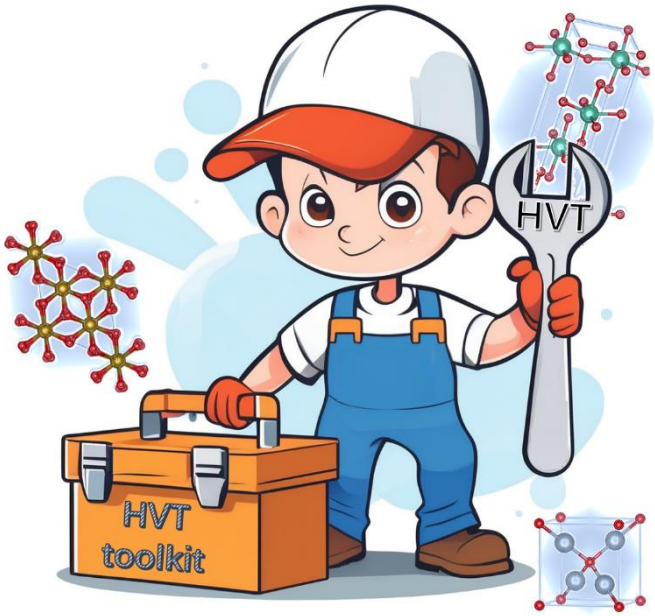


HVT

User's Guide Manual

Content

Introduction

With the advancement of algorithms and the surge in computing power, first-principles calculations have achieved unparalleled levels of precision and computational efficiency. These calculations, being both cost-effective and potent, are frequently employed to elucidate experimental mechanisms and forecast the properties of novel materials. First-principles research encompasses a wide array of systems, ranging from molecules and clusters to one-dimensional nanoribbons (or nanotubes), two-dimensional materials, and three-dimensional crystals. A plethora of software packages cater to materials computation needs, such as CP2K, VASP, WIEN2K, Materials Studio, GROMACS, and LAMMPS, among others. The majority of these packages offer Linux command line versions, renowned for their swiftness in computation and ease of code manipulation. Typically, first-principles simulation packages derive various physical properties of materials through a series of established procedural steps, including structure optimization, single-point energy calculations, band structure computations, phonon band analyses, and more. Often, the utilization of these packages necessitates the development of supplementary toolkits for preprocessing input files or postprocessing output files.

VASP stands out as one of the premier commercial software solutions for first-principles materials simulation. However, accessing its source codes is limited to the Linux platform, posing a challenge for newcomers, particularly those unfamiliar with Linux commands, as generating input files and processing output files often demands considerable expertise and a suite of auxiliary tools. In response, we have developed the HVT program to streamline the utilization of VASP, offering user-friendly interfaces and relatively independent program modules. The functionalities of the HVT program encompass the generation of VASP input files and the processing of VASP output files. Crafted using Python and Bash, our aim with the HVT program is to empower users to navigate material simulation and calculation with greater ease and efficiency.

Operating system and environmental requirements

If you need to use all the functions of HVT, be sure to set up the following environment:

- Python 3.9 or higher
- Numpy 1.26.0 or higher
- Scipy 1.11.3 or higher
- matplotlib 3.0.1 or higher
- pymatgen 2023.10.4 or higher
- pymatgen-analysis-diffusion 2023.8.15 or higher
- ase 3.22.1 or higher
- pandas 1.3.4 or higher

Runing Environment (Linux x64 Bit), we have tested it on:

Ubuntu >20.04 (Notice! need run "sudo dpkg-reconfigure dash" and choose "No")

Centos >6.0

RedHat >5.0

1 Overview of HVT

1.1 Functional features of HVT

HVT is based on the VASP package and is designed to help users get up to speed with vasp software with features including, but not limited to:

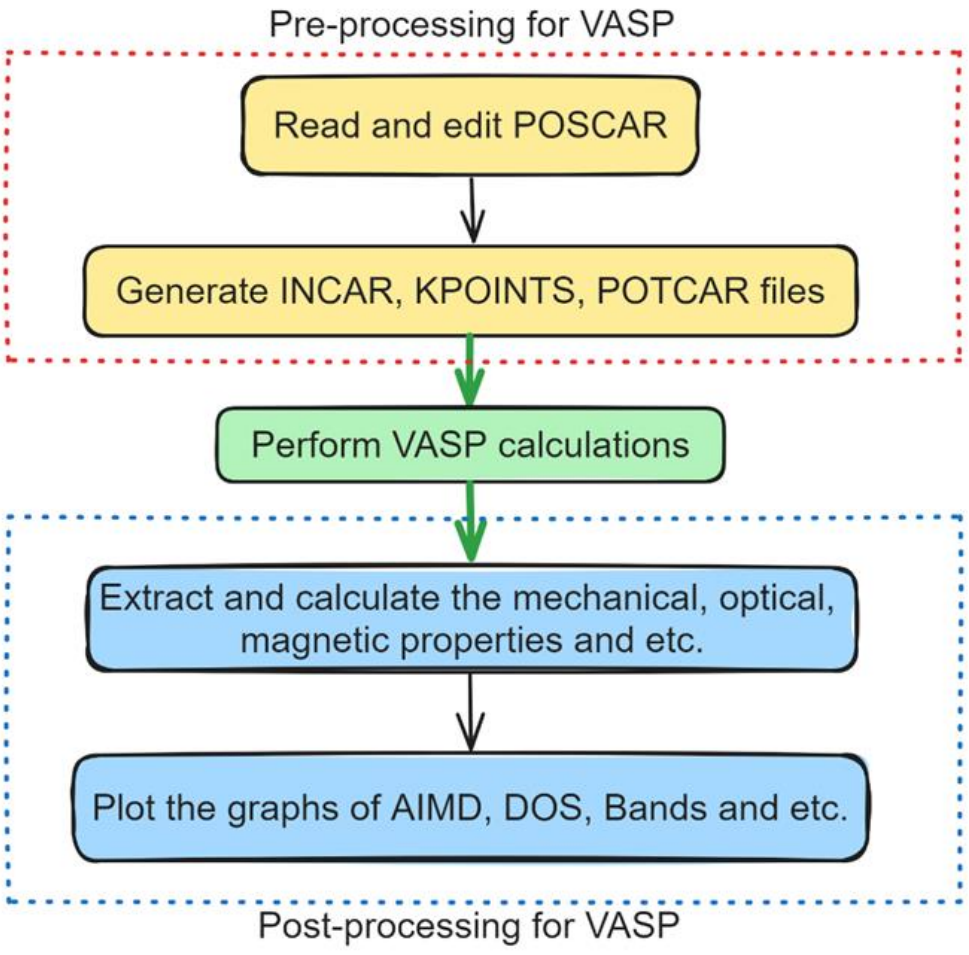
1. Assisting users to easily generate input files and process output files.
2. Help users to draw energy band diagrams, COHP diagrams, AIMD diagrams, etc.
3. Featured functions, including calculation of orbital energy levels and electron occupation number distributions, linear response method to find the U-value of DFT+U, etc.

The aim of this manual is to provide comprehensive guidance on understanding the functionality, installation process, and utilization of the HVT program. Authored primarily by H.B. He, the majority of the toolkit within the HVT program has been meticulously crafted, with additional modules sourced from the internet. In certain cases where direct communication with the original authors was not feasible due to the absence of contact information, modifications were made to ensure coherence and functionality. To mitigate any potential controversies, we are currently only offering the source codes in Bash and Python formats.

HVT is currently free and open for all to use. Since HVT introduces some code from the web, if it causes any inconvenience to the original author, please contact us, we will remove it and apologize as soon as possible.

1.2 The framework and workflow of HVT

Understanding the framework of the HVT program makes it easier for researchers to use or modify the program, and the file framework of the HVT program is similar to stacked blocks: the INCAR folder contains the INCAR template files, which can be modified according to the user's needs; the POSCAR folder contains the tools for processing the POSCAR; the KPOINTS folder contains the toolkit for generating the KPOINTS; the POTCAR folder contains the VASP pseudopotential and generation toolkit; the Tools folder contains the commonly used post-processing tools. The POTCAR folder contains the VASP pseudo-potentials and generation toolkit; the sourcecode folder contains most of the code for pre-processing and post-processing; the HVT program can quickly call the executables in the above folders based on user commands. The workflow of HVT is shown below:



1.3 How to use HVT

Users can use "**hvt -xxx**" to call different functions of HVT program. For example: "**hvt -chk**", the HVT program will call the functions of $HVT/sourcecode/vchpp to checking the consistency of POSCAR and POTCAR and the setting of KPOINTS.

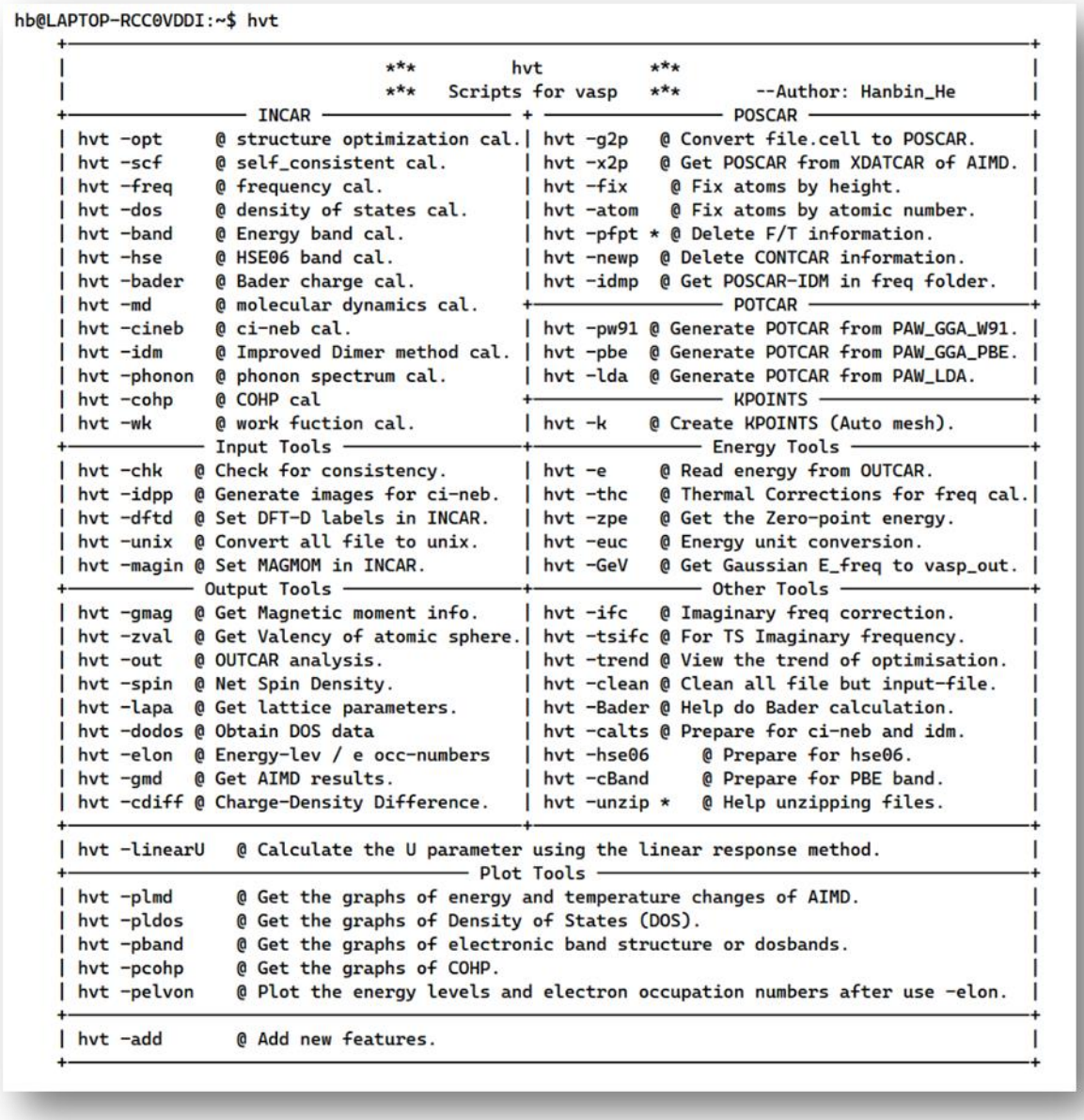
```
hb@LAPTOP-RCC0VDDI:~/hvasptools/examples/7-chk$ hvt -chk
  ........................................
  POSCAR is match to POTCAR !!!
  ****************************************
  **     GGGG   OOO    OOO   DDDD    !!!  **
  **    G      O   O  O   O  D   D   !!!  **
  **    G  GG  O   O  O   O  D    D  !!!  **
  **    G   G  O   O  O   O  D   D    !   **
  **     GGGG   OOO    OOO   DDDD     @   **
  ****************************************
  ........................................
  Elements in POSCAR--->
   N ~ 24  C ~ 48  Co ~ 3  Ca ~ 1
  Pseudopotential in POTCAR--->
   TITEL  = PAW_PBE N 08Apr2002
   TITEL  = PAW_PBE C 08Apr2002
   TITEL  = PAW_PBE Co 06Sep2000
   TITEL  = PAW_PBE Ca_sv 06Sep2000
  KPOINTS-->:
   Monkhorst-Pack :    3   3   1
  ........................................
```

In another case, the user can use the "hvt -k" command to generate the KPOINTS file after preparing the POSCAR:

```
hb@LAPTOP-RCC0VDDI:~/hvasptools/examples/3-KPOINTS$ hvt -k
--> K-Mesh Scheme
--> Please make your selection: 1. Monkhorst-Pack Scheme; 2. Gamma Scheme --> 2
--> Gamma Scheme.
--> Lattice parameters :
    a = 8.76090 , b = 25.11620 , c = 13.83450
    alpha = 90.00000 , beta = 90.00000 , gamma = 90.00000
--> Input Kmesh-Resolved Value (in Units of 2*PI/Angstrom) (reference : 0.03-0.04) --> 0.03
--> The k-mesh was calculated as: 4 1 2
--> KPOINTS was created with k-kmesh: 4 1 2 due to the material.
```

1.4 The tips document of HVT

The user can use the "hvt" command to bring up the help document, and when the user enters the wrong commands, such as "**hvt -error**", HVT program will immediately print out the help document for reference.



The HVT help document provides precise commands for executing specific tasks, enabling users to leverage the relevant functions of HVT based on their computational objectives. Furthermore, users can augment the functionality of HVT

by adding their own code to "$hvtpath/addscripts". Once added, users can grant permissions to execute this custom code by running the command "hvt -add", thereafter enabling its invocation from any location.

1.5 How to install HVT

1.First you need to download the HVT installer from Github （https://github.com/Fakescode/HVT-toolkit）.

2.Place the HVT installation package in the installation folder of your server, run “tar -zxvf hvt.tar.gz” to extract the “hvt.tar.gz” file.

3.After completing the above steps a folder named “hvasptools” will be created, go to this folder，use the command “bash install.sh” to install HVT.

Users can also add the following lines to the environment variable (e.g. ~/.bashrc):

```
######### hvt #############
 export hvtpath=/home/hb/hvasptools
 export PATH=/home/hb/hvasptools:$PATH
 export PATH=/home/hb/hvasptools/bin/Tools/vtst:$PATH
```

4.Download POTCAR files from https://www.vasp.at/ and exact in $hvtpath/bin/POTCAR accordingly.

5.Then users could relogin and use command "hvt" to get the help document.

1.6 Personalization for users

The HVT program provides user personalization settings, including:

INCAR personalization (users can directly modify or replace files in $hvtpath/bin/INCAR)

Script personalization: Users can place their own code in $hvtpath/addscripts and add executable permissions with the “hvt -add” command, and then call it from anywhere. For example, you can add “anli.sh” to $hvtpath/addscripts, and after running the “hvt -add” command, you can run “anli.sh” using the “anli.sh” command from anywhere.

2 Pre-processing function: Generating Input files

2.1 Pre-processing function for INCAR

"**hvt -opt**" can generate INCAR file for structural optimization.

```
hb@LAPTOP-RCC0VDDI:~$ hvt -opt

⟶ The INCAR for structure optimize has been generated
```

"**hvt -scf**" can generate INCAR file for single point energy calculations.
"**hvt -freq**" can generate INCAR file for frequency calculations.
"**hvt -dos**" can generate INCAR file for the density of states calculations.
"**hvt -band**" can generate INCAR files for the band structure calculation.
"**hvt -hse**" can generate INCAR file for HSE06 calculations.
"**hvt -bader**" can generate INCAR file for Bader charge calculations.
"**hvt -md**" can generate INCAR file for molecular dynamics calculations.
"**hvt -cineb**" can generate INCAR file for transition states search (ci-neb method).
"**hvt -idm**" can generate INCAR file for transition states search (Improved Dimer method).
"**hvt -phonon**" can generate INCAR file for the phonon spectrum calculations.
"**hvt -cohp**" can generate an INCAR file for Crystal orbital Hamilton population calculations.
"**hvt -wk**" can generate an INCAR file for the work function calculations.

In addition to the above function of giving INCAR directly, HVT also provides the function of suffixing INCAR labels, for example:

"**hvt -magin**" can be used to add MAGMOM tags to INCAR based on the element information in POSCAR.

```
hb@LAPTOP-RCC0VDDI:~$ hvt -magin
⟶ Elements in POSCAR:     C    Fe   N
⟶ MAGMOM = 17*0 1*6 2*0
⟶ Adjustment is required for the AFM system
```

"**hvt -dftd**" can be suffixed with a vdW corrections tag in INCAR based on element information in POSCAR (IVDW, etc.)

```
hb@LAPTOP-RCC0VDDI:~$ hvt -dftd
*********************************************************************************
*Enter the serial number of the following options to select the method you need:*
*  1.DFT-D2 method of Grimme                                                    *
*  2.DFT-D3 method of Grimme                                                    *
*  3.DFT-D3 method with Becke-Jonson damping                                    *
*  4.DFT-D4 method                                                              *
*  5.Tkatchenko-Scheffler method                                                *
*  6.Tkatchenko-Scheffler method with iterative Hirshfeld partitioning          *
*  7.Many-body dispersion energy method                                         *
*  8.dDsC dispersion correction method                                          *
*  9.Install DFT-D.sh                                                           *
*  You can enter any other character to exit                                    *
*********************************************************************************
Enter your select number >
```

For example, when using function 1 of “hvt -dftd”, HVT will set the corresponding label in INCAR according to the setting standard of DFT-D2, and the result is shown in the figure below:

```
hb@LAPTOP-RCC0VDDI:~/anli$ head POSCAR
C4
1.00000000000000
    7.4044289588999996    0.0000000000000000    0.0000000000000000
    -3.7022144793999998    6.4124235789000004    0.0000000000000000
    0.0000000000000000    0.0000000000000000   13.6389875412000006
C    Fe   N
    17     1     2
Direct
-0.0029063856159961 -0.0039404695076441  0.2325843157608666
-0.0016860312490152  0.3299612707992881  0.2306149773664951
hb@LAPTOP-RCC0VDDI:~/anli$ tail -7 INCAR
LVDW = .TRUE.
VDW_RADIUS = 30.0
VDW_SCALING = 0.75
VDW_D = 20.0

VDW_C6 = 1.75 10.8 1.23
VDW_R0 = 1.452 1.562 1.397
```

2.2 Pre-processing function for KPOINTS

The "hvt -k" function generates a KPOINTS file according to the user's specified KPOINTS type and desired K-mesh density. It's important to highlight that this function incorporates a cell calculation feature. Consequently, when there's a substantial vacuum layer in one direction of the cell, the program automatically accounts for this vacuum layer during computation. When employing "hvt -k", if the user inputs a density of 0.03, the resulting K-grid density generated by HVT will be $2\pi \times 0.03$ Å^{-1}.

```
hb@LAPTOP-RCC0VDDI:~/hvasptools/examples/3-KPOINTS$ hvt -k
⟶ K-Mesh Scheme
⟶ Please make your selection: 1. Monkhorst-Pack Scheme; 2. Gamma Scheme → 2
⟶ Gamma Scheme.
⟶ Lattice parameters :
    a = 8.76090 , b = 25.11620 , c = 13.83450
    alpha = 90.00000 , beta = 90.00000 , gamma = 90.00000
⟶ Input Kmesh-Resolved Value (in Units of 2*PI/Angstrom) (reference : 0.03-0.04) → 0.03
⟶ The k-mesh was calculated as: 4 1 2
⟶ KPOINTS was created with k-kmesh: 4 1 2 due to the material.
```

It should be noted that HVT has not developed a function to generate K-points for electronic band calculations and hse06 calculations, and K-points generated using the “hvt -k” function are not applicable to electronic band calculations and hse06 calculations.

2.3 Pre-processing function for POSCAR

“**hvt -g2p**” is to convert CASTEP cell file to POSCAR.

```
hb@LAPTOP-RCC0VDDI:~$ hvt -g2p

Notice!  >>>   *.cell will be converted to POSCAR

*********************************************************************
* This is the script to convert CASTEP cell file to vasp POSCAR. *
*>>>>>>>>>>> Converting CASTEP cell file to POSCAR ... <<<<<<<<<<*
*                        POSCAR Generated.                          *
*********************************************************************
```

"**hvt -x2p**" can extract the POSCAR of the formulation step or all the POSCARs of the overall change process from the XDATCAR calculated by AIMD.

```
hb@LAPTOP-RCC0VDDI:~/work/MD$ hvt -x2p
+-------------------------------------------------------+
|1.Generate all images | 2.Generate selected images|
+-------------------------------------------------------+
Enter your select >
```

"**hvt -fix input.vasp**" can fix the atomic coordinates in the certain surface model layers of POSCAR by height.

```
hb@LAPTOP-RCC0VDDI:~/hvasptools/examples/2-POSCAR/1-fix-high$ hvt -fix
Selecting the direction to be fixed(x,y,z) > y
⟶ Atoms below your chosen section in y direction will be fixed
⟶ Type y_max, Note: y_max (0-1) > 0.2
```

"**hvt -atom**" can fix the atomic coordinates in the certain surface model layers of POSCAR by atom number.

```
hb@LAPTOP-RCC0VDDI:~/mywork/hvasptools/examples/2-POSCAR/2-fix-select$ hvt -atom
⟶  Fixed coordinates according to atomic number.
⟶  Enter the serial number of the atom to be processed (e.g. 1 3 5 7)
⟶  Select: 2 4 13 34 12
⟶  Operations on selected atoms (1:release-T/2:fix-F) → 2
⟶  Finish, output file: POSCAR-FIX
```

It is worth noting that here the atoms are fixed or released based on the user-entered atomic numbers, which are allowed to be disordered or repeated.

"**hvt -pfpt filename**" is to delete the F/T information in POSCAR.

"**hvt -newp**" can convert CONTCAR to POSCAR and remove the calculation information from CONTCAR.

"**hvt -idmp**" is a feature of HVT that can be used to generate the POSCAR required for IDM calculations. It is important to note that this feature must be used in the folder of vibration

calculations after the vibration calculations have been completed for the initial structure.

```
hb@LAPTOP-RCC0VDDI:~/work/MD$ hvt -idmp
+-------------------------------------------------------------+
|          Script used to generate POSCAR for idm             |
+-------------------------------------------------------------+
——>1.Refresh and generate POSCAR-dimer ...
——>2.Add dxdydz obtained from freq to POSCAR-dimer ...
——>3.The Max energy of the imaginary vibrational mode = 4.020298
——>4.Operation complete, output file: POSCAR-dimer
```

2.4 Pre-processing function for POTCAR

Required to use this feature: Download potential (POTCAR) files from https://www.vasp.at/ and exact in $hvtpath/bin/POTCAR accordingly.

"**hvt -pbe**" can generate PAW_PBE pseudopotential.

```
hb@LAPTOP-RCC0VDDI:~/POTCAR/test$ hvt -pbe
hb@LAPTOP-RCC0VDDI:~/POTCAR/test$ ls
POSCAR  POTCAR
hb@LAPTOP-RCC0VDDI:~/POTCAR/test$ sed -n 6p POSCAR
N    C    Fe
hb@LAPTOP-RCC0VDDI:~/POTCAR/test$ grep TITEL POTCAR
    TITEL  = PAW_PBE N 08Apr2002
    TITEL  = PAW_PBE C 08Apr2002
    TITEL  = PAW_PBE Fe 06Sep2000
```

The other two functions (**hvt -pw91**, **hvt -lda**) are similar and will not be repeated.

3 Postprocessing Output files

3.1 Postprocessing toolkits

"**hvt -gmag**" can read the magnetic moment information from OUTCAR and output to the magnetization file.

```
hb@LAPTOP-RCC0VDDI:~/POTCAR/test$ hvt -gmag
⟶Script for extracting magnetization
⟶Operation complete, output file: magnetization
```

"**hvt -zval**" can get the Valency of atomic sphere.

```
hb@LAPTOP-RCC0VDDI:~/POTCAR/test$ hvt -zval
Valency of each atomic sphere:
N ~ 5.000  C ~ 4.000  Fe ~ 8.000
```

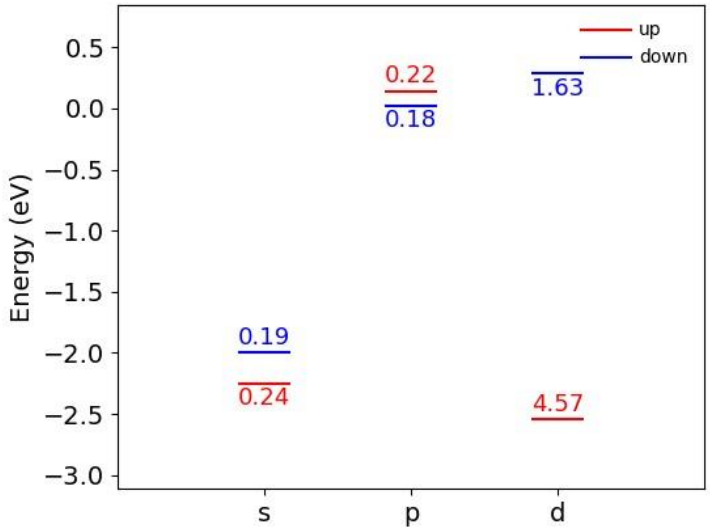
"**hvt -lapa**" can get the lattice parameters of POSCAR or CONTCAR.

```
hb@LAPTOP-RCC0VDDI:~/POTCAR/test$ hvt -lapa
⟶ Select file: 1. POSCAR; 2. CONTCAR → 1
⟶ Lattice parameters :
    a = 16.65310 , b = 16.65350 , c = 20.12110
    alpha = 89.92910 , beta = 89.85490 , gamma = 120.00180
```

"**hvt -dodos**" can split DOS by different atoms or classes of atoms.

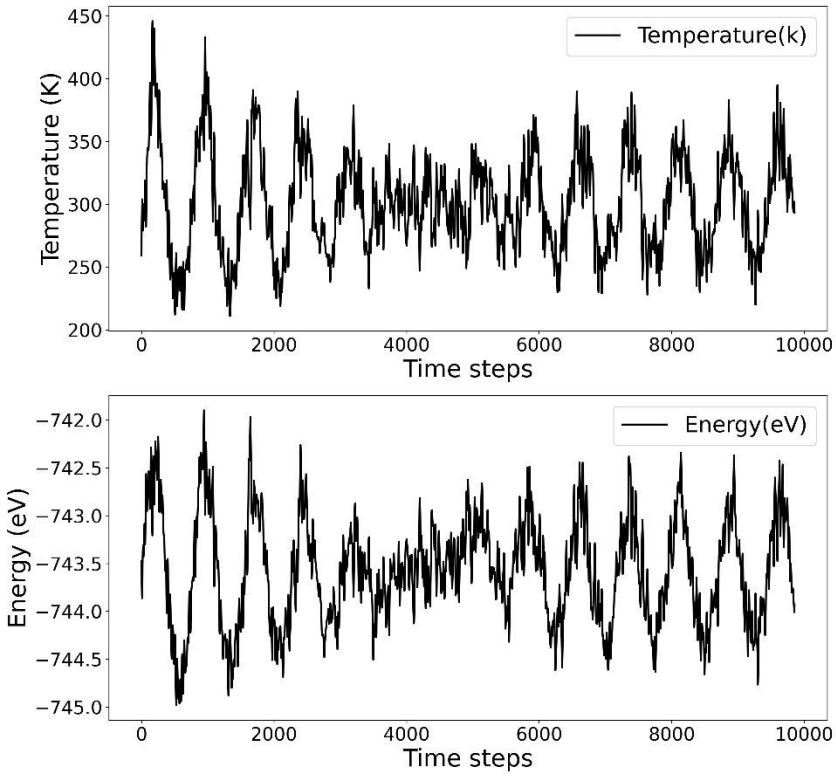
"**hvt -elon**" can get the energy levels and electron occupation numbers for every orbital after DOS calculation. It should be noted here that this function is best used in conjunction with the vaspkit-115 function. After using "hvt -elon", you can run "**hvt -pelvon**" to do the plotting. These two functions are unique features of the HVT.

```
hb@LAPTOP-RCC0VDDI:~/work/energy-levels-electron-occupation-numbers$ hvt -elon
⟶ The file containing the dos data: PDOS_USER.dat
⟶ Output: Elv_OccN_results.dat
#Orbital    Energy LV  Occ Num
up_74_s  -2.249467   0.241014
dw_74_s  -1.995911   0.190995
up_74_p   0.142659   0.220755
dw_74_p   0.023700   0.175333
up_74_d  -2.545828   4.570789
dw_74_d   0.283550   1.629907
hb@LAPTOP-RCC0VDDI:~/work/energy-levels-electron-occupation-numbers$ hvt -pelvon
⟶ Finish
```



"**hvt -gmd**" can extracting the data calculated by AIMD. This function can be used in conjunction with "**hvt -plmd**", which runs and plots the extracted data.

```
hb@LAPTOP-RCC0VDDI:~/work/1-MD$ hvt -gmd
⟶Script for extracting AIMD data
⟶Operation complete, output file: AIMDout.dat
hb@LAPTOP-RCC0VDDI:~/work/1-MD$ hvt -plmd
⟶ Now reading vasp MD energies and temperature ...
⟶ Output: AIMDout.dat
⟶ Start drawing ...
⟶ Output: Temperature.png & Energy.png
```



"**hvt -cdiff**" can calculate the Charge-Density Difference. After using this function, import the output "CHGCAR_diff.vasp" into VESTA to get the charge density difference figure.

```
hb@LAPTOP-RCC0VDDI:~/work/chgdiff$ hvt -cdiff
⟶ Charge-Density Difference
⟶ delta_rho = rho_AB - rho_A - rho_B
⟶ rho_AB = 1-CHGCAR
⟶ rho_A  = 2-CHGCAR
⟶ rho_B  = 3-CHGCAR
⟶ Reading Structural Parameters ...
⟶ Reading Charge Density ...
⟶ Output: CHGCAR_diff.vasp
```

"**hvt -ifc**" and "**hvt -tsifc**" is designed for Imaginary frequency. This function empirically tries to remove multiple imaginary frequencies of the searched reaction transition state by entering the command directly into the Frequency Calculation folder and entering a correction factor in the range of 0.0 to 0.5. This function generates a new POSCAR, which then allows the user to calculate

the new frequency.

```
hb@LAPTOP-RCC0VDDI:~/work$ hvt -tsifc
1.Refresh and generate a new POSCAR ...
2.Deal dxdydz obtained from freq ...
3.The energy of the first imaginary vibrational mode = 6.401653
4.The energy of the second imaginary vibrational mode = 32.040490
5.Detected Selective dynamics ...
6.Generate new xyz, please wait ...
7.Please input the correction facter(0-0.5) :  0.3
8.Processing atom : [175]-[step]
9.Processing POSCAR : [176]/[176]
10.Operation complete, output file: POSCAR-out
>>Prayers and good luck for the next calculation<<
```

"**hvt -e**" can read total energy from OUTCAR.

"**hvt -calts**" is used to prepare files for ci-neb and idm calculations. The corresponding folder will be created automatically after using this function, and the user can perform calculations in the corresponding folder.

```
hb@LAPTOP-RCC0VDDI:~/work$ hvt -calts
⟶ initial structure : POSCAR-fin
⟶ final structure : POSCAR-ini
⟶ dist = 4.30089256407513
⟶ Recommended insertion points : 6
⟶ Finish, please check
⟶ The INCAR for CI-NEB has been generated
⟶ The INCAR for frequency has been generated
hb@LAPTOP-RCC0VDDI:~/work$ ls
POSCAR-fin  POSCAR-ini  cineb  freq-cineb  idm
```

It is worth noting that this feature will help users to complete the image creation of CI-NEB.

The function of "**hvt -hse06**" and "**hvt -cBand**" is similar to that of "**hvt -calts**" and will not be repeated.

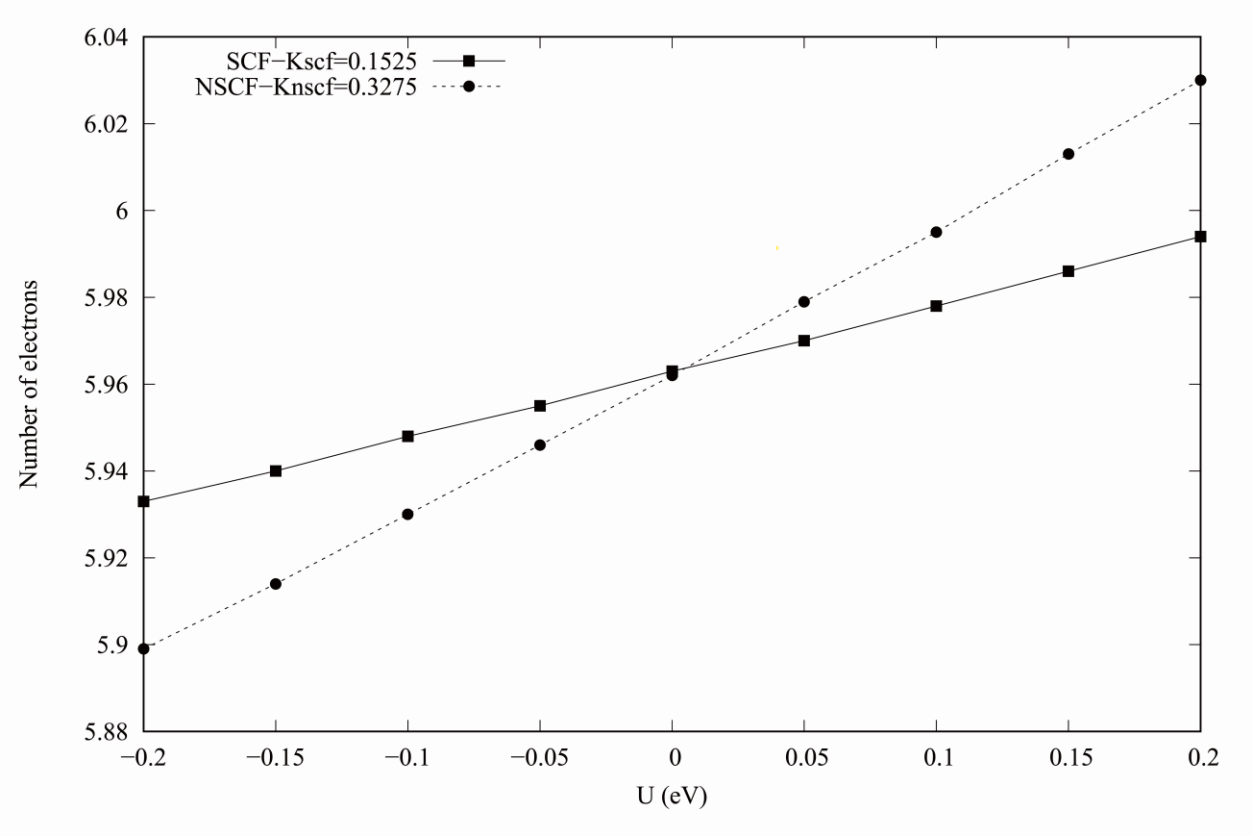
"**hvt -linearU**" is a feature of HVT that is used to perform linear response method calculations to obtain the U-value for performing DFT+U calculations.

```
hb@LAPTOP-RCC0VDDI:~/work$ hvt -linearU
*** Script for Linear response calculation of U ***
--> Select the calculation you want to perform:
    0.Instructions;
    1.Input example;
    2.DFT groudstate;
    3.SCF and NSCF for DFT+U(After 2);
    4.Submit all calculations(After 3);
    5.Extract Ueff data(After 3).
    6.Linear fitting of the potential and the number of electrons (drawing)
-->
```

This function provides detailed aids to the calculation of the linear response method. The main menu of "hvt -linearU" provides detailed instructions and calculation examples, as well as preparation and processing functions for each step of the calculation.

After completing the self-consistent and non-self-consistent calculations of the linear response method, the user can run function 5 to obtain and process the results of the linear response method calculations and output them to the Uout.dat file, and run function 6 to plot the results.

```
hb@LAPTOP-RCC0VDD:~/work$ cat Udata.log
   U        DFT       NSCF      SCF
 ------    ------    ------    ------
 -0.20     5.963     5.899     5.933
 -0.15     5.963     5.914     5.940
 -0.10     5.963     5.930     5.948
 -0.05     5.963     5.946     5.956
  0.00     5.963     5.962     5.963
  0.05     5.963     5.979     5.970
  0.10     5.963     5.962     5.963
  0.15     5.963     5.962     5.963
  0.20     5.963     6.030     5.994
 ------    ------    ------    ------
 Ueff:   3.50394
```

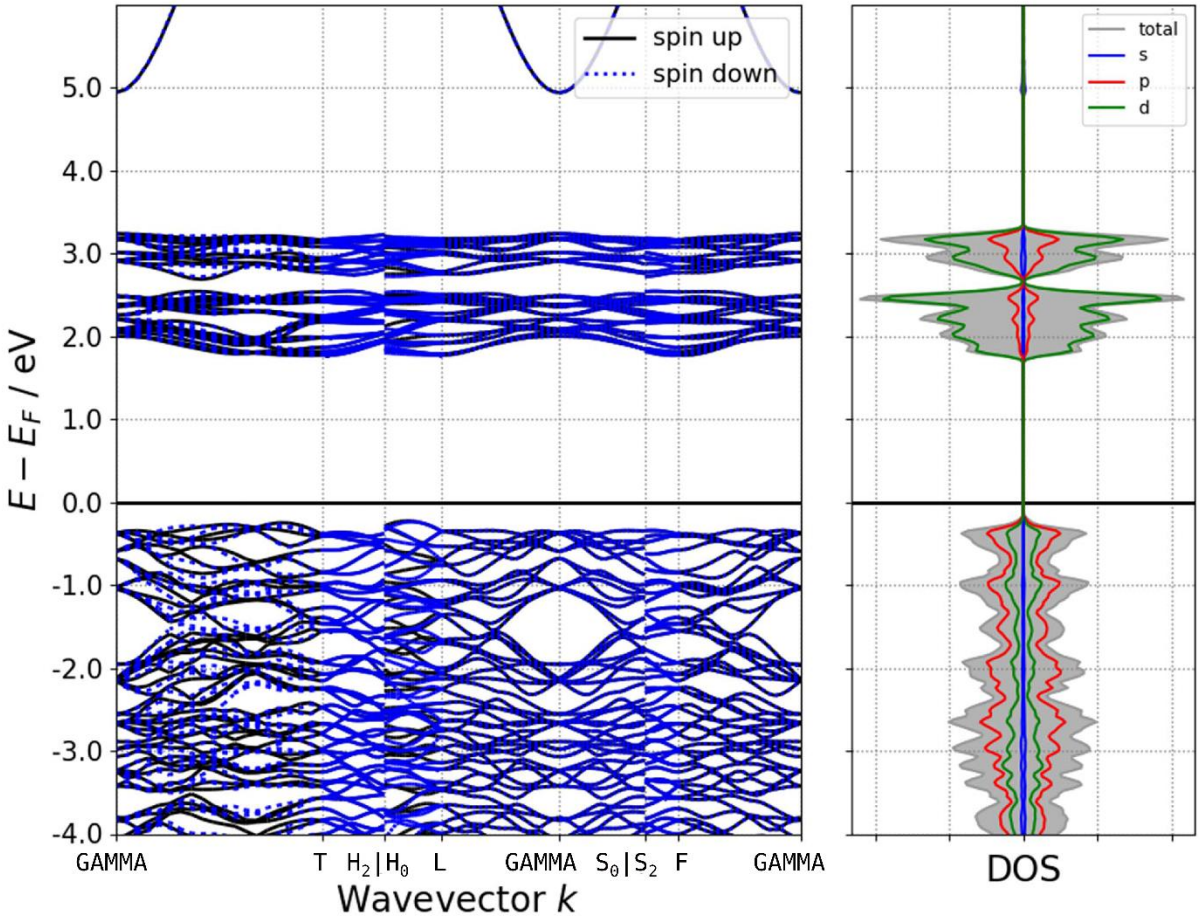


We have carried out some validation of this function, and the results show that the U-values calculated by this function are sufficiently accurate, e.g., the two figures above show the calculated U-values of Fe for Fe_2O_3, which are in general agreement with the literature results.

In addition to the above functions, HVT provides functions such as band plotting and COHP plotting, which can be accomplished by directly running the corresponding commands.

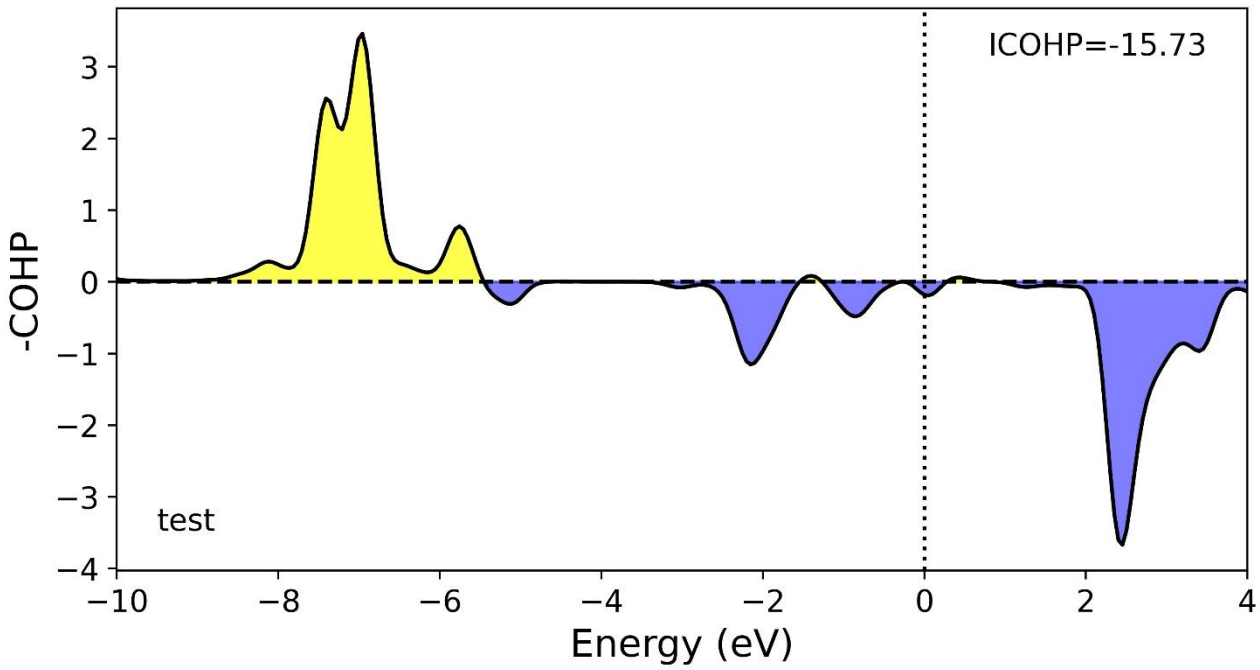
"**hvt -pband**" provides a function to draw a map, run this function and select the style of drawing according to the need to get the corresponding map. It is important to note that this feature only provides plotting functionality and does not provide the ability to extract data.

```
hb@LAPTOP-RCC0VDD:~/work$  hvt -pband
⟶ Please select a drawing format:
⟶ 1. Normal energy band diagram.
⟶ 2. Energy band diagrams projected onto elements.
⟶ 3. Banddos diagram (band: ordinary; dos: elemental).
⟶ 4. Banddos diagram (band: elemental; dos: elemental).
⟶ 5. Banddos diagram (band: elemental; dos: orbit).
⟶ Select: 5
⟶ Getting band and dos information from vasprun.xml  ...
⟶ Start drawing  ...
⟶ Operation complete, output: dosbands.png
```



"**hvt -pcohp**" provides COHP drawing function, the function has more complete settings, the user can first use its function 1 to generate sketches, and then according to the sketches to use function 2 to adjust the font size and other settings in the diagram.

```
hb@LAPTOP-RCC0VDD:~/work$ hvt -pcohp
⟶ @ 1. Default, Drawing with default parameters for correction.
    @ 2. Inputting correction parameters to modify the COHP diagram.
    @ 3. Drawing with the output file to other drawing software.
⟶ select :  2
⟶ Now reading COHPCAR.lobster ...
⟶ Data processing, output: COHPCAR.plot
⟶ Now reading ICOHPCAR.lobster ...
⟶ Maximum value of x-coordinate :  4
⟶ Minimum value of x-coordinate :  -10
⟶ Name of the figure : test
⟶ Setting the font size of scale values(xticks and yticks)(default : 12): 12
⟶ Setting the font size of ICOHP label(default : 12): 12
⟶ Setting the font size of figure name(default : 12): 12
⟶ Setting the font size of xlabel and ylabel(default : 15): 15
⟶ Start drawing ...
⟶ Done, output: COHP.png
```



3.2 Other features.

The authors encountered many problems in the process of learning material simulation, and thus only made some targeted functions for certain problems, for example, “hvt -GeV” can extract information about zero-point energy, entropy, enthalpy, etc. in the result of opt+freq calculated by Gaussian and print it on the screen; “hvt -euc” provides the function of energy turn-off, run this function and input according to the prompts.

```
hb@LAPTOP-RCC0VDD:~/work$ hvt -GeV
+-------------------------------------------------------------------------------------+
Sum of electronic and zero-point Energies=   → U(0) = E_DFT + ZPE
Sum of electronic and thermal Energies=      → U(T) = E_DFT + ZPE + deltaU[0→T]
Sum of electronic and thermal Enthalpies=    → H(T) =  E_DFT + ZPE + deltaH[0→T]
Sum of electronic and thermal Free Energies= → G(T) = E_DFT + ZPE + deltaG[0→T]
+-------------------------------------------------------------------------------------+
NH3.log :
-------------------------------------------------------------------------------
.      E_ZPE      U[0]         U[T]        H[T]         G[T]       Units     .
.    0.034770   -56.532443 -56.529584 -56.528640 -56.550472 Hartrees    .
-------------------------------------------------------------------------------
NH3.log ⟶ Summary:
-----------------------------------------------------------------
.    E_ZPE      TS         DelatH     DelatG     Units       .
.    0.034770   0.021832   0.003803   0.016741   Hartrees    .
.    0.946140   0.594079   0.103485   0.455546   eV          .
-----------------------------------------------------------------
+-------------------------------------------------------------------------------------+
```

Acknowledgements

The software is not yet perfect and there are still some minor problems, which we will finish fixing in the next update. Note that this article may not be included in any organization or repository without permission.