on the three inverse lattice vectors is calculated by the following equation.  $|a_i|/KSPAC$ . Finally VASPMATE also includes a general method for determining KPOINTS by manually setting the values of k\_1, k\_2, and k\_3. KMESH is the type of lattice used and is usually classified as Gamma shift ("Gamma") and Monkhorst-Pack ("MP").

For each setting method, VASPMATE provides default parameters as far as possible, even if no scattering accuracy or grid type is set, it can still generate more accurate k-point files according to the default parameters, which improves the flexibility of use, and the generated k-point files are uniformly named NEWKPT (for PBE calculation can bring the k-point files on the high symmetry path, the difference is named NEWKPATH). For KPPRA and KSPAC, the recommended default values are 1000 and 0.5, respectively (for energy band and density of states calculations, it is recommended to improve the accuracy appropriately).

*Examples:*



VASPMATE --ka 8000 G

VASPMATE --kv 0.5 G

VASPMATE --km 7 7 1 G (both -kmesh and -k are available)

### 3.1.4 POTCAR

*Mode: According To File*

*Syntax:*

VASPMATE --pot (default type)

VASPMATE --pot -type

VASPMATE --pot -type postfix1 postfix2...

(-type = -LDA, -GGA, -PBE)

(postfix = s , d, h, sv, pv, GW)

*Description:*

VASPMATE will generate POTCAR by extracting and combining the elements from the pseudopotential library according to their types in POSCAR, provided that the user creates a file named .potpath in the ~/ directory and enters the path to the pseudopotential library in the following format (case sensitive) VASPMATE supports three pseudopotential models, PBE, LDA and GGA, default type is the PBE pseudopotential without postfix, postfix is the postfix option for the pseudopotential, e.g. s, h, sv, pv, etc.

```
PBE_PATH = /vol-th/home/zrfcms1/P0TPBE54
```

*Examples:*

```
VASPMATE --pot -PBE
```

```
VASPMATE --pot -LDA
```

```
VASPMATE --pot -PBE s
```

```
VASPMATE --pot -PBE sv GW
```

*Mode: According To Elements**Syntax:*

```
VASPMATE --pote (default type) elem1 elem2... postfix1 postfix2...
```

```
VASPMATE --pote -type elem1 elem2... postfix1 postfix2...
```

(-type = -LDA, -GGA, -PBE)

(elem = B N C ...)

(postfix = s , d, h, sv, pv, GW)

*Description:*

In addition to the element types in the POSCAR file, VASPMATE can also generate a POTCAR file based on the user-specified elements, also requiring the user to create a file named .potpath in the ~/ directory and enter the path to the pseudopotential library. Default type is PBE pseudopotential.

*Examples:*

```
VASPMATE --pote B N C
```

```
VASPMATE --pote -LDA B N C
```

```
VASPMATE --pote -GGA B N C sv GW
```

### 3.1.5 Input file checking

*Mode:* **check**

*Syntax:*

VASPMATE --check

*Description:*

VASPMATE makes a judgment on whether the current submission task is satisfied for VASP calculation based on the files in the current folder. VASP does not automatically check whether the element type in POSCAR is the same as that in POTCAR, even if the pseudopotential of H element is used in the calculation of graphene, VASP will not give an error but continue the calculation, but the result will definitely be incorrect; VASPMATE will automatically check whether the current pseudopotential element is the same and give a timely error.

VASPMATE checks if the four files needed for VASP to run are present in the current folder: INCAR, POSCAR, KPOINTS and POTCAR, and gives a prompt if one of them is missing.

VASPMATE also checks if the input file required for VASP to run exists in the current folder according to the control parameters in INCAR: when ISTART equals 1 or 2 or 3, it checks if WAVECAR exists, and if not, it reminds "Attention: ISTART = 1, But No WAVECAR is found!". When ICHARG is equal to 11 or 12 or 13, then check if CHGCAR exists, if not, then remind "Attention: ICHARG = 11, But No CHGCAR

is found!

## 3.2 Energy band calculation

The energy band structure calculation requires the preparation of a normalized progenitor cell and the corresponding k-point path (k-path) and the corresponding integrable Brillouin zone. The integrable Brillouin zone is the first Brillouin zone reduced by all symmetries in the lattice point group (crystal point group). The high symmetry points are identified and selected and connected along the edges of the integrable Brillouin zone.

Traditionally, the high symmetry points in the first Brillouin zone of the crystal inversion space are obtained through the SeeK-Path [6] website or Material Studio software, and then the k-points on the high symmetry points path are generated by scripted interpolation to obtain the required KPOINTS, but these operations are often tedious and not suitable for high throughput calculations. SeeK-Path algorithm to analyze the high symmetry points of crystals and give a set of suitable k-paths for energy band calculations. After testing, VASPMATE produces high symmetry point paths in full agreement with SeeK-Path for all 24 variants of the seven crystalline systems. The specific procedure of energy band calculation by VASPMATE using PBE generalization function and HSE06 hybridization generalization function, respectively, is described later by B<sub>6</sub>N example.

### 3.2.1 PBE generalized function calculation energy band steps

1. After constructing the POSCAR for B<sub>6</sub>N, the first step is to do the usual structure optimization calculations. Here is a simple combination of structure optimization commands.

```
cp POSCAR INPOS  
  
VASPMATE --i rlx  
  
cp incar_rlx INCAR  
  
VASPMATE --pot PBE  
  
VASPMATE --ka 4000  
  
cp NEWKPT KPOINTS
```

2. Optional step: do a single point self-consistent calculation to get CHGCAR.

Copy CONTCAR to INPOS. (Note that all VASPMATE operations in this case use the default object INPOS, so you will need to copy INPOS to POSCAR at the end)

*Syntax:*

```
VASPMATE --kpt3d file1(INPOS) file2(PRIMPOS) kppra(20)
```

*Description:*

Enter VASPMATE --kpt3d 20, VASPMATE will generate a standard primitive cell, note that the energy band calculation must use the standard primitive cell, so you need to enter cp PRIMPOS INPOS. 3d in this command means that the calculation is for 3D structures, for 2D structures there are different processing methods, 20 is the default number of scattering points per for each path. In addition to this, you will get the following file.

*NEWKPATH* contains high symmetry point paths, which can be copied directly for use as KPOITNS.

1	K-Path Generated by VASPMATE			
2	20			
3	Line-Mode			
4	Reciprocal			
5	0.000000	0.000000	0.000000	GAMMA
6	0.500000	0.500000	0.500000	T
7				
8	0.500000	0.500000	0.500000	T
9	0.734877	0.265123	0.500000	H_2
10				
11	0.500000	-0.265123	0.265123	H_0
12	0.500000	0.000000	0.000000	L
13				
14	0.500000	0.000000	0.000000	L
15	0.000000	0.000000	0.000000	GAMMA
16				
17	0.000000	0.000000	0.000000	GAMMA
18	0.382562	-0.382562	0.000000	S_0
19				
20	0.617438	0.000000	0.382562	S_2
21	0.500000	0.000000	0.500000	F
22				
23	0.500000	0.000000	0.500000	F
24	0.000000	0.000000	0.000000	GAMMA
25				
26				

*HIGH\_SYMMETRY\_POINT* holds the high symmetry point information and the recommended path.

1	High - symmetry points(in fractional coordinates).				
2	0.000000	0.000000	0.000000	GAMMA	
3	0.500000	0.500000	0.500000	T	
4	0.500000	0.000000	0.000000	L	
5	0.000000	-0.500000	0.000000	L_2	
6	0.000000	0.000000	-0.500000	L_4	
7	0.500000	0.000000	0.500000	F	
8	0.500000	0.500000	0.000000	F_2	
9	0.382562	-0.382562	0.000000	S_0	
10	0.617438	0.000000	0.382562	S_2	
11	0.382562	0.000000	-0.382562	S_4	
12	0.617438	0.382562	0.000000	S_6	
13	0.500000	-0.265123	0.265123	H_0	
14	0.734877	0.265123	0.500000	H_2	
15	0.734877	0.500000	0.265123	H_4	
16	0.500000	0.265123	-0.265123	H_6	
17	0.382562	-0.265123	0.382562	M_0	
18	0.617438	0.265123	0.617438	M_2	
19	0.734877	0.382562	0.382562	M_4	
20	0.617438	0.617438	0.265123	M_6	
21	0.382562	0.382562	-0.265123	M_8	
22	suggest path : GAMMA-T-H_2 H_0-L-GAMMA-S_0 S_2-F-GAMMA				
23					

Copy the CHGCAR from the previous step and prepare the single point self-consistent INCAR (VASPMATE --i stc) and finally submit the task. The following

describes the VASPMATE energy band calculation post-processing analysis method.

All data results have been set to 0 at the Fermi energy level.

It is worth noting that the Fermi energy level should be based on the self-consistent calculation, so if you want to get the accurate Fermi energy level, you can execute the following command after the structural optimization or static self-consistency to extract the Fermi energy level in DOSCAR after the self-consistency for processing the energy band data:

```
VASPMATE --dos -efermi >> FERMI_LEVEL
```

If VASPMATE does not detect the FERMI\_LEVEL file, it will perform the energy band analysis with the Fermi energy level in DOSCAR in the current folder and give the following message.

You should use the self-consistent calculation fermi\_level!

And the results are often inaccurate at this point.

## **Extraction of energy band data**

*Mode:* Basic Band Structure

*Syntax:*

```
VASPMATE --band -b
```

*Description:*

The output files *BAND.dat* and *BAND\_REFORMATTED.dat* hold the energy band information and *BAND.spa* can be viewed directly with SPaMD own analyzer.

1	#K-Path(1/A)	Energy-Level (eV)
2	# NKPTS & NBANDS: 140 48	
3	# Band-Index 1	
4	0.000000	-17.572077
5	0.040558	-17.570388
6	0.081116	-17.565358
7	0.121675	-17.557113
8	0.162233	-17.545714
9	0.202791	-17.531449
10	0.243349	-17.514537
11	0.283907	-17.495348
12	0.324466	-17.474306
13	0.365024	-17.451885
14	0.405582	-17.428610
15	0.446140	-17.405026
16	0.486698	-17.381823
17	0.527256	-17.359717
18	0.567815	-17.339357
19	0.608373	-17.321463
20	0.648931	-17.306706
21	0.689489	-17.295712
22	0.730047	-17.288932
23	0.770606	-17.286634
24	0.770606	-17.286634
25	0.800588	-17.285113
26	0.830570	-17.280551
27	0.860552	-17.272973
28	0.890534	-17.262418
29	0.920517	-17.248923
30	0.950499	-17.232565
31	0.980481	-17.213415
32	1.010463	-17.191566
33	1.040445	-17.167154
34	1.070428	-17.140311
35	1.100410	-17.111204
36	1.130392	-17.079983
37	1.160374	-17.046889
38	1.190356	-17.012127

The *KLABELS* file saves the location of high symmetry points on the energy band diagram.

1	K-Label	Coordinate of high-symmetry k-point in band-structure plots
2	GAMMA	0.000000
3	T	0.770606
4	H_2/H_0	1.340267
5	L	1.958396
6	GAMMA	2.690696
7	S_0/S_2	3.593656
8	F	3.878486
9	GAMMA	4.735345
10		
11		
12	*Tips: Tip: Please label each high symmetry point in NEWKPATH file. Otherwise, they will be identified as 'XX' in KLABELS file	
13		

*Fermi\_Energy* holds information on Fermi energy levels.

*Examples.*

VASPMATE --band -b

*Mode:* Projected Band-Structure for Each Atom

*Syntax:*

VASPMATE --band -a

*Description:*



The resulting output file name is *PBAND\_A\*.dat*, where \* is the atomic ordinal number.

Each file includes information about the projection on one atom, and the weight on each angular momentum,  $s$   $py$   $pz$   $px$   $dxy$   $dyz$   $dz^2$   $dxz$   $dx^2 + y^2$  tot. The first column is the length of the k-path in Å-1. The second column is the energy of the band. The next column is the projection of the  $lm$  orbital onto this band. The last column is the total projection of the atoms on this band.

1	#K-Path	Energy	s	py	pz	px	dxy	dyz	dz2	dxz	dx2	tot
2	# NKPTS & NBANDS: 140 48											
3	# Band-Index 1											
4	0.000000	-17.572077	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
5	0.040556	-17.570388	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
6	0.081116	-17.565358	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
7	0.121675	-17.557113	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
8	0.162233	-17.545714	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
9	0.202791	-17.531449	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
10	0.243349	-17.514537	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
11	0.283907	-17.495348	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
12	0.324466	-17.474306	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
13	0.365024	-17.451885	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
14	0.405582	-17.428610	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
15	0.446140	-17.405026	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
16	0.486698	-17.381823	0.042000	0.005000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
17	0.527256	-17.359717	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
18	0.567815	-17.339357	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
19	0.608373	-17.321463	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
20	0.648931	-17.306706	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
21	0.689489	-17.295712	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
22	0.730047	-17.289932	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
23	0.770606	-17.286634	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
24	0.811164	-17.286634	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
25	0.851722	-17.285113	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
26	0.892280	-17.280551	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
27	0.932838	-17.272973	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
28	0.973396	-17.262418	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
29	1.013954	-17.248923	0.042000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
30	1.054512	-17.232565	0.041000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
31	1.095070	-17.213415	0.041000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
32	1.135628	-17.191566	0.040000	0.006000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
33	1.176186	-17.167154	0.040000	0.007000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.047000
34	1.216744	-17.140311	0.039000	0.007000	0.000000	0.001000	0.000000	0.000000	0.000000	0.000000	0.000000	0.046000
35	1.257302	-17.111204	0.038000	0.007000	0.000000	0.001000	0.000000	0.000000	0.000000	0.000000	0.000000	0.046000
36	1.297860	-17.079983	0.038000	0.007000	0.000000	0.001000	0.000000	0.000000	0.000000	0.000000	0.000000	0.046000
37	1.338418	-17.046889	0.037000	0.008000	0.000000	0.001000	0.000000	0.000000	0.000000	0.000000	0.000000	0.045000
38	1.378976	-17.012889	0.036000	0.009000	0.000000	0.001000	0.000000	0.000000	0.000000	0.000000	0.000000	0.044000

*Examples.*

VASPMATE --band -a

*Mode:* Projected Band-Structure for Each Element

*Syntax:*

VASPMATE --band -e

*Description:*

The output file name generated is *PBAND\_\*.dat*, where \* is the element symbol. Each file includes information on the projection of one element, and the weight on each angular momentum,  $s$   $py$   $pz$   $px$   $dxy$   $dyz$   $dz^2$   $dxz$   $dx^2 + y^2$  tot. The first column is the

length of the k-path in Å<sup>-1</sup>. The second column is the energy of the band. The next column is the projection of the lm orbitals on this band. The last column is the total projection of the element on this energy band.

*Examples.*

VASPMATE --band -e

*Mode:* Projected Band-Structure for Selected Atoms & Elements

*Syntax:*

VASPMATE --band -s <atom-index> <element-index>

*Description:*

The generated output files are called *PBAND\_A\*.dat* and *PBAND\_B\*.dat*, where \* is the selected atomic number. All generated e.g. *PBAND\_A1.dat*, *PBAND\_B2.dat*... files can be used for origin plotting, and KLABELS is used to mark the location of high symmetry points on the plot.

*Examples.*

VASPMATE --band -s 1-4 B N

*Mode:* Projected Band-Structure for Selected Atoms

*Syntax:*

VASPMATE --band -sa <atom-index>

*Description:*

The output file name generated is *PBAND\_A\*.dat*, where \* is the selected atomic

number. Each file includes information about the projection on the selected atom, as well as the weight on each angular momentum,  $s$   $py$   $pz$   $px$   $dxy$   $dyz$   $dz^2$   $dxz$   $dx^2 + y^2$  tot. The first column is the length of the k-path in Å-1. The second column is the energy of the band. The next column is the projection of the lm orbital onto this band. The last column is the total projection of the selected atom on this energy band.

*Examples.*

VASPMATE --band -sa 1

VASPMATE --band -sa 2-5

*Mode:* Projected Band-Structure for Selected Elements

*Syntax:*

VASPMATE --band -se <element-index>

*Description:*

The name of the generated output file is *PBAND\_B\*.dat*, where B is the name of the selected element and \* is the corresponding atomic number in that element. All generated files such as *PBAND\_B1.dat*, *PBAND\_B2.dat...* files can be used for origin drawing, and KLABELS is used to mark the location of high symmetry points on the graph.

*Examples.*

VASPMATE --band -se B

VASPMATE --band -se B N

*Mode:* Projected Band-Structure for Multiple Atoms or Elements

*Syntax:*

VASPMATE --band -m/-ma/-me <atom-index>/<element-index>

*Description:*

The atomic numbers 1-4, B, N can be selected in free format after the command. The output *PBAND\_SUM\_A1-4\_B-N.dat* file includes the selected atoms and projections. This helps to study the laminar bands and to compare the surface bands with the internal bands.

*Examples.*

VASPMATE --band -m 1-4 B N

VASPMATE --band -ma 1-4

VASPMATE --band -me B N

*Mode:* Sum of Projected Band for Selected Orbitals of Selected Atoms/Elements

*Syntax:*

VASPMATE --band -o/-oa/-oe <atom-index>/<element-index> <orbit-index>

*Description:*

This module is used to select atoms or elements and the corresponding orbitals to generate *PBAND\_SUM.dat*. The orbitals are of type s py pz px dxy dyz dz2 dxz dx2+y2 (f1-f7). You can select all orbitals with all. Note that the combination of atoms + orbitals can only be done once, the result is inaccurate if appended consecutively.

*Examples.*

VASPMATE --band -oa 1-3 b n

VASPMATE --band -oa 1-3 s px py

VASPMATE --band -oe B all

*Mode:* Summary of band structure

*Syntax:*

VASPMATE --band -bg

*Description:*

This module is used to extract Band Gap, VBM, CBM, Highest-Occupied band & Lowest-Occupied Band, Location of VBM & CBM and whether it is a direct band gap or not for energy band calculations. Note that the above information is only available for insulator or semiconductor crystals with band gap, i.e., band\_gap > 0. For metallic crystals, VBM and CBM are not meaningful and only Band Character: Metallic information is given.

----- Band Information -----			
Band Gap (eV):	0.556672		
Band Character:	Indirect		
Fermi Energy (eV):	6.052482		
VBM Eigenvalue (eV):	5.759581		
CBM Eigenvalue (eV):	6.316252		
Highest-Occupied Band:	4		
Lowest-Occupied Band:	5		
Location of VBM:	0.000000	0.000000	0.000000
Location of CBM:	0.421053	0.000000	0.421053

*Examples.*

VASPMATE --band -bg

*Mode:* Effective Mass

*Syntax:*

VASPMATE --em -f [file] -b [band\_index]

*Description:*

This module is used to calculate the effective mass of a semiconductor or insulator. Currently, VASPMATE only supports the calculation of the effective mass of non-charged and non-magnetic semiconductors. The calculation of the effective mass is divided into two parts: pre-processing and post-processing. VASPMATE calculates the effective mass of the electron/hole at a specified high symmetry path by re-scattering the points at equal spacing and then fitting the calculated energy eigenvalues of each point to a quadratic curve with the k-path to obtain the effective mass of the electron/hole at that high symmetry point. In this process, VASPMATE takes a fitting approach by fitting all data points from the starting point and choosing the smallest R-squared as the fitting criterion, which ensures that the loss of information is minimized. Specifically, the effective mass calculation relies on a default file EMC.in (or use “-f” parameter to fix file name), which contains the following information: 1. processing mode 2. kmesh 3. Brillouin zone scattering accuracy (same as KSPACING parameter) 4. number of scattering points on the high symmetry path 5. spacing of scattering points on the high symmetry path 6. path direction between high symmetry points

The EMC.in file for calculating the effective mass pre-processing of MoS2 is shown below

```

1 1 # "1" for pre-process (generate KPOINTS), "2" for post-process (calculate m*)
2 G
3 0.21
4 6 # number of points for quadratic function fitting.
5 0.015 # k-cutoff, unit Å-1.
6 2 # number of tasks for effective mass calculation
7 0.333333 0.3333333 0.000 0.000 0.000 0.000 K->Γ # Specified two K-points and direction
8 0.333333 0.3333333 0.000 0.500 0.000 0.000 K->M # Specified two K-points and direction

```

Using VASPMATE --em, VASPMATE generates an EMCKPT file based on the file information, which is copied to a KPOINTS file for VASP calculation. After the

calculation, change the calculation mode in the first line to 2, leave the rest unchanged, and use VASPMATE --em to analyze again to output the Effective\_Mass file with the effective mass information, as shown below. In addition, the lumo and homo band index is searched by VASPMATE automatically, if you want to calculate the effective mass in the special band index, “-b” + index can do it, such as VASPMATE --em -b 4.

The calculated effective masses are  $0.45 m_0$  for electrons and  $0.54 m_0$  for holes, which are very close to the results of Kormányos [7] et al. of 0.44 and 0.54, respectively.

```

1 + -----Summary----- +
2 Fitting Point: 6
3 Band Index:          LUMO = 10      HOMO = 9
4 Effective-Mass (in m0) Electron (R2)  Hole (R2)
5 K-Path Index 1: K->Γ    0.454757 (0.999998) -0.546734 (1.000000)
6 K-Path Index 2: K->M    0.488905 (0.999967) -0.625759 (0.999928)
7 + -----Summary----- +

```

*Examples:*

VASPMATE --em

VASPMATE --em -f EMC.in -b 4

Although the above method is relatively simple, we need to introduce another fitting method to calculate the effective mass when considering the following two cases: 1. For materials that have already been calculated in the energy band, we do not need to perform the scattering points on the corresponding high-symmetry path again, but can directly fit the results of the energy band data. 2. For materials with a simple merge between the top of the valence band and the bottom of the conduction band, the above method cannot guarantee the accuracy. This can be effectively avoided by calculating directly from the energy band data.

*Syntax:*

VASPMATE --band -em <point> <direct> ... (-np <fitting-point>) (-nb <band-index>)

*Description:*

This module is used to calculate the effective mass of a semiconductor or insulator, where <point> is the name of the high symmetry point, such as G,K,M, etc., and <direct> indicates the choice of direction at that high symmetry point, divided into l (left) and r (right), <point> and <direct> can be <point> and <direct> can be used consecutively. <fitting-point> is the number of fitting points, default is 6, you can use the keyword -np to select (hint: try to select fewer points around the extreme points to improve the accuracy of the quadratic curve fitting recommended 3-6). <band-index> is the energy band index corresponding to cbm, which can be selected using -nb, and is automatically found by VASPMATE by default. Note that if you want to use the high symmetry points that are not used in the original calculation, just append the coordinates and the name of the high symmetry points at the end of the KPOINTS file.

*Examples.*

VASPMATE --band -em K l K r

VASPMATE --band -em -np 4 GAMMA l M r -nb 5

### 3.2.2 HSE generalized function calculation energy band step

(1) The same structure optimization of POSCAR as for the PBE generalized function calculation.

(2) Use VASPMATE --kpt3d 20 to obtain PRIMPOS proto-cell and NEKPATH high



symmetry path files.

(3) Choose to modify the high symmetry path (you can use the default path).

(4) The following three ways are selected to generate the hybrid functional band-structure to calculate the k points.

```
VASPMATE --kahse [kppra] [resolution] [kscheme]
```

```
VASPMATE --kvhse [kspac] [resolution] [kscheme]
```

```
VASPMATE --kmhse k_1 k_2 k_3 [resolution] [kscheme]
```

Compared with the common k-point generation path, HSE introduces an additional parameter resolution for the density of scattered points on the high symmetry path, with the default recommended value of 0.05. The generated file NEWKPT stores the k-point information calculated by HSE, which can be directly used for VASP calculation.

Note that the k-point of the HSE calculation energy band includes the SCF part, so the obtained Fermi energy level is accurate and does not require additional preparation of the FERMI\_LEVEL file for the same PBE calculation.

1	9	9	9	85	89	7	15	10	12	14	17	5	16	# Parameters to Generate KPOINTS (Do NOT Edit This Line)
2	174													
3	Reciprocal lattices													
4	0.000000	0.000000	0.000000	0.000000	1									
5	0.111111	0.000000	0.000000	0.000000	6									
6	0.222222	0.000000	0.000000	0.000000	6									
7	0.333333	0.000000	0.000000	0.000000	6									
8	0.444444	0.000000	0.000000	0.000000	6									
9	0.111111	0.111111	0.000000	0.000000	6									
10	0.222222	0.111111	0.000000	0.000000	12									
11	0.333333	0.111111	0.000000	0.000000	12									
12	0.444444	0.111111	0.000000	0.000000	12									
13	-0.444444	0.111111	0.000000	0.000000	12									
14	-0.333333	0.111111	0.000000	0.000000	12									
15	-0.222222	0.111111	0.000000	0.000000	12									
16	-0.111111	0.111111	0.000000	0.000000	6									
17	0.222222	0.222222	0.000000	0.000000	6									
18	0.333333	0.222222	0.000000	0.000000	12									
19	0.444444	0.222222	0.000000	0.000000	12									
20	-0.444444	0.222222	0.000000	0.000000	12									
21	-0.333333	0.222222	0.000000	0.000000	12									
22	-0.222222	0.222222	0.000000	0.000000	6									
23	0.333333	0.333333	0.000000	0.000000	6									
24	0.444444	0.333333	0.000000	0.000000	12									
25	-0.444444	0.333333	0.000000	0.000000	12									
26	-0.333333	0.333333	0.000000	0.000000	6									
27	0.444444	0.444444	0.000000	0.000000	6									
28	-0.444444	0.444444	0.000000	0.000000	6									
29	0.111111	0.111111	0.111111	0.000000	2									
30	0.222222	0.111111	0.111111	0.000000	6									
31	0.333333	0.111111	0.111111	0.000000	6									
32	0.444444	0.111111	0.111111	0.000000	6									
33	-0.444444	0.111111	0.111111	0.000000	6									
34	-0.333333	0.111111	0.111111	0.000000	6									
35	-0.222222	0.111111	0.111111	0.000000	6									
36	-0.111111	0.111111	0.111111	0.000000	6									
37	0.222222	0.222222	0.111111	0.000000	6									
38	0.333333	0.222222	0.111111	0.000000	12									

*Examples.*

VASPMATE --kahse 8000 0.05 g

VASPMATE --kvhse 0.5 0.05 G

VASPMATE --kmhse 7 7 1 0.05 g

(5) You can select the newly generated NEWKPT for PBE self-consistent calculation, and then read the wave function for the next HSE calculation, which can usually reduce the calculation time. It is also possible to skip this step and perform the HSE energy band calculation directly. Note that the INCAR for HSE calculation can be generated using the command VASPMATE --i stc hse.

(6) Do HSE can bring calculations.

For the post-processing part of the HSE calculation, VASPMATE has designed its command format to be similar to that of the PBE calculation, with only minor differences in the label. The specific input format can be referred to the details of the PBE calculation.

*Syntax:*

VASPMATE --band -hb

VASPMATE --band -ha

VASPMATE --band -he

VASPMATE --band -hs <atom-index> <element-index>

VASPMATE --band -hsa <atom-index>

VASPMATE --band -hse <element-index>

VASPMATE --band -hm/-hma/-hme <atom-index>/<element-index>

VASPMATE --band -ho/-hoa/-hoe <atom-index>/<element-index>

VASPMATE --band -hbg

*Examples.*

VASPMATE --band -hb

VASPMATE --band -ha

VASPMATE --band -he

VASPMATE --band -hs 1-4 B N

VASPMATE --band -hsa 1-4

VASPMATE --band -hse B N C

VASPMATE --band -hm/-hma/-hme 1-4 B N

VASPMATE --band -ho/-hoa/-hoe 1-3 s px py

VASPMATE --band -hbg

### 3.3 Density of states calculation

VASPMATE provides a powerful post-processing analysis function for density of states extraction. Please make sure that LORBIT=10 or 11 when using this function. all data results have the Fermi energy level set to 0.

In principle, the density of states can be used as a visualization result of the energy band structure. Many analysis and energy band analysis results can be corresponded to one another, and many terms are also related to energy band analysis. However, because it is more intuitive, it is more widely used in the discussion of results than energy band analysis. In the case of a quasi-continuous distribution of electron energy levels, the

number of electronic states per unit energy interval. That is, the ratio of the number of quantum states  $\Delta Z$  with energy between  $E \sim E + \Delta E$  to the energy difference  $\Delta E$  is the density of states. The energy density of states is closely related to the energy band structure and is an important fundamental function. Many properties of solids, such as specific heat of electrons, absorption and emission of light and X-rays, etc., are related to the energy density of states.

In VASP, by projecting the density of states to orbitals or atoms respectively, the split-wave density of states PDOS (partial DOS) and local density of states LDOS (local DOS) can be obtained for orbital analysis and component analysis, respectively. Besides, when LORBIT=10 in INCAR, the projected orbitals are  $s$ ,  $p$ ,  $d(f)$ ; and when LOBRIT=11, the projected orbitals are further subdivided into  $s$ ,  $px$ ,  $py$ ,  $pz$ ,  $dxy$ ,  $dyz$ ,  $dxz$ ,  $dx^2+y^2$ ,  $dz^2$  ( $f1-f7$ ).

#### Mode: Density of States

<i>Syntax</i>	<i>Description</i>
VASPMATE --dos -t <none>	Get the total DOS
VASPMATE --dos -a <none>	Output projected DOS for each atom to separate file.
VASPMATE --dos -e <none>	Output projected DOS for each element to separate file.
VASPMATE --dos -s <atom-index>	Output the projected DOS for selected atoms/element to separate file.
VASPMATE --dos -sa <atom-index>	Output the projected DOS for selected atoms to separate file.
VASPMATE --dos -se <element-index>	Output the projected DOS for selected elements to separate file.
VASPMATE --dos -m/-me/-ma <select-list>	Output sum of projected DOS for selected multiple atoms/elements
VASPMATE --dos -o/-oe/-oa <atom&orbit-list>	Output sum of projected DOS for selected atoms and orbitals

VASPMATE --dos -bc	Output Band Center(s,p,d,f) or (s,px,py,pz,dxy,dyz,dxz,fn...)
--------------------	---

*Examples:* (The file starting with I is the result of a DOS integration with the structure B<sub>6</sub>N and the atomic number 14)

<i>Input command</i>	<i>Output files</i>
VASPMATE --dos -t	TDOS.dat & ITDOS.dat
VASPMATE --dos -a	PDOS_A1.dat & IPDOS_A1.dat ~ PDOS_A9.dat & IPDOS_A9.dat
VASPMATE --dos -e	PDOS_B.dat & IPDOS_B.dat & PDOS_N.dat & IPDOS_N.dat
VASPMATE --dos -s 1-5	PDOS_A1.dat & IPDOS_A1.dat ~ PDOS_A5.dat & IPDOS_A5.dat
VASPMATE --dos -m 1-5 N	PDOS_SUM_A1-5_N.dat & IPDOS_SUM_A1-5_N.dat SELECT_ATOMS_LIST
VASPMATE --dos -oa 1-4 s px	PDOS_USER.dat
VASPMATE --dos -bc all	Band_Center

*TDOS.dat(PDOS\_A\*.dat,PDOS\_\*.dat,PDOS\_SUM\_A\*\_.dat,PDOS\_SUM\_\*\_.dat*

*are in the same format and can be used for origin drawing)*

	#Energy	TDOS
1		
2	-18.876479	0.000000
3	-18.858479	0.000000
4	-18.840479	0.000000
5	-18.823479	0.000000
6	-18.805479	0.000000
7	-18.787479	0.000000
8	-18.769479	0.000000
9	-18.751479	0.000000
10	-18.733479	0.000000
11	-18.715479	0.000000
12	-18.698479	0.000000
13	-18.680479	0.000000
14	-18.662479	0.000000
15	-18.644479	0.000000
16	-18.626479	0.000000
17	-18.608479	0.000000
18	-18.590479	0.000000
19	-18.573479	0.000000
20	-18.555479	0.000000

*PDOS\_SUM\_A1-5\_N.dat (holds energy and projection orbit information for selected atoms, information about selected atoms can be viewed at SELECT\_ATOMS\_LIST)*

1	#Energy	s	py	px	px	dxz	dyz	dx2	dxz	dx2	tot
2	-18.876479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	-18.858479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
4	-18.840479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
5	-18.823479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
6	-18.805479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
7	-18.787479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
8	-18.769479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
9	-18.751479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
10	-18.733479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
11	-18.715479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
12	-18.698479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
13	-18.680479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
14	-18.662479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
15	-18.644479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
16	-18.626479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
17	-18.608479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
18	-18.590479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
19	-18.573479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
20	-18.555479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
21	-18.537479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
22	-18.519479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
23	-18.501479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
24	-18.483479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25	-18.465479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
26	-18.448479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
27	-18.430479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
28	-18.412479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
29	-18.394479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
30	-18.376479	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

*PDOS\_USER.dat (the first line corresponds to the command VASPMATE --dos -o 1-2*

*s N px py)*

1	#Energy 1-2_s	N_px*py
2	-18.876479	0.000000
3	-18.858479	0.000000
4	-18.840479	0.000000
5	-18.823479	0.000000
6	-18.805479	0.000000
7	-18.787479	0.000000
8	-18.769479	0.000000
9	-18.751479	0.000000
10	-18.733479	0.000000
11	-18.715479	0.000000
12	-18.698479	0.000000
13	-18.680479	0.000000
14	-18.662479	0.000000
15	-18.644479	0.000000
16	-18.626479	0.000000
17	-18.608479	0.000000
18	-18.590479	0.000000
19	-18.573479	0.000000
20	-18.555479	0.000000
21	-18.537479	0.000000
22	-18.519479	0.000000
23	-18.501479	0.000000
24	-18.483479	0.000000
25	-18.465479	0.000000
26	-18.448479	0.000000
27	-18.430479	0.000000
28	-18.412479	0.000000
29	-18.394479	0.000000
30	-18.376479	0.000000

*SELECT\_ATOMS\_LIST (atom selection information, all atoms are selected in the figure)*

	ATOMS_ID	ATOM_LABEL	X_POSITION	Y_POSITION	Z_POSITION	SELECTED?
1						
2	1	B1	0.796745	0.328767	0.796745	T
3	2	B2	0.203255	0.671233	0.203255	T
4	3	B3	0.796745	0.796745	0.328767	T
5	4	B4	0.203255	0.203255	0.671233	T
6	5	B5	0.328767	0.796745	0.796745	T
7	6	B6	0.671233	0.203255	0.203255	T
8	7	B7	0.999036	0.664694	0.999036	T
9	8	B8	0.000964	0.335306	0.000964	T
10	9	B9	0.999036	0.999036	0.664694	T
11	10	B10	0.000964	0.000964	0.335306	T
12	11	B11	0.664694	0.999036	0.999036	T
13	12	B12	0.335306	0.000964	0.000964	T
14	13	N1	0.626852	0.626852	0.626852	T
15	14	N2	0.373148	0.373148	0.373148	T
16						

*PDOS\_USER.dat(VASPMATE --dos -oa 1-4 s px 5-9 all)*

	#Energy	1-4_s&px	5-9_all
1			
2	-18.887084	0.000000	0.000000
3	-18.869084	0.000000	0.000000
4	-18.851084	0.000000	0.000000
5	-18.834084	0.000000	0.000000
6	-18.816084	0.000000	0.000000
7	-18.798084	0.000000	0.000000
8	-18.780084	0.000000	0.000000
9	-18.762084	0.000000	0.000000
10	-18.744084	0.000000	0.000000
11	-18.726084	0.000000	0.000000
12	-18.709084	0.000000	0.000000
13	-18.691084	0.000000	0.000000
14	-18.673084	0.000000	0.000000
15	-18.655084	0.000000	0.000000
16	-18.637084	0.000000	0.000000
17	-18.619084	0.000000	0.000000
18	-18.602084	0.000000	0.000000
19	-18.584084	0.000000	0.000000
20	-18.566084	0.000000	0.000000
21	-18.548084	0.000000	0.000000
22	-18.530084	0.000000	0.000000
23	-18.512084	0.000000	0.000000
24	-18.494084	0.000000	0.000000
25	-18.476084	0.000000	0.000000

*Band\_Center (save s,p,d band center data)*

1	#	Atom_ID	s-band-center	p-band-center	d-band-center	(in units of eV)								
2	SPIN-UP	all	49.639715	51.622606	25.861222									
3	SPIN-DW	all	41.392813	53.453747	27.228907									
4														
5	#	Atom_ID	s	py	pz	px	dxy	dyz	dz2	dxz	dx2	(in units of eV)		
6	SPIN-UP	all	49.639715	51.695426	51.599076	51.572309	25.025426	24.530076	26.772782	26.027049	26.819471			
7	SPIN-DW	all	49.639715	53.600855	53.439957	53.316472	26.138259	25.684847	28.493581	27.208352	28.452890			
8														
9														
10	Remark:													
11	Energy window for integration is from -12.597178 to 76.665822.													
12	Band Center is respect to Fermi level, i.e., E_F = 0 eV.													
13														

### 3.4 Bader charge calculation

*Mode: Combine Charge Density*

*Syntax:*

VASPMATE --bader -comb AECCAR0 AECCAR2 (factor1) (factor2)

*Description:*

The output charge density  $\text{AECCAR0} \times \text{factor1} + \text{AECCAR2} \times \text{factor2}$ , the output file is CHGCAR\_SUM.

*Examples:*

VASPMATE --bader -comb AECCAR0 AECCAR2

VASPMATE --bader -comb AECCAR0 AECCAR2 1.1 0.9

### 3.5 Transition state NEB calculation

VASPMATE provides the following functions for the NEB transition state energy barrier calculation: 1) Similarity comparison between the beginning and end structures 2) Calculation of the number of interpolation points based on the differences between the beginning and end structures, and automatic generation of intermediate state files 3) After each intermediate state structure is calculated, OUTCAR is analyzed, data is extracted, and the corresponding files are output.

*Mode: Structure Similarity*

*Syntax:*

VASPMATE --neb -sim (ini) (fin)

*Description:*

This function is mainly used to compare the structural variability of the beginning and end structures and to give suggested interpolation point values. The default two



input files are ini and fin.

*Examples:*

```
VASPMATE --neb -sim
```

```
VASPMATE --neb -sim POSCAR POSCAR1
```

*Mode:* **Insert Points**

*Syntax:*

```
VASPMATE --neb -ins
```

*Description:*

This function is used to create intermediate state files for NEB calculation based on the proposed interpolation point values obtained from the calculation. OUTCAR, fin, fin.OUTCAR, respectively, the initial state structure, the initial state calculated OUTCAR, the final state structure, the final state calculated OUTCAR. then VASPMATE will create folders from the initial state to the final state according to the interpolation points, named 00, 01, 02 ...the folders for the initial and final states hold their corresponding POSCAR and OUTCAR, and the folder for the intermediate state holds its corresponding POSCAR.

*Examples:*

```
VASPMATE --neb -ins
```

*Mode:* **Output Data**

*Syntax:*

## VASPMATE --neb -out

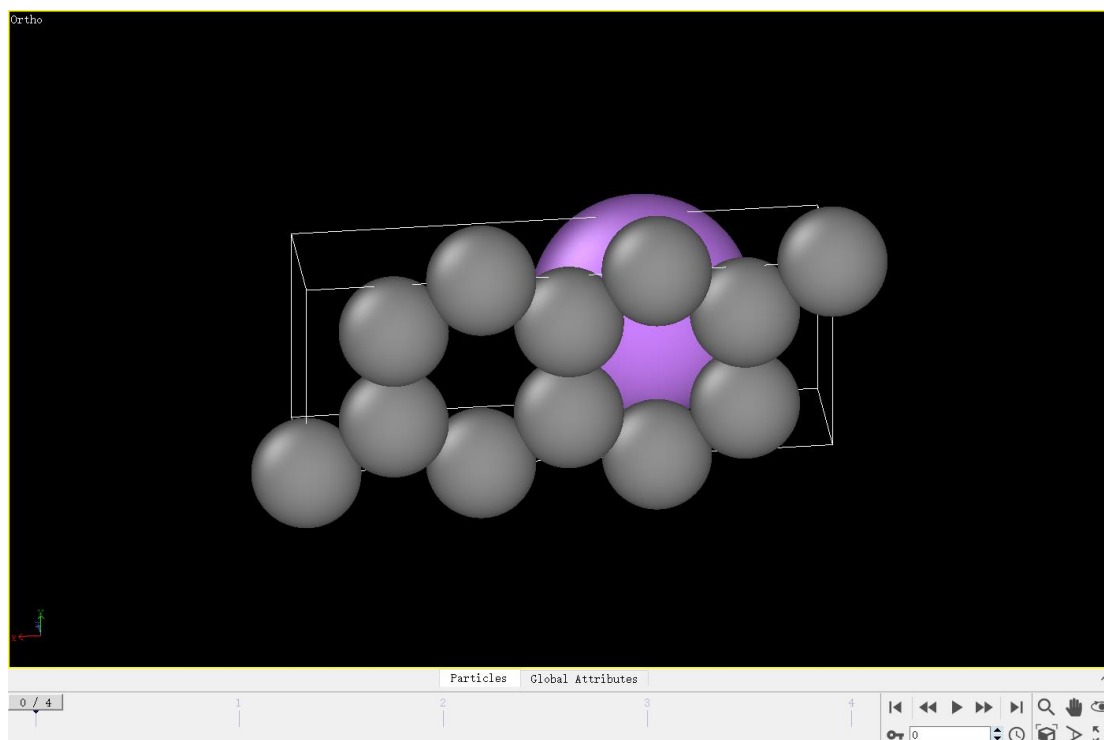
*Description:*

After using VASPMATE --neb -insert points and calculate the intermediate state structure (each intermediate state folder should have the corresponding OUTCAR, CONTCAR two files), you can use this function to analyze and extract the calculated data, the output file is neb.dat and movie.xyz.

The neb.dat file is shown below. The first column indicates the structure serial number, the second column indicates the distance (i.e., the calculation of dist.pl for the two adjacent structures), the third column indicates the energy (with the initial state energy as the reference value), and the fourth column indicates the force (forces along the neb).

Number	Distance	Energy	Relative Energy	Force
0	0.000000	-111.885923	0.000000	0.000000
1	0.625837	-111.765665	0.120258	-0.248757
2	1.252108	-111.680083	0.205840	-0.001109
3	1.876502	-111.763987	0.121936	0.247196
4	2.500640	-111.885900	0.000023	0.000000

The movie.xyz file can be animated in VMD or the whole process can be viewed in OVITO (jmol). Here is the effect of the movie.xyz file imported into OVITO after the LiC<sub>12</sub> structure NEB has been calculated.



*Examples:*

VASPMATE --neb -out

### 3.6 Calculation of partial charge

In the literature we often see calculations of the charge densities corresponding to the top of the valence band and the bottom of the conduction band, and by analyzing the charge densities at these particular locations to show which specific atom or atoms contribute at the top of the valence band and at the bottom of the conduction band. In systems with surface states, it is also common to calculate the charge density corresponding to the surface states and thus analyze the mechanisms that give rise to the surface states. This section focuses on how to prepare the calculation of the band-side charge density by going through VASPMATE.

First we need to perform structural optimization and static self-consistency of the target structure (refer to the previous command) to obtain CHGCAR and WAVECAR files. For the calculation of the charge density at the band edge there are three modes: 1. specify the energy band  $b$  and  $k$  points; 2. specify the energy interval  $[E1, E2]$ ; 3. specify the region near the Fermi energy level ( $[E_f - EINT, E_f]$ ). VASPMATE provides the user with a default pcd template (refer to incar) and an easy way to modify it in the three modes.

*Mode:* **Partial Charge Density**

*Syntax:*

VASPMATE --i -pcd <mode>

*Description:*

For the above three modes, we provide four methods at <mode> -ib/-ik/-en/-ef to modify the selected energy band, the selected  $k$ -point, the energy interval and the width around the Fermi energy level. Note that the modification mode should correspond to the method in INCAR, where -ik and -ib are both method I and can be used consecutively (see example).

The following figure shows the incar\_pcd\_ik file produced by the command VASPMATE --i pcd\_ik, corresponding to method I.

```

#Decomposed Charge Density
ISTART = 1          (Job: 0 - new 1 - cont 2 - samecut)
ICHARG = 1          (Read charge : 1 - file 2 - atom 10 - const)
LPARD = .TRUE.      (Activate decomposed charge density)
LSEPB = .TRUE.      (Separately write PARCHG.nb by every band or not)
LSEPK = .TRUE.      (Separately write PARCHG.nk by every kpoint or not)

# Method I : Partial Charge for the specified BANDS and KPOINTS
IBAND = 20 21 22 23 (Set this parameters manually)
KPUSE = 1 2 3 4     (Set this parameters manually)

# * *****Notes * *****
# (1) Copy IBZKPT as KPOINTS for static calculation,
# (2) Band structure calculation.

```

*Examples:*

VASPMATE --i -pcd -ik 1 2 3 4

VASPMATE --i -pcd -ib 1 2 3 4

VASPMATE --i -pcd -ik 1 2 -ib 3 4

VASPMATE --i -pcd -en -10 -5

VASPMATE --i -pcd -ef -1

### 3.7 Charge density calculation

VASPMATE provides powerful and easy post-processing commands for differential charge density calculations in the target file format CHGCAR. charge density differential Charge Density Difference is one of the most important ways to study electronic structure, which allows to visualize the electron flow from the interaction of two structures and thus analyze the nature of chemical bonds.

The differential charge density is divided into the following categories.

1. the overall charge density minus the charge density of the two or more structures of which it is a part

$$\Delta\rho = \rho_{AB} - \rho_A - \rho_B$$

2. the difference in charge density before and after self-consistency, that is, the deforming charge density (deforming charge density)

$$\Delta\rho = \rho(AB_{self-consistent}) - \rho(AB_{atomic})$$

Note that the cell parameters, the number of atoms and the grid size must be kept consistent in all charge density calculations. In addition, for files with CHGCAR format such as electrostatic potential (LOCPOT), electron localization function (ELFCAR) and partial charge (PARCHG) density difference calculations can also be done in the same way.

*Mode:* **Split**

*Syntax:*

VASPMATE --vcd -split file(CHGCAR)

*Description:*

When using the spin polarization parameter (ISPIN=2), the CHGCAR will contain charge density and spin density, and this function is used to extract both separately. The parameter file is the CHGCAR file to be read, the default is the CHGCAR generated by VASP. the output files are CHGTOT.vasp, CHGSPIN.vasp, CHGSPIN\_UP.vasp and CHGSPIN\_DW.vasp, which can be read by VESTA or SPAMD.

*Examples:*

VASPMATE --vcd -split

**Mode: Charge Density Summation***Syntax:*

```
VASPMATE --vcd -sum <File_list>
```

*Description:*

This function is used to quickly calculate the summation of multiple charge density files. <File\_list> is the list of CHGCAR files to be processed, and the -sum command is used to sum multiple charge density files, the output files are CHGSUMM.vasp, both can be viewed with VESTA or SPAMD.

*Examples:*

```
VASPMATE --vcd -sum CHGCAR_C CHGCAR_O
```

**Mode: Charge Density Difference***Syntax:*

```
VASPMATE --vcd -diff <File_list>
```

*Description:*

This function is used to quickly calculate the difference value of multiple charge density files. <File\_list> is the list of CHGCAR files to be processed, and the -diff command is used for differential charge density calculation. The output file is CHGDIFF.vasp, both can be viewed with VESTA or SPAMD.

*Examples:*

```
VASPMATE --vcd -diff CHGCAR_CO CHGCAR_C CHGCAR_O
```

### 3.8 Visual Analysis of Real Space Wave Functions

VASPMATE extracts the plane wave coefficients of the Kohn-Sham (KS) orbit from the WAVECAR file and outputs the real-space wave function, the user simply specifies the corresponding k points and energy bands.

*Mode:* **WaveFunction Visualization**

*Syntax:*

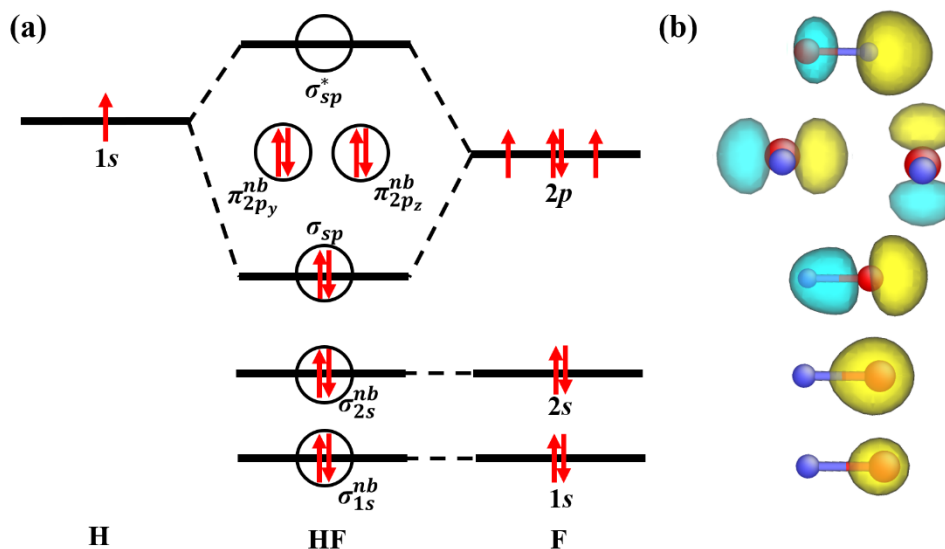
VASPMATE --wfun -k [kpoint\_index] -b [band\_index]

*Description:*

The output files are *wfn\_Bxxx\_Kxxx\_REAL.vasp* and *wfn\_Bxxx\_Kxxx\_IMAG.vasp*, which represent the real part and imaginary part of the wave function in real space respectively. The output files can be viewed by VESTA or SPAMD.

The following figure shows the calculated molecular orbital energy levels of HF, which indicates that the combination of H and F atomic orbitals forms HF with 6 molecular orbitals while the corresponding orbital energy gradually increases from bottom to top.





*Examples:*

VASPMATE --wfun

VASPMATE --wfun -k 10 -b 10

### 3.9 Fermi surface calculation

In condensed matter physics, the Fermi surface is commonly used to characterize the partition between occupied and unoccupied states, whose shape is determined by the periodicity and symmetry of the inverse easy space and the occupation of the electronic energy band. VASPMATE provides two modules for the calculation of the Fermi surface: 1) automatic generation of k points for Fermi surface calculations 2) generation of FERMISURFACE.bxsf or FERMISURFACE.frmsf file containing fermisurface data, which can be visualized with the XcrySDen [8] and fermisurfer [9] programs.

*Mode: Fermi Kpoints*

*Syntax:*

```
VASPMATE --fska [kppra] [kscheme]
```

```
VASPMATE --fskv [kspac] [kscheme]
```

```
VASPMATE --fskm k_1 k_2 k_3 [kmesh] [kscheme]
```

*Description:*

Similar to the automatic k-point generation method, VASPMATE provides three scattering methods, except that the k-points generated by the calculation of the Fermi surface are the integrable k-points within the first Brillouin zone after applying weights (this is similar to the way the HSE calculation generates k-points). We recommend using the --fskv uniform scattering method, and for the Fermi surface calculation we would like to use a higher precision, with a kspac recommendation of 0.05, and generate a FERMIKPT file to save the k-point information.

*Examples:*

```
VASPMATE --fskv 0.05
```

**Mode: Fermi Surface***Syntax:*

```
VASPMATE --fsxd -b <band_index>
```

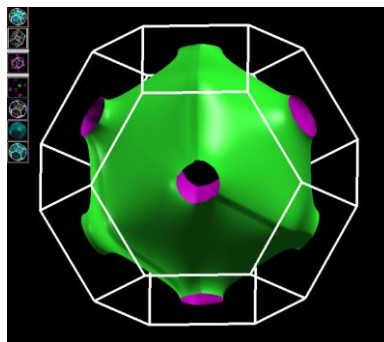
```
VASPMATE --fs -b <band_index> -o <atom&orbit index>
```

*Description:*

For the post-processing part of the Fermi surface calculation, VASPMATE provides the corresponding interfaces to the two mainstream Fermi surface analysis

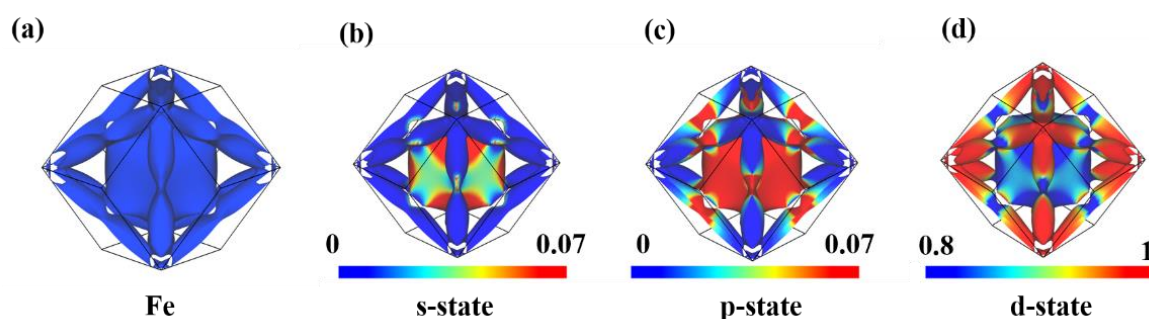
software XcrySDen and fermisurfer. VASPMATE processes the energy level data by extracting them from the EIGENVAL file and assigning them to each k-point according to the symmetry, in addition to comparing the In addition, by comparing the energy level values in the EIGENVAL file, the energy band closest to or across the Fermi energy level (the Fermi energy level size can be specified in the FERMI\_LEVEL file, the Fermi energy level in DOSCAR is read by default, for more details refer to the introduction of density of states calculation) is found as the Fermi surface, and finally the data is output to the corresponding file. In addition to the above default selection of energy bands near the Fermi energy level, users can specify the energy band number they wish to obtain by the -b parameter. The following two output methods are described separately.

The first command form - fsxd means to export the Fermi surface data to the recognizable format of XcrySDen program. The figure below shows the FERMISURFACE.bxsf file obtained from VASPMATE processing imported into XcrySDen after Cu calculation to obtain its Fermi surface visualization graph.



However, XcrySDen can only recognize fermionic planes without any weights, compared to the fermisurfer program, which is more complete. This is similar to how VASPMATE handles the energy band weights. Select the atom\_index with the -o

parameter and the atomic number or element symbol, and the corresponding orbit with the orbit name (note that when LORBIT=11 is set, VASPMATE can still recognize the spdf orbit, and the output value will be the sum of the corresponding split orbit). The following figure shows the visualization of the Fermi surface data file FERMISURFACE.frmsf obtained by VASPMATE after the Cu calculation, imported into SPaMD or fermisufer, from left to right, without weights, and with the spd orbitals of Fe atoms contributing to the weights, respectively.



*Examples:*

```
VASPMATE --fsxd
```

```
VASPMATE --fsxd -b 1 2
```

```
VASPMATE --fs -o B s dx
```

```
VASPMATE --fs -b 1 2 -o 1 s
```

### 3.10 3D energy band

For two-dimensional materials such as graphene, VASPMATE can automatically generate k-point files by scattering points on the surface of the integrable Brillouin zone, and the results of VASP calculations can be processed by VASPMATE to generate k-

point and energy files that are easy to draw with origin.

*Mode: **3dband Kpoints***

*Syntax:*

VASPMATE --3dka [kppra] [kscheme]

VASPMATE --3dkv [kspac] [kscheme]

VASPMATE --3dkm k\_1 k\_2 k\_3 [kmesh] [kscheme]

*Description:*

The k-point of 3d energy band is also a continuation of the VASPMATE scattering style, which provides three different scattering methods. We recommend using the --3dkv uniform scattering method, for the 3D energy band calculation we want to use a higher accuracy, kspac recommended value of 0.05, generate 3dbandKPT file to save k-point information.

*Examples:*

VASPMATE --3dkv 0.05

*Mode: **3dband***

*Syntax:*

VASPMATE --3db -b <band\_index>

*Description:*

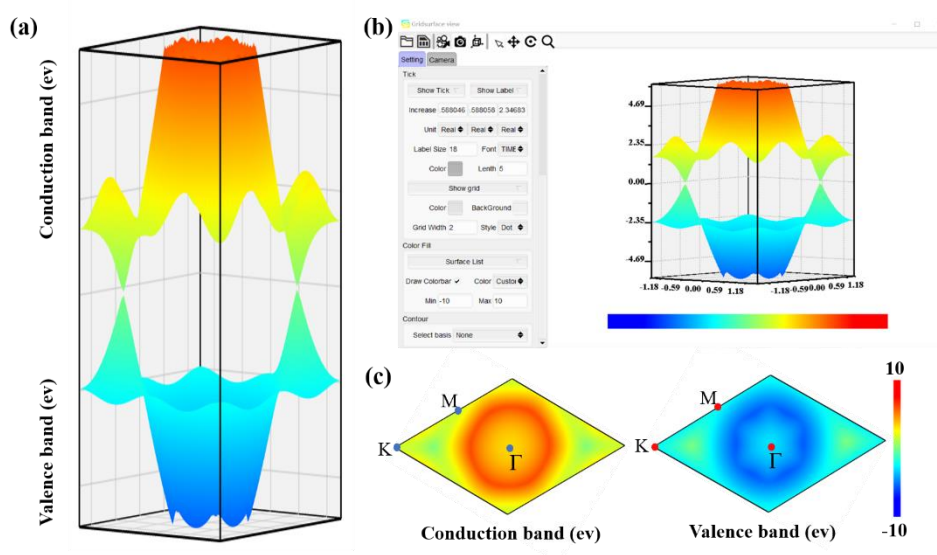
For the VASP output file after the calculation of 3d energy bands, VASPMATE provides two ways to process it. One is to specify the energy band number, i.e., to output the energy band with the corresponding number, and to specify more than one energy

band at a time, controlled by the parameter `-b`, and output them to the `BAND_B*.grd` file (for the system with spin, they are output to the `BAND_B*_UP.grd` and `BAND_B*_DW.grd` files respectively); the second is not to specify the number, VASPMATE automatically finds the energy bands where LUMO and HOMO are located and outputs their structures. In addition, VASPMATE will also output two files, `KX.grd` and `KY.grd`, which store the x and y coordinates of each k point in the inverse easy space, respectively. By importing the above three files into Origin or SPaMD and using 3D color mapped surfaces for mapping, we can finally obtain the following energy band structure (this figure imports two energy bands near the Fermi energy level).

*Examples:*

VASPMATE --3dbs

VASPMATE --3dbs -b 5 6



### 3.11 Thermodynamic quantity correction

VASP is an excellent package for ab initio first principles calculations, but the thermodynamic calculations are not satisfactory, as it can only calculate the electron energy of the system at 0 K, and cannot take into account the contribution of temperature, which obviously does not meet the needs when we calculate the free energy, the enthalpy to obtain the heat of reaction, and the ZPE to obtain the internal energy of the system at 0 K. In contrast, Gaussian [10] does a better job in this respect, but it is obviously not satisfying for users who are used to VASP, and the calculations are relatively cumbersome. Gaussian does a better job at this point, but it is clearly not satisfying for users used to VASP and the calculations are relatively cumbersome. The good news is that Sobereva has developed a very comprehensive thermodynamic calculation program, Sherma [11], which, among other things, calculates the internal energy, enthalpy, entropy, free energy, and heat capacity of a system at a given temperature and pressure by analyzing the output of Gaussian and calculating advection, rotation, vibration, and electronic contributions. It is highly recommended that users of VASPMATE read Sherma's appendix before calculating the thermodynamic data, which provides a detailed description of the calculation of each physicochemical property and the final contribution, which can be very helpful in understanding the output. Nevertheless, Sherma does not provide direct assistance to VASP users because it does not recognize the VASP output files and it is not user-friendly for high throughput calculations. Finally, VASPMATE integrates the above features and streamlines the parts that may not be of interest to VASP calculators, and finally designs

the module for easy and efficient thermodynamic correction of VASP output structures.

*Mode:* **Thermo**

*Syntax:*

VASPMATE --thermo <key-word> <key-value>

*Description:*

To provide convenient and accurate thermodynamic correction, the module provides a total of six control parameters and their appropriate default values, namely -T/-t , -P/-p , -sm/-s , -im /-i , -lf/-f , -cv/-v , which are described below one by one.

1) -T/-t 300 This parameter is used to set the thermodynamic temperature in K. The default is 298.15K.

2) -P/-p 1 This parameter is used to set the current atmospheric pressure in Atm, that is, the current actual air pressure is  $p \times 101.375 \text{ kPa}$ , and the default is 1 atm.

3) -sm/-s 1 This parameter is used to set the spin multiplicity of the current system Spin multiplicity can be simply set to the number of single electrons +1, the default value of this parameter is 1, but we prefer to be set by the user for different calculation systems, the wrong spin multiplicity will largely affect the rotation contribution.

4) -im /-i 1 This parameter is used to set the imode during the calculation, i.e., whether to ignore the contribution of advection and rotation to the thermodynamics, which plays a big role in the calculation of the free energy of the adsorbed molecules and will be further described in the following, and defaults to 0, i.e., all contributions are considered.

5) -lf/-f 1 This parameter is used to set the ilowfreq parameter, i.e., the treatment of



lower frequencies. It is well known that the smaller the vibration frequency the larger the contribution to the entropy. Very small vibrational frequencies are likely to lead to anomalous entropy and free energy corrections. According to the way of correction for lower frequencies mentioned by Sobereva in Shermo, VASPMATE provides three frequency treatments accordingly. The first one is the harmonic approximation (Harmonic approximation), also called the rigid rotor resonator model (RRHO), which corresponds to the  $-l=0$  case and is also the default parameter setting. However, the common RRHO model is prone to large errors in the calculation of the free energy at lower frequencies and thus has a very unstable performance. The second treatment is Raising low frequencies [12, 13], i.e., artificially raising the low frequencies to a certain level, e.g., VASPMATE is set to the default value of  $100\text{cm}^{-1}$ , which will improve the free energy calculation to a great extent. The third treatment is Grimme's entropy interpolation [14], which essentially interpolates between the harmonic approximation ( $S_{\text{RRHO}}$ ) and the vibration entropy calculated under the free rotor model ( $S_{\text{FR}}$ ), which seems more elegant than artificial intervention to raise the threshold. We recommend turning on  $-l\ 1$  for general free energy calculations to attenuate the effect of low frequencies on the results, while for more flexible systems (e.g., large molecules with many rotatable bonds or molecular complexes), we recommend setting it to 2.

6)  $-cv/-v\ 100$  This parameter corresponds to the second frequency processing mode, i.e., artificially set the frequency threshold, i.e., critical vibration critical vibra, the default value is  $100\text{cm}^{-1}$ .

*Examples:*

```
VASPMATE --thermo -T 300 -p 1 -s 3 -i 1 -l 1 -v 100
```

For the output, VASPMATE will print on the screen what is of most interest to the VASP thermodynamic calculator, i.e. the corrected thermodynamic quantities, and the user can save the output to a file using the linux redirection symbol `>>`, e.g. `VASPMATE --thermo -T 300 -p 1 -s 3 -i 1 -l 1 -r 100 &gt; H2O.dat`, which will facilitate the extraction of thermodynamic data in high throughput. In addition to this, VASPMATE will also output a sum of the respective thermodynamic contributions of advection, rotation, vibration, and electrons, as well as the final contribution, which will allow the user to further understand the entire thermodynamic calculation process. The following are two examples of the thermodynamic correction of gas-phase molecules and the thermodynamic correction of adsorbed molecules to further describe the entire process.

### **3.11.1 Thermodynamic correction of gas-phase molecules**

Let us take as an example the thermodynamic correction of the  $\text{H}_2\text{O}$  molecule, where we use the default value `imode = 0` in the molecular frequency calculation, i.e., we consider both translational and rotational contributions to the thermodynamic contribution. For linear molecules, the smallest five frequencies will be neglected, and for nonlinear molecules, the smallest six frequencies will be neglected (this does not and directly neglect the contribution of advection and rotation. Rather, the contribution to the thermodynamic quantities is calculated separately from the partition functions of advection and rotation). The advective entropy is the main contribution of the entropy

of the gas molecules). We use the command `VASPMATE --thermo -T 300 -p 1 -s 1 -l 1`, after which we will get the following information on the screen.

```
[roo@localhost freq]$ VASPMATE --thermo -T 300 -p 1 -s 1 -l 1
=====Thermo Energy Calculation!=====
Temperature(K): 300.000000
Pressure(Atm): 1
Spin multiplicity(Number of Unpaired electron + 1): 1
The treatment of low frequencies: Raising low frequencies.
Raising lower frequencies to 50.000000 (cm^-1)!
Molecular Symmetry: C2v
Zero point energy (ZPE): 13.102430 kcal/mol 0.568175 ev
Thermal correction to U: 14.893164 kcal/mol 0.645829 ev
Thermal correction to H: 15.489325 kcal/mol 0.671681 ev
Thermal correction to G: 1.927806 kcal/mol 0.083598 ev
Total S : 189.137994 J/mol/K 0.001960 ev/K
TS : 56741.398078 J/mol 0.588083 ev
More calculate details please read Thermol_Info.dat file!
```

From the VASPMATE output we can see the input parameters used for the current calculation, such as temperature, pressure, low frequency processing method and the point group of this molecule. Among the various output quantities, to obtain the free energy of H<sub>2</sub>O, we simply add the G value, i.e. 0.083598 for the same species, to the energy finally calculated by VASP. In addition to this in the output file Thermol\_Info.dat file, we will first see the information about the calculated system and the calculated input quantities

```
===== Thermo Energy Calculated by VASPMATE =====
===== Calculate Inparameter =====
Temperature(K): 300.000000
Pressure(Atm): 1
Spin multiplicity(Number of Unpaired electron + 1): 1
The treatment of low frequencies: Raising low frequencies.
Raising lower frequencies to 50.000000 (cm^-1)!
Note: Only for translation, U is different to H, and CV is different to CP
```

VASPMATE then outputs the respective contributions of advection, rotation, vibration and electrons to the thermodynamic quantities.

```

===== Translation =====
Translational q: 1.838161E+30      q/NA: 3.052338E+06
Translational U: 3.741509 kJ/mol 0.894242 kcal/mol
Translational H: 6.235849 kJ/mol 1.490404 kcal/mol
Translational S: 144.932915 J/mol/K 34.639798 cal/mol/K -TS: -10.391939 kcal/mol
Translational CV: 12.471697 J/mol/K 2.980807 cal/mol/K
Translational CP: 20.786162 J/mol/K 4.968012 cal/mol/K

===== Rotation =====
Rotational q: 4.525796E+01
Rotational U: 3.741509 kJ/mol 0.894242 kcal/mol =H
Rotational S: 44.169584 J/mol/K 10.556784 cal/mol/K -TS: -3.167035 kcal/mol
Rotational CV: 12.471697 J/mol/K 2.980807 cal/mol/K =CP

===== Vibration =====
Vibrational q(V=0): 1.000496
Vibrational q(bot): 0.000000
Vibrational U(T)-U(0): 0.009412 kJ/mol 0.002250 kcal/mol =H(T)-H(0)
Vibrational U: 54.829979 kJ/mol 13.104679 kcal/mol =H
Vibrational S: 0.035495 J/mol/K 0.008483 cal/mol/K -TS: -0.002545 kcal/mol
Vibrational CV: 0.238908 J/mol/K 0.057100 cal/mol/K =CP
Zero-point energy (ZPE): 54.820567 kJ/mol 13.102430 kcal/mol 0.020880 a.u.

===== Electron =====
Electronic q: 1.000000
Electronic U: 0.000000 kJ/mol 0.000000 kcal/mol =H
Electronic S: 0.000000 J/mol/K 0.000000 cal/mol/K -TS: -0.000000 kcal/mol
Electronic CV: 0.000000 J/mol/K 0.000000 cal/mol/K =CP

```

The final output is the sum of the contributions and the corrected thermodynamic results.

Where 'Electronic energy' is the VASP calculated electronic energy -14.219011ev, by adding the VASPMATE to the Gibbs free energy correction 'G' 0.083598ev, the final obtained 'Sum of electronic energy and thermal correction to G' -14.135414ev which is the free energy of the H<sub>2</sub> O molecule at the current temperature pressure.

```

===== Total =====
Total q(V=0): 8.323267E+31
Total q(bot): 2.373468E+22
Total q(V=0)/NA: 1.382111E+08
Total q(bot)/NA: 3.941236E-02
Total CV: 25.182302 J/mol/K 6.018715 cal/mol/K
Total CP: 33.496767 J/mol/K 8.005919 cal/mol/K
Total S: 189.137994 J/mol/K 45.205065 cal/mol/K -TS: -56741.398078 J/mol -0.588083 ev
Zero point energy (ZPE): 54.820567 kJ/mol/K 13.102430 kcal/mol 0.020880 a.u. 0.568175 ev
Thermal correction to U: 62.312997 kJ/mol/K 14.893164 kcal/mol 0.023734 a.u. 0.645829 ev
Thermal correction to H: 64.807337 kJ/mol/K 15.489325 kcal/mol 0.024684 a.u. 0.671681 ev
Thermal correction to G: 8.065938 kJ/mol/K 1.927806 kcal/mol 0.003072 a.u. 0.083598 ev
Electronic energy: -0.522539 a.u. -14.219011 ev
Sum of electronic energy and ZPE, namely U/H/G at 0 K: -0.501659 a.u. -13.650836 ev
Sum of electronic energy and thermal correction to U: -0.498805 a.u. -13.573183 ev
Sum of electronic energy and thermal correction to H: -0.497855 a.u. -13.547331 ev
Sum of electronic energy and thermal correction to G: -0.519467 a.u. -14.135414 ev

Thanks to Sobereva!(sobereva@sina.com)
More information about Thermol and Gaussian please refer to http://sobereva.com/552 and https://gaussian.com/thermo/

```

### 3.11.2 Thermodynamic correction of adsorbed molecules

Unlike gas molecular free energy calculations, the adsorbed molecule forms a

chemical bond with the substrate, which reduces the degrees of freedom of translation and rotation, thus limiting the contribution of translation and rotation to entropy and enthalpy. The approach used by VASPMATE for this is to attribute the contribution of translation and rotation to vibration, i.e., all  $3N$  degrees of freedom of the adsorbed surface molecule are attributed to vibration for thermodynamic calculations (note that VASPMATE ignores the (note that VASPMATE ignores the role of imaginary frequencies), and these operations can be performed by simply setting the imode parameter to 0 (note that when imode is 0, VASPMATE will also ignore the contribution of electron motion because it is very small at this point). Let's take the example of the adsorption of  $\text{PO}_2$  molecules on the graphite surface and enter the command VASPMATE --thermo -T 300 -p 1 -s 1 -l 1 -i 1, which will give us the following result on the screen

```
[roo@localhost ORR]$ VASPMATE --thermo -T 300 -p 1 -s 1 -l 1 -i 1
=====Thermo Energy Calculation!=====
Temperature(K): 300.000000
Pressure(Atm): 1
Spin multiplicity(Number of Unpaired electron +1): 1
The treatment of low frequencies: Raising low frequencies.
Raising lower frequencies to 50.000000 (cm^-1)!
Molecular Symmetry: NULL!
Warning: Cannot identify point group! Assume rotational symmetry number to be 1!
Zero point energy (ZPE): 4.481927 kcal/mol 0.194355 ev
Thermal correction to U: 6.482744 kcal/mol 0.281118 ev
Thermal correction to H: 6.482744 kcal/mol 0.281118 ev
Thermal correction to G: 2.885191 kcal/mol 0.125114 ev
Total S          : 50.173881 J/mol/K 0.000520 ev/K
TS              : 15052.164394 J/mol 0.156005 ev
More calculate details please read Thermol_Info.dat file!
```

The zero energy ZPE and the U, H, and G corrected results can be clearly seen, and by comparing the data with the Shermo output is completely consistent, demonstrating the accuracy of the VASPMATE calculation results.

### 3.12 Calculation of enthalpy of formation of compounds

Mode: Enthalpy of formation

Syntax:

VASPMATE --enth enx eny enz ...

Description:

The enthalpy of formation of the compound  $A_xB_yC_z...$   $\Delta H$  is calculated by the following equation

$$\Delta H = \frac{1}{x + y + z + \dots} [ H(A_xB_yC_z) - x * H(A) - y * H(B) - z * H(C) - \dots ]$$

enx, eny, enz are the corresponding  $H(A)$ ,  $H(B)$ ,  $H(C)$ , representing the total enthalpy of A, B, C. VASPMATE further calculates the enthalpy of formation of the corresponding compound by reading the energy information in OUTCAR under the current folder.

Examples:

VASPMATE --enth -6.70392 -4.94403 (calculates the enthalpy of formation of BO binary compound)

### 3.13 High throughput programs: ABAND and AEDOS

VASPMATE adds the energy band and density of states high throughput calculation module to the integrated platform SPaMD developed by the group, and designs the high throughput first principles calculation GUI programs ABAND and

AEDOS, which transform the cumbersome command line and script format into a friendly GUI interactive interface, and greatly improve the efficiency of energy band and density of states calculation. Detailed descriptions of this part of the functionality can be found in the manuals of AEDOS and ABAND.

## 4 Examples

### 4.1 B<sub>6</sub>N energy band structure calculation (PBE generalized function)

```

    #! /bin/bash
    VASPMATE_DIR=''
    VASP_CAL=''

    cp . /structure/* . /INPOS

    VASPMATE --kpt3d 20
    #>>> create POSCAR file
    cp PRIMPOS POSCAR
    cp POSCAR INPOS

    #-----Relaxtion-----
    #>>> create INCAR file using VASPMATE and the files incar_rlx and
    incar_stc are created
    ${VASPMATE_DIR}/VASPMATE --i rlx
    ${VASPMATE_DIR}/VASPMATE --i stc
    #>>> HSE calculation
    ${VASPMATE_DIR}/VASPMATE --i hse stc
    cp incar_rlx INCAR
    #>>> correct parameters
    ${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
    ${VASPMATE_DIR}/VASPMATE --i_replace ENCUT 400
    #>>> create KPOINTS file
    ${VASPMATE_DIR}/VASPMATE --ka 4000
    cp NEWKPT KPOINTS
    #>>> create POSAR file
    cp INPOS POSCAR
    #>>> create POTCAR file
    ${VASPMATE_DIR}/VASPMATE --pot -PBE
    ${VASP_CAL} >> log1.vasp
    #-----

    #-----option(self-consistent)-----
    cp CONTCAR INPOS
    cp incar_stc INCAR
    ${VASPMATE_DIR}/VASPMATE --i_replace LCHARG T
    ${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
    ${VASPMATE_DIR}/VASPMATE --i_replace ICHARG 2

```



```

${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
cp INPOS POSCAR
${VASP_CAL} >> log2.vasp
#-----

#>>> get the efermi energy from the self-consistent calculation
${VASPMATE_DIR}/VASPMATE --dos -efermi >> FERMI_LEVEL

#-----band calculation-----
#>>> Bandstructure calculation
mkdir band
cd band
cp ... /CONTCAR POSCAR
cp ... /POTCAR . /
cp ... /CHGCAR . /
cp ... /NEWKPATH . /NEWKPATH .
cp ... /FERMI_LEVEL . /
#>>> create KPOINTS file
cp NEWKPATH KPOINTS
#>>> HSEcalculaiton
#VASPMATE --kahse 8000 0.05 G
#>>> create INCAR file
${VASPMATE_DIR}/VASPMATE --i pbs
cp incar_pbs INCAR
${VASP_CAL} >> log3.vasp

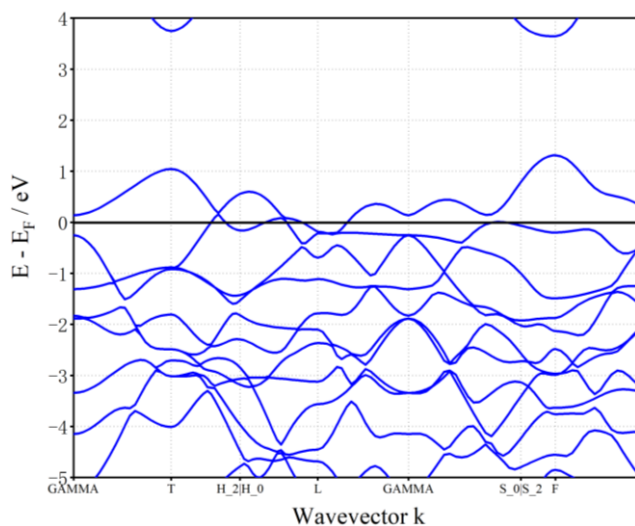
#-----

${VASPMATE_DIR}/VASPMATE --band -bg
#>>>select output mode
#${VASPMATE_DIR}/VASPMATE --band -b >> VASPMATE.log
#${VASPMATE_DIR}/VASPMATE --band -a
#${VASPMATE_DIR}/VASPMATE --band -e
#${VASPMATE_DIR}/VASPMATE --band -s 1-4 N
#${VASPMATE_DIR}/VASPMATE --band -m 1-4 N
#${VASPMATE_DIR}/VASPMATE --band -o 1-3 s px py

#rm POSCAR KPOINTS* INCAR* POTCAR
#rm CHG* CONTCAR DOSCAR OSZICAR OUTCAR EIGENVAL PCDAT WAVECAR
XDATCAR IBZKPT vasprun.xml
#END

```

Energy band plot of B<sub>6</sub>N based on *Band.txt* and *KLABELS* using origin.



## 4.2 B<sub>6</sub>N energy band structure calculation (HSE generalized function)

```

    #!/bin/bash
    VASPMATE_DIR=''
    VASP_CAL=''

    cp ./structure/* ./INPOS

    VASPMATE --kpt3d 20
    #>>> create POSCAR file
    cp PRIMPOS POSCAR
    cp POSCAR INPOS

    #-----Relaxtion-----
    #>>> create INCAR file using VASPMATE and the files incar_rlx and
    incar_stc are created
    ${VASPMATE_DIR}/VASPMATE --i rlx
    ${VASPMATE_DIR}/VASPMATE --i stc
    #>>> HSE calculation
    #${VASPMATE_DIR}/VASPMATE --i hse stc
    cp incar_rlx INCAR
    #>>> correct parameters
    #${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
    #${VASPMATE_DIR}/VASPMATE --i_replace ENCUT 400
    #>>> create KPOINTS file
    ${VASPMATE_DIR}/VASPMATE --ka 4000
    cp NEWKPT KPOINTS
    #>>> create POSAR file
    cp INPOS POSCAR
  
```

```

#>>> create POTCAR file
${VASPMATE_DIR}/VASPMATE --pot -PBE
${VASP_CAL} >> log1.vasp
#-----

#-----option(self-consistent)-----
cp CONTCAR INPOS
cp incar_stc INCAR
${VASPMATE_DIR}/VASPMATE --i_replace LCHARG T
# ${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
# ${VASPMATE_DIR}/VASPMATE --i_replace ICHARG 2
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
cp INPOS POSCAR
${VASP_CAL} >> log2.vasp
#-----

#>>> get the efermi energy from the self-consistent calculation
${VASPMATE_DIR}/VASPMATE --dos -efermi >> FERMI_LEVEL

#-----hse band calculation-----
#>>> Bandstructure calculation
mkdir band
cd band
cp ... /CONTCAR POSCAR
cp ... /POTCAR . /
cp ... /CHGCAR . /
cp ... /NEWKPATH . /NEWKPATH .
cp ... /FERMI_LEVEL . /
#>>> create KPOINTS file
#cp NEWKPATH KPOINTS
#>>> HSEcalculaiton
VASPMATE --kahse 8000 0.05 g
cp NEWKPT KPOINTS
#>>> create INCAR file
${VASPMATE_DIR}/VASPMATE --i pbs hse
cp incar_pbs_hse INCAR
${VASP_CAL} >> log3.vasp

#-----

${VASPMATE_DIR}/VASPMATE --band -bg
#>>>select output mode
# ${VASPMATE_DIR}/VASPMATE --band -hb >> VASPMATE.log
# ${VASPMATE_DIR}/VASPMATE --band -ha

```

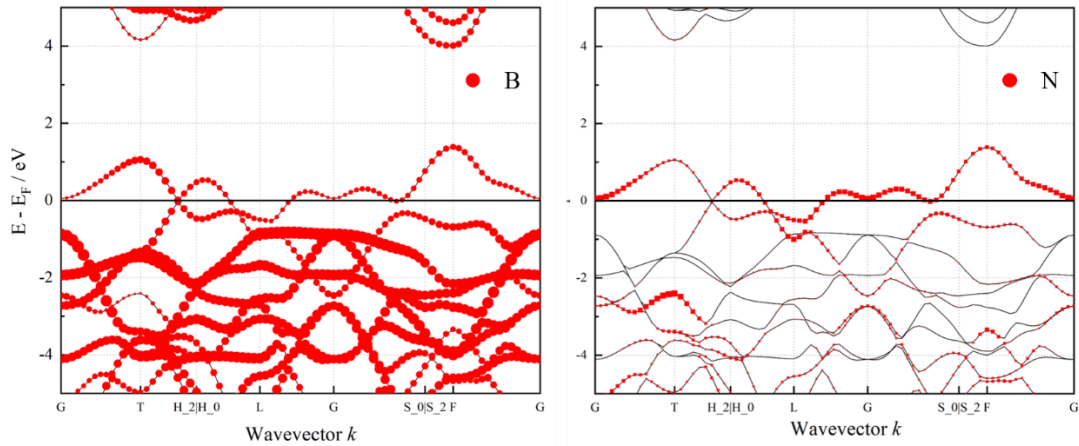
```

# ${VASPMATE_DIR}/VASPMATE --band -he
# ${VASPMATE_DIR}/VASPMATE --band -hs 1-4 N
# ${VASPMATE_DIR}/VASPMATE --band -hm 1-4 N
# ${VASPMATE_DIR}/VASPMATE --band -ho 1-3 s px py

#rm POSCAR KPOINTS* INCAR* POTCAR
#rm CHG* CONTCAR DOSCAR OSZICAR OUTCAR EIGENVAL PCDAT WAVECAR
XDATCAR IBZKPT vasprun.xml

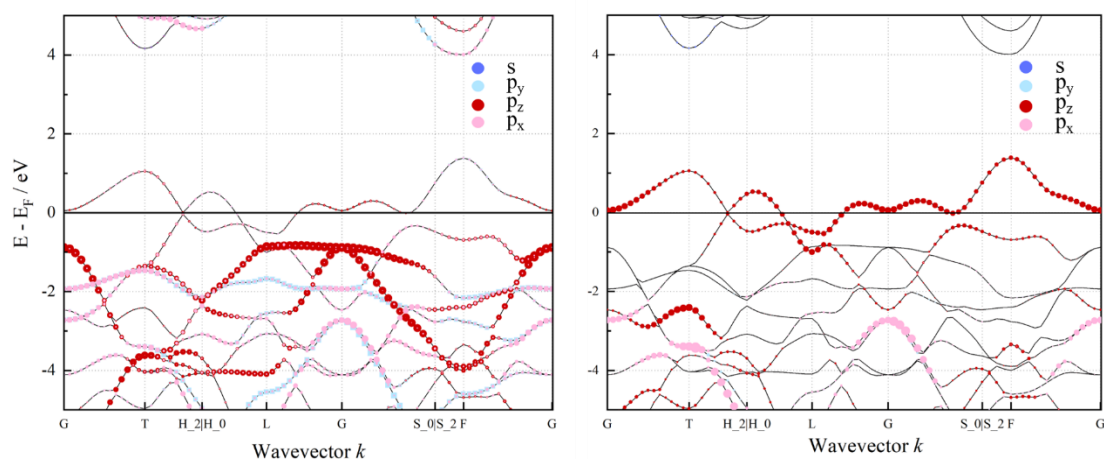
#END

```



The fatband maps drawn by *PBAND\_B.dat* and *PBAND\_N.dat* clearly show the contribution of different elements to the energy band.

Besides, by choosing the contributions of different orbitals as weights for plotting, the role of each orbital on the energy band can be seen more clearly. The following figure is a fatband plot based on the contributions of  $s$ ,  $p_y$ ,  $p_z$ ,  $p_x$  orbitals in B and N elements, and it can be seen that  $p_z$  and  $p_x$  orbitals play a dominant role in the  $B_6N$  system.



### 4.3 B<sub>6</sub>N density of states calculation

```

#!/bin/bash
VASPMATE_DIR=' '
VASP_CAL=' '

cp . /structure/* . /INPOS

#>>> create POSCAR file
cp INPOS POSCAR

#-----Relaxation-----
#>>> create INCAR file using VASPMATE and the files incar_rlx and
incar_stc are created
${VASPMATE_DIR}/VASPMATE --i rlx
${VASPMATE_DIR}/VASPMATE --i stc
#>>> HSE calculation
${VASPMATE_DIR}/VASPMATE --i hse stc
cp incar_rlx INCAR
#>>> correct parameters
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
${VASPMATE_DIR}/VASPMATE --i_replace ENCUT 400
#>>> create KPOINTS file
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
#>>> create POSAR file
cp INPOS POSCAR
#>>> create POTCAR file
${VASPMATE_DIR}/VASPMATE --pot -PBE

```

```

${VASP_CAL} >> log1.vasp
#-----

#-----option(self-consistent)-----
cp CONTCAR INPOS
cp incar_stc INCAR
${VASPMATE_DIR}/VASPMATE --i_replace LCHARG T
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
${VASPMATE_DIR}/VASPMATE --i_replace ICHARG 2
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
cp INPOS POSCAR
${VASP_CAL} >> log2.vasp
#-----

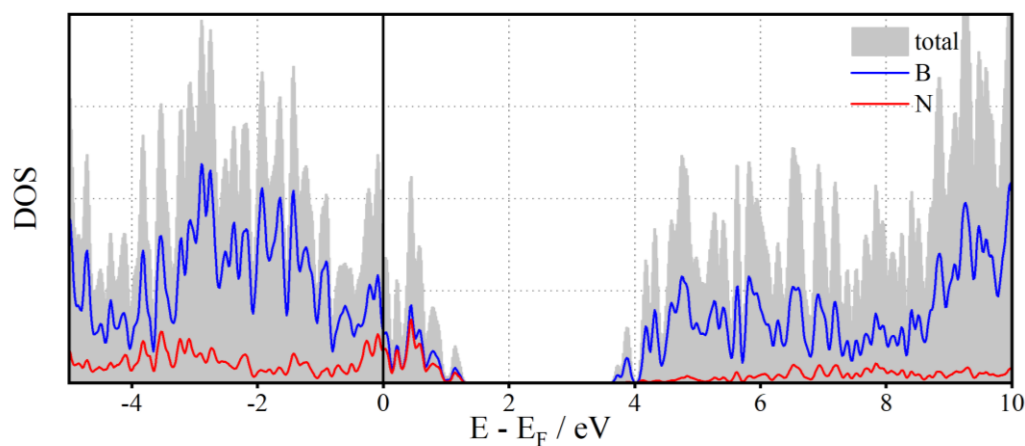
#-----dos calculation-----
#>>> DOS calculation
mkdir dos
cd dos
cp ... /CONTCAR POSCAR
cp ... /POTCAR . /
cp ... /CHGCAR . /
#>>> create KPOINTS file
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
#>>> create INCAR file
${VASPMATE_DIR}/VASPMATE --i dos
${VASPMATE_DIR}/VASPMATE --i_replace ICHARG 11
${VASPMATE_DIR}/VASPMATE --i_replace LORBIT 11
${VASPMATE_DIR}/VASPMATE --i_replace NEDOS 1000
cp incar_dos INCAR
${VASP_CAL} >> log3.vasp
#-----

#>>>select output mode
${VASPMATE_DIR}/VASPMATE --dos -t >> VASPMATE.log
${VASPMATE_DIR}/VASPMATE --dos -a
${VASPMATE_DIR}/VASPMATE --dos -e
${VASPMATE_DIR}/VASPMATE --dos -s 1-4 N
${VASPMATE_DIR}/VASPMATE --dos -m 1-4 N
${VASPMATE_DIR}/VASPMATE --dos -o 1-3 s px py
${VASPMATE_DIR}/VASPMATE --dos -bc all
#rm POSCAR KPOINTS* INCAR* POTCAR

```

```
#rm CHG* CONTCAR DOSCAR OSZICAR OUTCAR EIGENVAL PCDAT WAVECAR
XDATCAR IBZKPT vasprun.xml
#END
```

Density of states plotted with *TDOS.dat*, *PDOS\_B.dat* and *PDOS\_N.dat* for B<sub>6</sub>N



## 4.4 Calculation of energy bands and density of states of Fe (with spin)

Shell capable

```
#!/bin/bash
VASPMATE_DIR=' '
VASP_CAL=' '

cp ./structure/* ./INPOS

VASPMATE --kpt3d 20
#>>> create POSCAR file
cp PRIMPOS POSCAR
cp POSCAR INPOS

#-----Relaxation-----
#>>> create INCAR file using VASPMATE and the files incar_rlx and
incar_stc are created
${VASPMATE_DIR}/VASPMATE --i rlx
${VASPMATE_DIR}/VASPMATE --i stc
#>>> HSE calculation
#${VASPMATE_DIR}/VASPMATE --i hse stc
cp incar_rlx INCAR
```

```

#>>> correct parameters
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
# ${VASPMATE_DIR}/VASPMATE --i_replace ENCUT 400
#>>> create KPOINTS file
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
#>>> create POSAR file
cp INPOS POSCAR
#>>> create POTCAR file
${VASPMATE_DIR}/VASPMATE --pot -PBE
${VASP_CAL} >> log1.vasp
#-----

#-----option(self-consistent)-----
cp CONTCAR INPOS
cp incar_stc INCAR
${VASPMATE_DIR}/VASPMATE --i_replace LCHARG T
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
# ${VASPMATE_DIR}/VASPMATE --i_replace ICHARG 2
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
cp INPOS POSCAR
${VASP_CAL} >> log2.vasp
#-----

#>>> get the efermi energy from the self-consistent calculation
${VASPMATE_DIR}/VASPMATE --dos -efermi >> FERMI_LEVEL

#-----band calculation-----
#>>> Bandstructure calculation
mkdir band
cd band
cp ... /CONTCAR POSCAR
cp ... /POTCAR . /
cp ... /CHGCAR . /
cp ... /NEWKPATH . /NEWKPATH .
cp ... /FERMI_LEVEL . /
#>>> create KPOINTS file
cp NEWKPATH KPOINTS
#>>> HSEcalculaiton
#VASPMATE --kahse 8000 0.05 G
#>>> create INCAR file
${VASPMATE_DIR}/VASPMATE --i pbs
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2

```



```

cp incar_pbs INCAR
${VASP_CAL} >> log3.vasp

#-----
${VASPMATE_DIR}/VASPMATE --band -bg
#>>>select output mode
${VASPMATE_DIR}/VASPMATE --band -b >> VASPMATE.log
${VASPMATE_DIR}/VASPMATE --band -a
${VASPMATE_DIR}/VASPMATE --band -e
${VASPMATE_DIR}/VASPMATE --band -s 1-4 N
${VASPMATE_DIR}/VASPMATE --band -m 1-4 N
${VASPMATE_DIR}/VASPMATE --band -o 1-3 s px py

#rm POSCAR KPOINTS* INCAR* POTCAR
#rm CHG* CONTCAR DOSCAR OSZICAR OUTCAR EIGENVAL PCDAT WAVECAR
XDATCAR IBZKPT vasprun.xml
#END

```

### Density of state Shell

```

#! /bin/bash
VASPMATE_DIR=''
VASP_CAL=''

cp . /structure/* . /INPOS

#>>> create POSCAR file
cp INPOS POSCAR

#-----Relaxtion-----
#>>> create INCAR file using VASPMATE and the files incar_rlx and
incar_stc are created
${VASPMATE_DIR}/VASPMATE --i rlx
${VASPMATE_DIR}/VASPMATE --i stc
#>>> HSE calculation
${VASPMATE_DIR}/VASPMATE --i hse stc
cp incar_rlx INCAR
#>>> correct parameters
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
${VASPMATE_DIR}/VASPMATE --i_replace ENCUT 400
#>>> create KPOINTS file
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS

```

```

#>>> create POSCAR file
cp INPOS POSCAR
#>>> create POTCAR file
${VASPMATE_DIR}/VASPMATE --pot -PBE
${VASP_CAL} >> log1.vasp
#-----

#-----option(self-consistent)-----
cp CONTCAR INPOS
cp incar_stc INCAR
${VASPMATE_DIR}/VASPMATE --i_replace LCHARG T
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
# ${VASPMATE_DIR}/VASPMATE --i_replace ICHARG 2
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
cp INPOS POSCAR
${VASP_CAL} >> log2.vasp
#-----

#-----dos calculation-----
#>>> DOS calculation
mkdir dos
cd dos
cp ... /CONTCAR POSCAR
cp ... /POTCAR . /
cp ... /CHGCAR . /
#>>> create KPOINTS file
${VASPMATE_DIR}/VASPMATE --ka 4000
cp NEWKPT KPOINTS
#>>> create INCAR file
${VASPMATE_DIR}/VASPMATE --i dos
${VASPMATE_DIR}/VASPMATE --i_replace ICHARG 11
${VASPMATE_DIR}/VASPMATE --i_replace ISPIN 2
# ${VASPMATE_DIR}/VASPMATE --i_replace LORBIT 11
# ${VASPMATE_DIR}/VASPMATE --i_replace NEDOS 1000
cp incar_dos INCAR
${VASP_CAL} >> log3.vasp
#-----

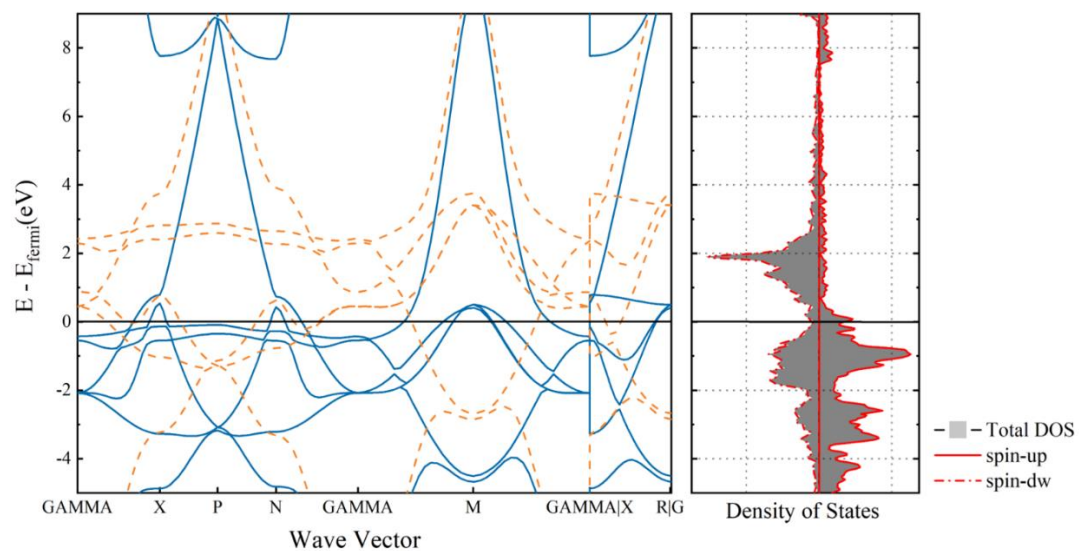
#>>>select output mode
${VASPMATE_DIR}/VASPMATE --dos -t >> VASPMATE.log
# ${VASPMATE_DIR}/VASPMATE --dos -a
# ${VASPMATE_DIR}/VASPMATE --dos -e
# ${VASPMATE_DIR}/VASPMATE --dos -s 1-4 N

```

```

# ${VASPMATE_DIR}/VASPMATE --dos -m 1-4 N
# ${VASPMATE_DIR}/VASPMATE --dos -o 1-3 s px py
# ${VASPMATE_DIR}/VASPMATE --dos -bc all
#rm POSCAR KPOINTS* INCAR* POTCAR
#rm CHG* CONTCAR DOSCAR OSZICAR OUTCAR EIGENVAL PCDAT WAVECAR
XDATCAR IBZKPT vasprun.xml
#END

```



## References

- [1] S. Curtarolo, W. Setyawan, G.L.W. Hart, M. Jahnatek, R.V. Chepulskii, R.H. Taylor, S. Wang, J. Xue, K. Yang, O. Levy, M.J. Mehl, H.T. Stokes, D.O. Demchenko, D. Morgan, *Computational Materials Science*, 58 (2012) 218-226.
- [2] V. Wang, N. Xu, J.-C. Liu, G. Tang, W.-T. Geng, *Computer Physics Communications*, 267 (2021).
- [3] Z.R. Liu, B.N. Yao, R.F. Zhang, *Computational Materials Science*, 210 (2022).
- [4] S.H. Zhang, Z.H. Fu, R.F. Zhang, *Computer Physics Communications*, 238 (2019) 244-253.
- [5] S.H. Zhang, R.F. Zhang, *Computer Physics Communications*, 220 (2017) 403-416.
- [6] Y. Hinuma, G. Pizzi, Y. Kumagai, F. Oba, I. Tanaka, *Computational Materials Science*, 128 (2017) 140-184.
- [7] S. Hastrup, M. Strange, M. Pandey, T. Deilmann, P.S. Schmidt, N.F. Hinsche, M.N. Gjerding, D. Torelli, P.M. Larsen, A.C. Riis-Jensen, J. Gath, K.W. Jacobsen, J. Jørgen Mortensen, T. Olsen, K.S. Thygesen, *2D Materials*, 5 (2018).
- [8] A. Kokalj, *Journal of Molecular Graphics and Modelling*, 17 (1999) 176-179.
- [9] M. Kawamura, *Computer Physics Communications*, 239 (2019) 197-203.
- [10] M. Frisch, <http://www.gaussian.com/>, (2009).
- [11] T. Lu, Q. Chen, *Computational and Theoretical Chemistry*, 1200 (2021).
- [12] R.F. Ribeiro, A.V. Marenich, C.J. Cramer, D.G. Truhlar, *J Phys Chem B*, 115 (2011) 14556-14562.
- [13] A.V. Marenich, W. Ding, C.J. Cramer, D.G. Truhlar, *J Phys Chem Lett*, 3 (2012) 1437-1442.
- [14] L. Xu, M.L. Coote, *J Phys Chem A*, 123 (2019) 7430-7438.